Modeling Aspects of Human Trails on the Web

Dipl.-Ing. Philipp Singer, BSc

Dissertation
zur Verleihung des akademischen Grades
Doktor der technischen Wissenschaften

Knowledge Technologies Institute
Technische Universität Graz

Begutachter: Univ.-Ass. Dipl.-Ing. Dr.techn. Markus Strohmaier

Graz, Dezember 2014

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Graz, am …………………………… ………………………………………………..
(Unterschrift)

Englische Fassung:

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

…………………………… ………………………………………………..
date (signature)

Abstract

When humans engage with the Web, they produce sequential digital trails on a massive scale. Examples of such human trails on the Web are websites humans navigate, consecutive businesses they review or successive songs they listen. Studying human trails has been of interest to our research community since the advent of the World Wide Web; different models, insights and hypotheses have emerged. While many of them are advanced and well studied, some of them warrant further investigations. This thesis deals with modeling aspects of human trails based on this challenge. Mainly, it provides methodological contributions facilitating future research concerned with the analysis of patterns, regularities and strategies in human trails on the Web focusing on some sub-problems. (i) First, this thesis deals with the open question of whether human trails on the Web exhibit memory effects which is an important factor for the Markov chain model that is frequently applied to human trails on the Web. Predominantly, the Markov chain model has been memoryless in a wide range of applications such as Google's PageRank. The usefulness of this memoryless property has been discussed without much consensus in the past. To that end, this thesis presents a framework that allows researchers to comprehensively evaluate the appropriate Markov chain order based on several advanced statistical inference methods. (ii) Apart from memory effects, it is partially unclear whether other structural patterns in human trails might be utilized beyond what previous work has done. This thesis tackles this by exemplarily demonstrating that human navigational trails on the Web can be leveraged for the task of calculating semantic relatedness between concepts. Thus, this thesis argues for an expansion of the existing arsenal of Web data sources to also consider human trails on the Web. (iii) Given these structural and behavioral investigations, it has been of interest to our research community to further understand how human trails on the Web are produced. However, it has been difficult to make informed decisions about hypotheses regarding this production (beliefs in transitions). Thus, this thesis presents HypTrails, an approach for expressing and comparing hypotheses about human trails on the Web. Overall, the aspects presented in this thesis are relevant for researchers interested in studying and modeling human trails on the Web.

Kurzfassung

Durch die Kommunikation von Menschen mit dem Web wird eine Vielzahl von sequentiellen, digitalen Pfaden produziert. Beispiele solcher menschlichen Pfade im Web sind navigierte Webseiten, aufeinanderfolgende Unternehmen welche Menschen bewerten oder Folgen von Liedern die sie hören. Seit Bestehen des World Wide Webs ist die Studie menschlicher Pfade ein wichtiger Aspekt unserer Forschungsgemeinschaft wodurch verschiedene Modelle, Erkenntnisse und Hypothesen entstanden sind. Obwohl viele von ihnen gründlich untersucht und weit entwickelt sind, erfordern einige von ihnen weitere Untersuchungen. Basierend auf dieser Herausforderung beschäftigt sich diese Dissertation mit der Modellierung verschiedener Aspekte menschlicher Pfade. Sie präsentiert eine Reihe von methodologischen Beiträgen, welche zukünftige Forschung bezüglich Muster, Regelmäßigkeiten und Strategien im menschlichen Pfaden erleichtern sollen. (i) Zu Beginn beschäftigt sich diese Arbeit mit der offenen Frage ob menschliche Pfade Gedächtniseffekte aufweisen, welche ein wichtiger Faktor für Markov Ketten Modelle darstellen. Das Markov Ketten Modell ist ein prominentes Modell für menschliche Pfade im Web, jedoch ist es in einer Vielzahl von Applikationen (z.B. Google's PageRank) gedächtnislos. Die Nützlichkeit dieser gedächtnislosen Eigenschaft wurde in einigen Studien ohne klaren Konsens untersucht. Dementsprechend präsentiert diese Arbeit ein Framework, welches Forschern ermöglicht umfassende Einsichten in die Evaluierung der geeigneten Markov Ketten Ordnung anhand verschiedener statistischer Inferenzmethoden zu erlangen. (ii) Abgesehen davon, ist es teilweise unklar ob andere strukturelle Muster menschlicher Pfade über bisherige Anwendungen hinaus nutzbar gemacht werden können. Um dies zu untersuchen, demonstriert diese Dissertation beispielsweise, dass menschliche Navigationspfade im Web für die Bestimmung semantischer Ähnlichkeit zwischen Konzepten nutzbar gemacht werden können. Folglich argumentiert diese Arbeit, dass das bestehende Arsenal an Web Datenquellen auf menschliche Pfade des Webs erweitert werden soll. (iii) Gegeben dieser strukturellen und verhaltensbezogenen Untersuchungen ist es für unsere Forschungsgemeinschaft auch von Bedeutung ein erweitertes Verständnis der Produktion menschlicher Pfade zu erlangen. Jedoch ist es schwierig fundierte Entscheidungen bezüglich Hypothesen (Glaube an

Transitionen) dieser Produktion zu treffen. Um dieses Problem zu adressieren, präsentiert diese Dissertation HypTrails, ein Ansatz der Forschern erlaubt Hypothesen über menschliches Verhalten auszudrücken und zu vergleichen. Gesamt gesehen sind die Aspekte dieser Dissertation von Relevanz für Forscher, welche an der Analyse und Modellierung menschlicher Pfade am Web interessiert sind.

Acknowledgements

First an foremost, I would like to sincerely thank my PhD adviser Markus Strohmaier for his excellent guidance throughout the course of my PhD studies. Without his excellent feedback, ideas and input this thesis would not have been possible.

Further, I am very grateful for the collaboration with Denis Helic from the Technical University of Graz. I could always consult Denis when I had some uncertainties with methodological concepts and his input has been very valuable to me. Also, I am grateful for the collaboration with Simon Walk who helped me with applying the developed methods of this thesis to problems that matter and hence, demonstrating the usefulness of what I have done in this thesis.

I also want to thank the Data Mining and Information Retrieval Group of the University of Würzburg as well as the Knowledge and Data Engineering Group of the University of Kassel for the warm welcome as well as excellent collaboration not only during my time in Kassel and Würzburg but also throughout the whole course of my thesis. Specifically, I want to thank Andreas Hotho for his insights, inspiration and guidance as well as his students Thomas Niebler, Stephan Dörfel and Daniel Zoller.

I would also like to express my gratitude to several colleagues I have met and co-worked with during my stay at the Technical University of Graz as well as at GESIS in Cologne. Especially, I want to thank Claudia Wagner who has been very supportive of me and who has been a very important factor for my thesis. Also, I want to thank my old office colleagues in Graz Florian Geigl, Florian Klien, Daniel Lamprecht, Silvia Mitter and Lisa Posch for their help and constructive feedback as well as Christian Körner who guided me at the beginning of my Phd studies. Additional thanks go out to Fabian Flöck whom I met during a conference visit and who has become a colleague, friend and room mate.

Finally, I want to thank my family for their help and their nonstop availability in case of trouble. Angelika, thank you for your support, kindness and positivity. Without you all, this thesis would not have been possible.

Institutional acknowledgements. This thesis was partially funded by the FWF Austrian Science Fund Grant I677 PoSTs “Pragmatics and Semantics in Social Tagging Systems” as well as by the DFG German Science Fund Grant STR 1191/3-2 PoSTs II ”Pragmatics and Semantics in Social Tagging Systems II”. Furthermore, I want to thank the Technical University of Graz as well as the Leibniz-Institute for the Social Sciences (GESIS) for giving me the opportunity to do this thesis.

Contents

1. Introduction

1.1. Motivation

In 1945, Vannevar Bush introduced the hypothetical system called *memex* in his seminal work "As We May Think" [Bush, 1945]. The word memex is a portmanteau of the words *memory* and *index* or *index*. Bush argued that the human brain operates by association based on an internal web of trails which it leverages for associative reasoning. In this web, frequent trails influence and dominate a person's decisions while infrequent ones become irrelevant. Bush prototyped the memex as a large device (actually a desk) that allows humans to store books and other manuscripts and which represents an "enlarged intimate supplement to a person's memory". In analogy to the brain's web of trails, his idea was to allow humans to store common trails such as sequences of books retrieved. These stored trails should then facilitate later retrieval and sharing. Bush also proposed a potential new profession called *trail blazers* who should store common transitions and then share them with others.

Decades later, the ideas of Vannevar Bush led to the development and concept of Hypertext [Nelson, 1965] as well as the World Wide Web [Berners-Lee and Fischetti, 2000]. In these new information structures, humans produce sequential digital trails on a much more massive scale than possibly ever imagined by Bush. Some examples of such emerging *human trails on the Web* are: consecutive websites humans navigate, sequences of friends they add on social media sites like Facebook or successive songs they play in online music services.

Understanding human trails and how they are produced has been a complex challenge for researchers for years. Several prominent models such as Google's PageRank [Brin and Page, 1998] or advertising models [Archak et al., 2010] are based on assumptions about successive human behavior on the Web. Consequently, it is crucial for studying human trails on the Web for gaining insights into humans' associative reasonings and behavioral aspects. Apart from mentioned examples, acquired findings can e.g., be useful for enhancing information network structures [Borges and Levene, 2000; Perkowitz and Etzioni, 1997], for predicting human clicks [Bestavros, 1995] or for recommender systems [Rendle et al., 2010]. This thesis deals with modeling aspects of human trails on the Web based on this larger challenge. Primarily, it aims at providing tools for facilitating future research concerned with the analysis of patterns, regularities and strategies in human trails. Thus, this thesis is relevant for researchers interested in studying human trails on the Web.

In the following Section 1.2, this thesis provides a short overview of what human trails are and challenges and opportunities they implicate. Subsequently, an overview of the problem statement, the objectives and main approach of this thesis is provided in Section 1.3 before this thesis elaborates the exact research questions in detail in Section 1.4. All main publications part of this cumulative thesis are listed in Section 1.5, the main contributions are emphasized in Section 1.6 and the structure of this thesis is outlined in Section 1.7.

1.2. Human Trails on the Web

This thesis considers human trails as *sequences of at least two consecutive states* which are produced when humans interact with the Web. States are observed from a given categorical and finite state space. In Figure 1.1, an illustration of the general schema of human trails is provided. It depicts two different trails that are produced by two distinct persons. Overall, five different states are considered. The type of states depends on the kind of trails one is interested in. Next, a series of types of human trails are introduced which are also subject to this thesis that mainly focuses on human navigational trails.

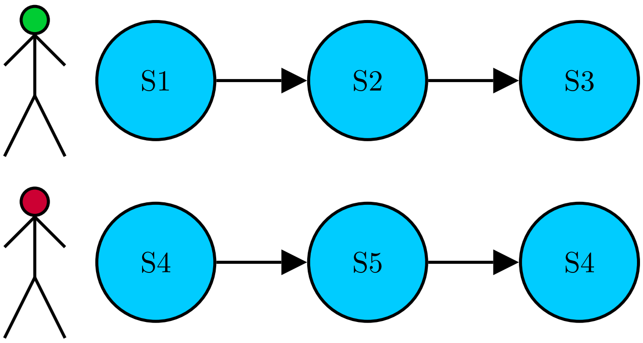


Figure 1.1.: **General schema of human trails.** This figure depicts an example of two human trails produced. The state space consists of five different states $S = \{S1, S2, S3, S4, S5\}$. The type of states depends on the type of trails one is interested in. For example, let us suppose that states refer to five distinct songs listened to on Last.fm. The first person (top row, green) starts by listening to song S1, before she listens to song S2 and finally to song S3. The second person (bottom row, red) first listens to song S2 before she plays song S5. Finally, the person again plays track S2. States can refer to other kinds of categorical observations such as websites or other entities such as songs.

1.2.1. Human navigational trails

Navigating websites represents one of the most fundamental interactions of humans with the Web and is important to study [Huberman et al., 1998]. By navigating the Web, humans produce *human navigational trails*. Thus, the state space consists of all sites humans navigated over in observed data. As an example, suppose a person currently is on Facebook and sees a funny video on Youtube posted by a friend. The person decides to click on the hyperlink leading to corresponding video. By doing so, she has performed a navigational step producing a two-step navigational trail. For matters of simplicity, this thesis focuses on studying *intra navigational trails* consisting of navigational steps between sites of a single platform (e.g., Wikipedia). Figure 1.2 depicts an illustration of an exemplary trail over three consecutive Wikipedia pages visited: *Austria*, *Germany* and *Carl Friedrich Gauss*.

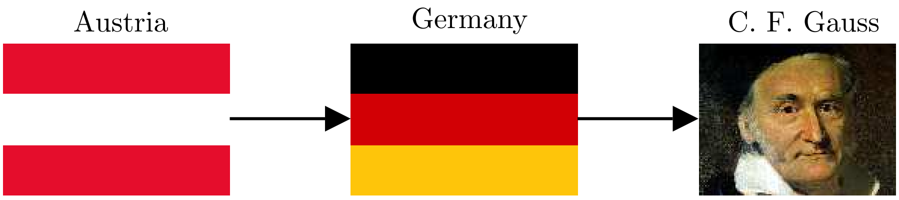


Figure 1.2.: **Example of human navigational trail on Wikipedia.** This figure depicts an example of one human navigational trail over Wikipedia pages produced by one person. The person starts on the Wikipedia page of *Austria*; then she decides to click on the hyperlink on Austria's page leading to *Germany*. Let us assume that the person is very interested in learning more about important researchers of Germany and hence, clicks on the hyperlink leading to the Wikipedia page of *Carl Friedrich Gauss*. After that, the person leaves Wikipedia and her navigational steps have produced the intra navigational trail consisting of three subsequent Wikipedia pages visited as depicted.

1.2.2. Human edit trails in collaborative ontology engineering projects.

Humans not only click, but also edit content on the Web such as concepts on Wikipedia or on ontologies. As an example, *human edit trails in collaborative ontology engineering projects* are produced by humans consecutively editing concepts or properties in ontology projects. Several different types of edit trails can be derived from usage logs of ontology engineering projects. The two main types are (i) *class-based* and (ii) *person-based* edit trails. The former capture the chronology of a specific feature of all changes that were performed *by any person on a single class*, while the latter depict the ordered list of specific features of changes that were performed *on any class by a single person.* Exemplary features are the properties of classes edited by humans. In that case, the state space at interest consists of all properties edited. Figure 1.3 illustrates two examples of such human edit trails (class and person-based).

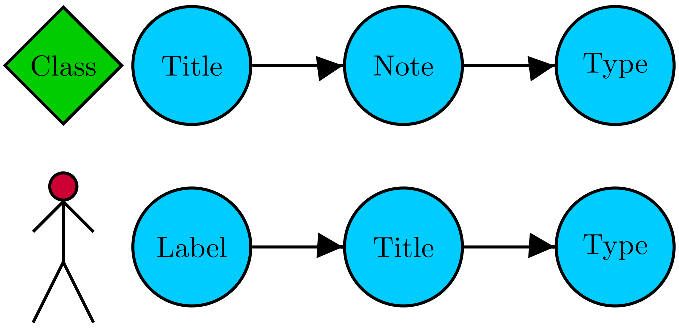


Figure 1.3.: **Example of human edit trails in collaborative ontology engineering projects.** This figure depicts examples of human trails that are produced when humans perform edits in collaborative ontology engineering projects. The first row shows a class-based trail where properties of the given class are consecutively changed by any person. In this example, some person first edited the property *Title* before another one edited the property *Note* and finally, some other arbitrary person edited the *Type* property. The second row illustrates a person-based trail which covers subsequent properties changed by one person on any class. In this case, the person first edited the *Label* property on some class, before she edited the *Title* property and finally the *Type* property of some other classes.

1.2.3. Human business review trails.

Nowadays, humans frequently consult the Web for getting recommendations about businesses, products, movies, songs or other things. For doing so, they often resort to reviews given by other humans. A series of platforms offer reviewing mechanisms such as Yelp for businesses, IMDB for movies or Amazon for products. When providing reviews, humans do that in successive manner which produces human review trails. As an example, *human business review trails* on Yelp consist of trails that capture the business reviewing history of a single person in some given time frame. Figure 1.4 provides an example of such a trail for which the state space consists of five different businesses – in this case restaurants – a person reviewed during a trip to Italy.

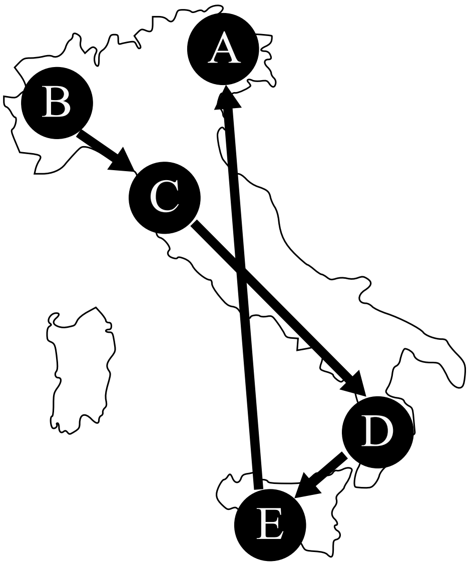


Figure 1.4.: **Example of human business review trail.** This figure depicts an example of a person successively reviewing five different businesses. In this example, suppose someone is traveling through Italy and reviews the restaurants visited on a platform like Yelp. The person starts in the north and travels to the south. Hence, she first reviews restaurant *B*, then *C*, *D* and finally, in the south restaurant *E*. At the end of the trip, the person is going back north and before leaving Italy she reviews restaurant *A*.

1.2.4. Human listening trails.

Several Web platforms allow humans to listen to music. For example, on Last.fm, humans can listen to songs and use a comprehensive recommendation system. On YouTube, they can listen to songs by watching the corresponding music videos and Spotify offers a large library of songs to listen to by paying a monthly fee. By consecutively listening to songs, humans produce what we call *human listening trails*. For example, such trails may capture the history of songs played on a given time frame or for a given playlist. Figure 1.5 provides a simple example of such a human trail that consists of three consecutive songs listened to. Hence, the resulting state space consists of distinct songs that humans have listened to.

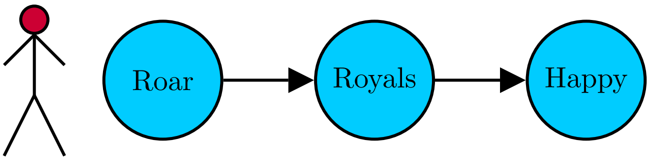


Figure 1.5.: **Example of human listening trail.** This figure depicts an example of a human trail that is produced by a person successively listening to different songs. The person starts to listen to the song *Roar* by *Katy Perry* before she listens to *Royals* by *Lorde* and finally to *Happy* by *Pharrell Williams.*

1.2.5. Challenges and Opportunities

The production of human trails on the Web is omnipresent. The examples provided in Section 1.1 as well as Section 1.2 are just a small selection and part of a massive set of diverse human trails. Due to the importance of gaining a better understanding of the production of these trails, a series of previous works has focused on studying human trails on the Web in various aspects. In the rest of this section, I want to highlight some few challenges and opportunities that this thesis deals with which are mostly motivated by related work. Based on these, I introduce the problem statement, objectives and general approach of this thesis in Section 1.3 before I present the detailed research questions in Section 1.4.

In early work, Huberman et al. [1998] emphasized that strong regularities in World Wide Web surfing behavior of humans exists. Such online behavioral regularities were also observed and confirmed by the work of Wang and Huberman [2012]. Actually, we might even say that these regularities are reasoned by inherent regularities of human behavior in general as pointed out by Song et al. [2010]. For handling the huge amounts of information on the Web, humans might also follow certain strategies [Chi et al., 2001]. Yet, one can hypothesize that our research community has not fully utilized these opportunities that human trails on the Web offer.

For instance, many prominent models such as Google's PageRank [Brin and Page, 1998] assume that humans act in a memoryless way. This means that they make their next choice only based on their current one and not on a series of preceding ones. This assumption is rooted in the Markov chain model which is prominently used for modeling human trails on the Web. Given the inherent regularities of human trails as mentioned, one might argue that memory may play a more crucial role for humans as considered. This has also been studied in the past (see e.g. [Pirolli and Pitkow, 1999; Borges and Levene, 2000; Chierichetti et al., 2012]), but observations have been contradictory. This warrants further studies regarding the presence of memory and structure in human trails. According insights may change some basic assumptions of models of human trails on the Web.

Furthermore, if human trails on the Web indeed exhibit common patterns and regularities at least on some level, they might be leveraged for tasks that go beyond modeling aspects. If one can utilize patterns that are produced by humans simply interacting with the Web – e.g., navigating websites – one may have a new promising source for inferring knowledge. For example, a prominent task conducted on the Web is calculating semantic relatedness between concepts. Predominantly, this has been tackled by using Web content such as Wikipedia page text that has been produced by a small set of people. One can hypothesize that human trails on the Web may be leveraged in a similar way as also suggested in previous work [Chalmers et al., 1998; West et al., 2009]. If this holds, this argues for further studies into how humans make their consecutive decisions on the Web. Which hypotheses about how human trails on the Web are produced are more plausible than others? Gaining insights into these and similar questions can unseal new perspectives on human behavior on the Web which may be utilized in several ways.

1.3. Problem Statement, Objectives and General Approach

This section introduces the central problem statement of this thesis motivated by the challenges and opportunities that human trails on the Web implicate as discussed in Section 1.2.5. Also, I define the main objectives and general approach taken for tackling these problems. The following Section 1.4 presents the fine-grained research questions accordingly.

Problem statement. The production of human trails on the Web is omnipresent and has evoked a significant array of studies concerned with various aspects of human trails. Many different tools, models, insights and hypotheses have emerged. While some are well established, contradictory statements have been made regarding some aspects of human trails. An open question has been whether human trails on the Web exhibit memory effects as well as structural patterns which might be utilized beyond what previous work has done. Also, researchers have questioned what drives the production of human trails on the Web. However, it is difficult for our research community to make informed decisions about these aspects within coherent research approaches.

Objectives. To that end, this thesis aims at providing tools for facilitating future research concerned with the analysis of patterns, regularities and strategies in human trails on the Web. Of special interest is the detection of memory effects in human trails as well as gaining insights into how human produce human trails based on how they transition between states. Additionally, this thesis has the objective to investigate the usefulness of utilizing human trails on the Web for inferring knowledge. For generality, this thesis aims at studying several types of different human trails on the Web, across domains.

General approach. The main approach of this work is to model human trails on the Web with Markov chain models which allow to probabilistically model transitions between states of human trails. Specific configurations of model parameters allow to study different assumption and hypotheses about human trails. By utilizing statistical inference methods such as Bayesian inference, this thesis can make informed decisions about the plausibility of these.

1.4. Research Questions

This thesis aims at studying three sub-problems regarding the greater challenge of studying human trails on the Web. To that end, I aim at modeling the following aspects of human trails: (i) studying memory and structure in human trails by utilizing varying order Markov chain models,

(ii) leveraging human trails for the task of calculating semantic relatedness between concepts as well as (iii) expressing and comparing hypotheses about human trails that focus on beliefs about transitions. Next, the detailed research questions are introduced. Furthermore, a structural overview of each research question is provided in Table 1.1. This table highlights the relation of all articles of this thesis to the research questions described next. Additionally, it describes which (i) types of human trails, (ii) main topics, (iii) main contributions and (iv) methods each research question and article focuses on.

RQ1: What is the memory and structure in human trails on the Web?

Problem. For accurately modeling aspects of human trails on the Web, it is important to understand whether human behavior on the Web is memoryless or not and which structural patterns emerge. This is particularly essential to the Markov chain model – a prominent model for human trails on the Web [Pirolli and Pitkow, 1999]. Predominantly, the Markov chain model has been memoryless in a wide range of applications such as Google's PageRank [Brin and Page, 1998]. This means that the next state in a trail only depends on the current one and not on a sequence of preceding ones. The appropriateness of this memoryless property has been discussed in a series of works in the past (e.g., [Pirolli and Pitkow, 1999; Borges and Levene, 2000; Sen and Hansen, 2003; Gonçalves et al., 2009]). However, the statements about the appropriate Markov chain order have been quite contradictory. Yet, the dominant consensus has been that the memoryless model is an appropriate way for modeling human trails on the Web. Recently, an article by Chierichetti et al. [2012] has picked up on this question and has argued that Web users are not Markovian leading to the potential benefit of using higher order Markov chain models. However, such higher order Markov chain models have much higher complexity due to the exponentially rising number of parameters needed. Their better fit may be simply reasoned by overfitting. Thus, this warrants further investigations about the appropriate Markov chain order given human trails on the Web that specifically consider whether higher order models expose statistically significant improvements over lower order ones. As a consequence, our research community would benefit from a general framework that allows a comprehensive detection of the appropriate Markov chain model order based on statistical inference.

Approach. For tackling this research question, this thesis utilizes Markov chain models of varying order which are fitted to the data by utilizing (i) Bayesian and (ii) frequentist inference. The appropriate Markov chain order is evaluated by resorting to a series of statistical methods: (i) likelihood ratio test, (ii) Bayes factors, (iii) information-theoretic methods (AIC and BIC) as well as (iv) cross validation.

Findings and contributions. To that end, in Section 3.2, I present a *framework for detecting the appropriate Markov chain order given human trail data* as developed in [Singer et al., 2014c]. This framework is one of the main contributions of this thesis. It allows researchers to make informed decisions about the appropriate Markov chain order with the aid of a series of different approaches for evaluating Markov chain orders as described above. Based on the availability of this broad spectrum of evaluation methods as well by having the ability to study the consistency of their results, this framework allows for a comprehensive analysis of memory in human trails on the Web going beyond what previous work has done.

For demonstrating the general applicability and mechanics of the framework, presented work [Singer et al., 2014c] (Section 3.2) applies it to a series of human navigational trail datasets. By now having the feasibility to compare the results of several advanced statistical inference methods – each with its own advantages and disadvantages – this work demonstrates that by applying the framework one can make informed decision about the appropriate Markov chain order. However, as this thesis infers from theory and as previous work has suggested (e.g., [Pirolli and Pitkow, 1999]), all methods consistently highlight the difficulty of making statements about the appropriate Markov chain order having insufficient data but a large number of states. Yet, by reducing the state space by abstracting away from the page to a topical level, the results show that memory plays at least some role in human navigational trails on the Web. Additionally, this work showcases that the Markov chain framework can also be utilized for deriving common structural patterns in given trails.

Subsequently, this thesis studies memory and structure in human edit trails in collaborative ontology engineering patterns n Section 3.3 and in Section 3.4 based on joint work [Walk et al., 2014b,a]. These works further demonstrate the general applicability and features of the framework. The results indicate that the framework is capable of eliciting certain regularities, patterns and memory effects in such edit trails. Additionally, this thesis successfully confirms the importance of studying memory effects in human trails on the Web. By incorporating memory into Markov chain models applied to some types of human trails on the Web, we can improve the accuracy when predicting trails.

The presented applications of the framework argue that the results are dependent on several factors such as complexity and choice of data. However, the main benefit of presented framework is that it allows researchers to detect the appropriate Markov chain order given their specific data and problem setting by consulting several statistical inference methods implemented by the framework.

RQ2: Can we leverage human navigational trails for the task of calculating semantic relatedness between concepts?

Problem. This research questions aims at leveraging human navigational trails for calculating semantic relatedness. The calculation of semantic relatedness between concepts is an important step towards a semantically-enabled Web. It determines a score – usually between zero (not related) and one (synonymous) – that specifies how semantically related two concepts are to each other Predominantly, this and many other knowledge inferring tasks on the Web have been solved by using mostly human-generated content. While such content has been shown to be valuable, it only captures the semantics of a limited set of people who generated it. Thus, such data is restricted to humans who actively contribute to the Web, but neglects the massive number of humans simply interacting with the Web – e.g., lurkers [Nonnecke and Preece, 2000].

As an example, millions of humans navigate Wikipedia daily and produce human navigational trails. The findings of the first research question suggest that at least some structural patterns, regularities and strategies guide human navigational behavior on the Web. Thus, I hypothesize that we can also utilize these patterns for inferring knowledge. Tailored to this research question as well as motivated by insights of previous research [Chalmers et al., 1998; West et al., 2009], I study whether we can leverage human navigational trails for the task of calculating semantic relatedness between concepts.

Approach. For calculating semantic relatedness between concepts, this thesis resorts to the idea that the relative position between concepts in a trail influences their semantic relatedness. If many people navigate between two concepts, I hypothesize that they are semantically related in some way. Based on the observations of the first research question, I want to harness emerging structural patterns in human trails on the Web. For now calculating semantic relatedness between concepts based on this idea, this thesis utilizes a method based on co-occurrence information. Co-occurrence is calculated by counting how many times humans have navigated between two concepts and this information is used for weighing a vector-space model. Finally, for calculating semantic relatedness between two concepts, the approach calculates the cosine similarity between two vectors. In analogy to our Markov chain models of the first research question, we can think of these vectors as rows of a first-order Markov chain transition matrix (without accounting for the order of states). In order to investigate whether memory effects might be useful for this task, this thesis extends the calculation to also consider the co-occurrence between two concepts that are not adjacent in a given trail. This can be achieved by using sliding windows over trails.

Findings and contributions. In Section 3.5 , I study whether human navigational trails derived from the Wikigame can be leveraged for the task of calculating accurate semantic relatedness scores between concepts based on joint work with colleagues [Singer et al., 2013a]. Presented work uses the abovementioned co-occurrence approach and evaluates the results based on a series of gold-standard datasets such as the WordSimilarity-353 corpus. The results indicate that it is indeed possible to *leverage human* *navigational trails for the task of calculating semantic relatedness between concepts* which is the second main contribution of this thesis. Additional experiments on baseline corpora reveal that semantic relatedness calculated on this kind of human navigational trails can be more precise than semantic relatedness calculated on trails automatically extracted from Wikipedia's topological link network. However, not all trail corpora are equally useful and intelligent selection can be beneficial.

This work further highlights that certain patterns, regularities and strategies guide humans' consecutive behavior. This suggests that we can also utilize human trails on the Web for tasks that are usually solved by using Web content only and argues for existing and future methods to also consider them for deriving knowledge. While this work only focuses on human navigational trails and semantic relatedness, the ideas can be extended to study other kinds of human trails on the Web as well as to infer knowledge for other tasks.

RQ3: How can we compare hypotheses about human trails on the Web?

Problem. This research question tackles the issue of studying how human trails on the Web are produced. In detail, it questions how we can express and compare hypotheses about human trails on the Web. The study of the first two research questions has revealed that human behavior on the Web is at least partly guided by regularities, patterns and strategies. These studies and a series of previous works have identified cognitive strategies humans seem to apply while producing human trails on the Web. This leads to the development of hypotheses about human trails. In this thesis, I define such hypotheses as beliefs about transitions in human trails. For example, based on the insights of the second research question of this thesis, a belief could be that humans navigate the Web by choosing semantically related websites while a contrasting hypotheses could express that we believe in humans navigating randomly. It is crucial for our research community to make judgements about which hypotheses are more relevant than others for making informed decisions about models of human trails. Also, corresponding insights can unfold unexplored behavioral aspects that we also might successfully leverage or steer. Yet, expressing and comparing such hypotheses about human trails is difficult and is the main objective of this research question.

Approach. For expressing and comparing hypotheses with each other, this thesis fundamentally resorts to first-order Markov chain models and Bayesian inference. Hypotheses are intuitively expressed as adjacency matrices with values corresponding to believes in transitions. For example, if someone has the hypothesis that humans navigate by choosing semantically related websites consecutively, she could set the values of the matrix according to their semantic relatedness. Higher values refer to higher beliefs. The main idea of the approach is to incorporate these hypotheses expressed as matrices as informative Dirichlet priors into the Bayesian inference process. For doing so, it elicits Dirichlet priors from hypotheses by resorting to an adapted version of the so-called (trial) roulette method that sets the pseudo counts of the priors according to specified matrices. Finally, it utilizes the sensitivity of the Bayes factor on the prior for making informed decisions about the relative plausibility of hypotheses. If a hypotheses represents a valid hypotheses about behavior producing human trails on the Web according to given data, the evidence is higher compared to a uniform prior or an unlikely hypothesis.

Findings and contributions. To that end, in Section 3.6, I present *HypTrails*, an approach for expressing and comparing hypotheses about human trails, based on collaborative work [Singer et al., 2014b]. HypTrails implements the approach as described above and is the final main contribution of this thesis. For demonstrating the general applicability and mechanics, presented work applies HypTrails to different types of human trails: (i) business reviews on Yelp, (ii) tracks listened to on Last.fm, (iii) navigational trails over Wikipedia and (iv) synthetic trails produced according to know mechanisms.

1.5. Main Publications

This cumulative thesis consists of the following publications:

- **Article 1:** [Singer et al., 2014c] Singer, P., Helic, D., Taraghi, B., and Strohmaier, M. (2014c). Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PloS One*, 9(7):e102070

- **Article 2:** [Walk et al., 2014b] Walk, S., Singer, P., Strohmaier, M., Tudorache, T., Musen, M. A., and Noy, N. F. (2014b). Discovering beaten paths in collaborative ontology-engineering projects using markov chains. *Journal of Biomedical Informatics*, 51:254–271

- **Article 3:** [Walk et al., 2014a] Walk, S., Singer, P., and Strohmaier, M. (2014a). Sequential action patterns in collaborative ontology-engineering projects: A case-study in the biomedical domain. In *International Conference on Information and Knowledge Management*

- **Article 4:** [Singer et al., 2013a] Singer, P., Niebler, T., Strohmaier, M., and Hotho, A. (2013a). Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70

- **Article 5:** [Singer et al., 2014b] Singer, P., Helic, D., Hotho, A., and Strohmaier, M. (2014b). Hyptrails: A bayesian approach for comparing hypotheses about human trails. *arXiv:1411.2844 [cs.SI]*

A full list of co-authored articles that have been published during the course of my PhD studies can be found in Appendix A.

1.6. Contributions and Implications

Most of the contributions of this thesis are of methodological nature supplemented with empirical experiments. Overall, this thesis makes the following three main contributions:

- First, this work presents a *framework for evaluating the appropriate Markov chain order given human trail data* based on a series of advanced statistical inference methods each with their own advantages and disadvantages. By having this broad spectrum of different methods for evaluating varying order Markov chain models as well as being able to study the consistency of their results, this framework allows researchers to comprehensively gain insights into potential memory effects in human trails on the Web. Additionally, the framework and corresponding models can be utilized for detecting structural patterns as well as for predicting trails. The framework is made open-source and is available online[1]. By applying the framework to a series of empirical trail data stemming from distinct domains, this thesis demonstrates the mechanics of the framework as well as suggests that regularities, patterns and memory effects drive the production of human trails on the Web at least at some scale.
- Second, this thesis experimentally showcases the potential usefulness of harnessing human trails for tasks that are usually solved by utilizing content data. As an example, the thesis demonstrates that we can *successfully leverage human navigational trails for the task of calculating semantic relatedness between concepts*. The utilized method uses co-occurrence information between concepts in trails and an implementation is also made available online[2].
- Third, this thesis presents *HypTrails, an approach that allows researchers to express and compare hypotheses about human trails*. Hypotheses are defined as beliefs about transitions of human trails. This thesis demonstrates the general mechanics and applicability of HypTrails in a series of experiments on experimental and empirical human trail data. The approach is made open-source and is available online[3].

The methodological contributions of this thesis can provide helpful tools for researchers and practitioners interested in modeling aspects of human trails on the Web. The models and observations may not only help to identify

[1] https://github.com/psinger/PathTools/

[2] https://github.com/psinger/PathTools/

[3] https://github.com/psinger/HypTrails/

the behavior of humans in a platform, but may also be used for improving the work-flow, models of human trails, interface aspects or simply the user experience. Additionally, this thesis makes an arguments for expanding the existing arsenal of data sources by also considering human trails on the Web. By and large, this thesis provides a further stepping stone for the larger challenge of modeling human trails and gaining a better understanding of human trails and how they are produced.

1.7. Structure of this Thesis

The rest of this thesis is structured as follows. I start with a discussion of related work in Chapter 2. In this chapter, I first focus on methodological concepts utilized throughout this thesis in Section 2.1. Next, I discuss empirical studies and theories of human trails on the Web relevant for this thesis in Section 2.2.

The following Chapter 3 represents the main body of this cumulative thesis by presenting the main publications as mentioned in Section 1.5 for answering the research questions elaborated in Section 1.4. To give a better overview over the content and aspects studied in this thesis as presented in Chapter 3, I provide a structural overview in Table 1.1. In this table, I describe the relation of each section and article of Chapter 3 to the research questions at interest. Furthermore, I highlight which (i) types of human trails, (ii) main topics, (iii) main contributions and (iv) methods utilized in corresponding sections and publications.

Finally, Chapter 4 concludes the work by summarizing the research results and contributions in Section 4.1 as well as implications in Section 4.2. Limitations and future work are shortly discussed in Section 4.3. Apart from the publications presented in this cumulative thesis, I have co-authored a list of further publications as listed in Appendix A.

Table 1.1.: **Structural overview over content and aspects of this thesis.** This table provides a structural overview of the main content of this thesis by describing the relation of each section and article to the research questions at interest. Furthermore, the table highlights the corresponding (i) types of human trails, (ii) main topics, (iii) main contributions and (iv) utilized methods for each section (article).

Section (Article)	**RQ**	**Human Trails**	**Topics**	**Main contribution**	**Main methods applied**
Section 3.2 (Article 1)	RQ 1	navigational trails	memory, structure	A framework for the detection of the appropriate Markov chain order (memory effects) given human trails	MC modeling, frequentist statistics, Bayesian statistics, information-theoretic statistics, cross validation
Section 3.3 (Article 2)	RQ 1	edit trails in ontology projects	structure	Demonstration of the MC framework for pattern detection	MC modeling, frequentist statistics
Section 3.4 (Article 3)	RQ 1	edit trails in ontology projects	structure, memory, prediction	Demonstration of the MC framework for prediction	MC modeling, Bayesian statistics, cross validation, pattern mining
Section 3.5 (Article 4)	RQ 2	navigational trails, synthetic trails	semantic relatedness	Demonstration of the usefulness of leveraging human navigational trails for calculating semantic relatedness	co-occurrence, vector space model, cosine similarity
Section 3.6 (Article 5)	RQ 3	navigational trails, review trails, listening trails, synthetic trails	hypotheses about human trails	HypTrails approach for comparing hypotheses about human trails	MC modeling, Bayesian statistics, (trial) roulette method

2. Related Work

This chapter is structured by first discussing the methodological concepts utilized in this thesis in Section 2.1 before it presents empirical studies and theories of human trails on the Web in Section 2.2. I have the main intention of giving a broad overview of the research areas relevant for this thesis. For a higher level of detail, please consult the corresponding related work sections of the papers part of this cumulative thesis.

2.1. Methodologies

This section aims at introducing and discussing the main methodological concepts utilized in this thesis. First, I present the Markov chain model in Section 2.1.1 which is the main model applied in this thesis and which represents an intuitive way for modeling human trails on the Web. The Markov chain model is of specific importance for the first and third research question of this thesis. After that, I focus on statistical inference and model comparison methods that play a crucial role to this thesis (first and third research question) in Section 2.1.2. I focus on parameter inference and model comparison for Markov chain models, but want to emphasize that the developed concepts can be applied to other models and problem settings. Finally, I discuss methods for calculating semantic relatedness between concepts in Section 2.1.3 as this this elemental to the second research question of this thesis.

2.1.1. Markov Chain Modeling

Introduction and definition. Markov chain models have a long history and high importance for a wide range of scientific fields. A Markov chain model represents a stochastic system that models transitions between states from a given state space S. A Markov chain can be seen as a discrete-valued Markov process. A Markov process itself can be defined as a stochastic process that adheres to the Markovian property described later in this section. In this thesis, let us focus on discrete-time Markov chain models which correspond to a process having a discrete set of times. Also, let us only consider finite state spaces $S = \{s_1, s_2, ..., s_m\}$ with $m = |S|$; e.g., the state space could include all distinct Wikipedia pages navigated over. Transitions between states are expressed via probabilistic values that give the probability of transitioning from one state to another. The eponymous Markov chain models are named after Andrey A. Markov who was the first who theoretically introduced Markov chains in 1906 [Markov, 1906]. As an application, Markov focused on applying the method to poetries where he analyzed patterns of vowels and consonants. He presented the results in 1913 [Markov, 2006] doing all calculations by hand [Hayes et al., 2013]. In the following years, researchers such as Kolmogoroff [1936] picked up on the ideas by Markov and the model has been established as a fundamental method still having high impacts nowadays (e.g., Google's PageRank [Brin and Page, 1998]). I will present a small excerpt of related works applying Markov chain models after introducing their mechanics.

Usually, a Markov chain process is defined as memoryless. This means, that the next state is only dependent on the current one and not on a sequence of preceding ones. This is also known as the *Markovian property* and the resulting model is a first-order Markov chain model. For a sequence of random variables $X_1, X_2, ..., X_t$ we can define it as follows:

$$\begin{aligned} P(X_{t+1} = s_j | X_1 = s_{i_1}, ..., X_{t-1} = s_{i_{t-1}}, X_t = s_{i_t}) = \\ P(X_{t+1} = s_j | X_t = s_{i_t}) = p_{i,j}. \end{aligned} \tag{2.1}$$

We can represent a Markov chain model by a stochastic transition matrix P that has elements $p_{i,j}$. These elements describe the probability of transitioning from state s_i to s_j. As this transition matrix is stochastic, the probabilities of each row i sum to 1.

In this thesis, I am also interested in studying memory effects in human trails. Hence, it is of interest to extend Markov chain models to also incorporate them. In such higher-order Markov chain models, the next state not only depends on the current one, but on a series of preceding ones. Let us denote the order of a model as k – i.e., a chain with memory k. Then, we can formally write:

$$\begin{aligned} P(X_{t+1} = s_j | X_1 = s_{i_1}, ..., X_{t-1} = s_{i_{t-1}}, X_t = s_{i_t}) = \\ P(X_{t+1} = s_j | X_{t-k+1} = s_{i_{t-k+1}}, ..., X_t = s_{i_t}). \end{aligned} \tag{2.2}$$

For tractability, this thesis always converts higher-order Markov chain models to first-order models. This can be done in straight-forward manner by using compound states [Chierichetti et al., 2012]. Hence, the new state space includes all sequences of length k which results in a state space of size $m^k m$. For example, for a second-order Markov chain model, the state space would look like $S = \{(s_1, s_1), (s_1, s_2), (s_1, s_3), ..., (s_{m-1}, s_m), (s_m, s_m)\}$. Note that higher order Markov chain models are always better fits to the data compared to lower order models by definition as lower order models are nested within higher order models. However, higher order Markov chain models need more parameters than lower order ones due to the need of using compound states. Hence, their better fit may be due to simple overfitting which is why one needs to gauge whether their improvements are statistically significant. This is fundamental to this thesis and one of the main tasks as tackled by the first research question. In Section 2.1.2, I discuss several methods that can be utilized for this task.

Applications. Apart from the application of Markov chain models for modeling human trails on the Web as I will discuss in detail in Section 2.2.4, they have been historically applied to a series of problem settings as I shortly want to discuss next. One of the most classic application of Markov chain models is for modeling weather data. At universities around the world, a Markov chain model is frequently introduced by stochastically modeling states that refer to weather conditions. This has also been done in a more rigorous statistical context. For example, a prominent study is by Gabriel and Neumann [1962] who used Markov chain models for modeling rainfall data in Tel Aviv.

Markov chain models are applied in various scientific domains. Amongst many others, I want to highlight a few examples next. In [Nix and Vose, 1992], Markov chain models are used for modeling genetic algorithms. Kurzweil [2012] leveraged them for converting speech to text. Furthermore, Markov chain models can be utilized for statistical software testing as discussed by Whittaker et al. [1994]. They have also found their place in methods applied in economics and finance such as the work by Kijima and Komoribayashi [1998] demonstrates by studying the process of valuing credit risk derivatives.

Extensions. Markov chains and corresponding processes with the Markovian property do not only play a role for the Markov chain models explained in this chapter and utilized in this thesis, but are also relevant for a series of extension. First of all, I want to mention the *variable order Markov chain model* which extends higher order Markov chain models to have different orders based on the context [Bühlmann et al., 1999]. This means that some states are independent from the future states. By applying variable order Markov chain models, one can reduce the complexity of higher order Markov chain models, by still retaining some of its benefits. A further very prominent method is the so-called *Markov chain Monte Carlo (MCMC)* which refers to a method for sampling from a probability distribution. A thorough introduction is given by Gilks [2005]. This is achieved by constructing a Markov chain that has a stationary distribution in the form of the desired distribution. By successively sampling from this Markov chain, it is possible to achieve a sample from the desired distribution. Markov chain Monte Carlo methods have gained popularity in the past few years for the evaluation of posterior distributions in Bayesian models. As mentioned later in Section 2.1.2, the marginal likelihoods frequently can not be integrated analytically; then MCMC methods come in handy [Kass and Raftery, 1995].

Another well-known model is the *hidden Markov model (HMM)* which refers to a Markov process with states that are unobserved and hidden. For an introduction, see the work by Rabiner and Juang [1986] and Rabiner

[1989]. This is in contrast to the classic Markov chain model utilized in this thesis where states can be observed directly and hence, transition parameters need to be determined only. Hidden Markov models have been applied in a wide range of applications such as speech recognition [Rabiner, 1989] or bioinformatics [Karplus et al., 1998]. A further alternative to the classic Markov chain model is the *Markov decision process* which is applied when states can be fully observed, but state transitions additionally are related to an action vector. Markov decision processes can be dated back to Bellman [1957].

As a final extension, I want to mention *Markov random fields* also called *Markov networks*. In a Markov random field, a state not only depends on the previous state in time as for the classic Markov chain model, but rather on the neighbors as described by a graph. Instead of *Bayesian networks* which are directed, Markov random fields are undirected representations. Markov random fields are applied in domains like computer vision [Li, 1995] or text processing [Metzler and Croft, 2005]. An introduction to Markov random fields and a further elaboration of applications can be found in the work by Kindermann et al. [1980].

2.1.2. Statistical inference and model comparison

When fitting models such as the Markov chain model of interest in this thesis, statistical inference for determining the parameters of the models is necessary. Furthermore, we need proper statistical methods for comparing models with each other which I frequently tackle throughout this thesis. For providing a broad spectrum of methods, I resort to the two main statistical schools: (i) *Frequentist statistics* and (ii) *Bayesian statistics*. Both offer diverse methods for statistical inference and model comparison. For model comparison, I supplement these methods with (iii) *information-theoretic* and (iv) *cross validation* approaches. By providing such a broad spectrum of different statistical methods, I can get a thorough view on the appropriateness of competing models and also counteract some potential disadvantages of methods. All approaches resort at least partly to the *likelihood function* which I describe next. Then, I give introductions to the distinct statistical methods as well as some historic background and their advantages and disadvantages. For a thorough historic recap of statistics please also refer to the excellent work by Stigler [2002]. I lay focus on Bayesian inference as it is utilized in several contexts of this thesis. Also, all elaborations are tailored towards Markov chain models. However, the ideas and concepts can be applied to other models and settings at interest. A more thorough discussion about these methods can be found in one of the papers of this cumulative thesis [Singer et al., 2014c] (Section 3.2) as well as in [Singer et al., 2014b] (Section 3.6).

Likelihood function. The term *likelihood* and corresponding *likelihood function* was coined by R.A. Fisher in the 1920's [Fisher, 1922]. The likelihood function represents the probability of observing the data given a model with specific parameter configurations. It represents the fundament for many statistical methods today and is used by statisticians of all schools. It also is elementary for statistical inference. For tractability, the natural logarithm is mostly used for calculating the likelihood function, which we can call the log-likelihood. For Markov chain models having the Markovian property as defined in Equation 2.1, we can define the likelihood function as:

$$P(D|\theta) = \prod_i \prod_j p_{i,j}^{n_{i,j}} \tag{2.3}$$

where $p_{i,j}$ correspond to the probabilities of transitions of transition matrix P and $n_{i,j}$ is the number of transitions observed in data D from state s_i to state s_j.

Frequentist statistics. Frequentist statistics are usually associated with concepts by Fisher, Neyman and Pearson. While the adjective *frequentist* is commonly used nowadays, it was uncommon in the early days of this statistical field in the 1920's [Fienberg et al., 2006]. Frequentist inference can be characterized by the motion of drawing conclusions by looking at the frequency of data. Usually, parameters of models are seen as being fixed, but unknown.

A popular method for determining these parameters is the *maximum likelihood estimate (MLE)* which is the estimation of parameters that maximize the likelihood function (see [Royall, 1997] for a detailed introduction to MLE). For Markov chains, the MLE for the parameters ($p_{i,j}$) is simply the number of transitions between two states s_i and s_j expressed as $n_{i,j}$ divided by the total number of transitions from one state s_i to all other states:

$$p_{i,j} = \frac{n_{i,j}}{\sum_j n_{i,j}} \tag{2.4}$$

Hence, we use the frequency of data for making decisions about the parameters of the models. Due to abovementioned higher complexity of higher order Markov chain models paired with potential overfitting [Murphy, 2002], looking at likelihoods of varying order models is not enough for gauging their appropriateness. For properly comparing the models, we can resort to statistical hypothesis testing which is a key technique of frequentist statistics and which was coined by Fisher [1925]. The main idea is to compare a *null* and *alternative* hypothesis with each other by calculating the evidence against the null hypothesis. For doing so, one has to choose a proper test statistic and calculate the p-value which gives insights into the statistical significance of the result [Goodman, 1999]. We can reject the null hypothesis if the p-value is below a given significance level.

One commonly used test statistic of the frequentist community is the *likelihood ratio test* [Neyman and Pearson, 1992]. For comparing Markov chain models, we can also utilize these likelihood ratio tests as discussed by Tong [1975]. The test is suited for comparing the fit of two competing models who are nested. The final ratio tells whether one model is more likely than the other one. For calculating the statistical significance of this ratio, a χ^2 test is utilized with a degree of freedom equal to the difference of the number of parameters of both models [Bartlett, 1951].

While frequentist statistics and corresponding hypothesis tests utilizing p-values are widely-used, they also have been criticized in the past (e.g., see [Cohen, 1994; Loftus, 1996; Goodman, 2008; Nuzzo, 2014; Morrison and Henkel, 2006]). Amongst others, p-values can measure whether a result happens by chance, but not what the odds are that a hypothesis is true as pointed out by Nuzzo [2014]. The author also stated that p-values can not make statements about the underlying reality and that they cannot make a statement about how much of an effect can be observed. It may also happen that a small p-value is the result of small datasets. Nonetheless, studies have shown that likelihood ratio tests represent a very understandable way of specifying statistical significance between model fits [Perneger and Courvoisier, 2010] which is also coupled with the clear statement the resulting ratio tells us. We need to note though that the likelihood ratio test only works with nested models. While this is the case for our nested Markov chain order models, it may be problematic for many other scenarios.

Bayesian statistics. Bayesian statistics can be traced back to the ideas and concepts of Reverent Thomas Bayes who first talked about it in 1764. The famous *Bayes' theorem* itself was introduced later by Pierre-Simon Laplace [Stigler, 2002]. Bayesian inference refers to the statistical method that aims at determining the parameters of a model by utilizing the *Bayes' rule* defined as follows:

$$\overbrace{P(\theta|D,M)}^{\text{posterior}} = \frac{\overbrace{P(D|\theta,M)}^{\text{likelihood}}\overbrace{P(\theta|M)}^{\text{prior}}}{\underbrace{P(D|M)}_{\text{marginal likelihood}}} \tag{2.5}$$

θ corresponds to the parameters (transition probabilities) we want to determine, D to the underlying data and M to a specific model (e.g., first-order Markov chain model) at interest. The *likelihood* $P(D|\theta, M)$ refers to the probability of the parameters given data and model. $P(\theta|M)$ is the so-called *prior probability* which refers to our prior estimates of the parameters. $P(D|M)$ is called *marginal likelihood* or *evidence* and plays an important rule for comparing models. The final $P(\theta|D, M)$ is the *posterior* to be determined.

For a detailed introduction to Bayesian inference please refer to [Box and Tiao, 2011]. I now shortly introduce the corresponding aspects of Bayesian inference relevant for Markov chain modeling. For further details, please refer to [Strelioff et al., 2007] and to one of the publications of this thesis [Singer et al., 2014c] (Section 3.2).

As earlier, the *likelihood function* for Markov chain modeling is defined as in Equation 2.3. The *prior* reflects our belief in the parameters of a model, before we see the data. In Markov chain models, each row of the transition matrix represents a categorical distribution for which the *Dirichlet distribution* is the *conjugate prior*. The advantage of conjugate priors is that by using them, the posterior distribution is from the same distribution family as the prior. For each row i of the transition matrix P of a Markov chain model, we have a prior Dirichlet distribution $Dir(\boldsymbol{\alpha})$ with $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_m]$. The hyperparameters $\alpha_{i,j}$ correspond to the mentioned prior belief in the parameters and can be seen as pseudo counts. Hence, the posterior distribution represents a combination of our prior belief in transitions $\alpha_{i,j}$ and the actual transitions $n_{i,j}$ we observe. For a more thorough introduction into conjugate priors and the Dirichlet distribution, please refer to Huelsenbeck and Andolfatto [2007].

The *marginal likelihood* or *evidence* as used in Equation 2.5 expresses the probability of the data D according to a model M. It is specifically utilized for comparing models with each other. For Markov chain inference, we can define it as follows (derivation provided in Section 3.2 and in [Singer et al., 2014c; Strelioff et al., 2007]):

$$P(D|M) = \prod_i \frac{\Gamma(\sum_j \alpha_{i,j})}{\prod_j \Gamma(\alpha_{i,j})} \frac{\prod_j \Gamma(n_{i,j} + \alpha_{i,j})}{\Gamma(\sum_j (n_{i,j} + \alpha_{i,j}))} \tag{2.6}$$

For comparing models with each other, Bayesians usually resort to *Bayes factors* which represent a Bayesian alternative to the hypothesis testing methods of frequentist statistics. Kass and Raftery [1995] describe Bayes factors in detail. By again using Bayes' theorem, we can determine the posterior probability of a model H given the data D:

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \tag{2.7}$$

The likelihood $P(D|M)$ is the marginal likelihood (evidence) defined in Equation 2.6, $P(D)$ is the probability of data regardless the model and $P(M)$ is the prior probability of a model M. For a fair and unbiased comparison researchers usually assume that all models are equally likely a priori by using a uniform prior over the models. In such a case, we can define the Bayes factor for comparing two models M_1 and M_2 given observed data D as:

$$B_{1,2} = \frac{P(D|M_1)}{P(D|M_2)} \tag{2.8}$$

Throughout this thesis, I resort to such Bayes factors in several occasions. For example, I use them for comparing Markov chain models of different orders or for comparing hypotheses about human trails. One disadvantage of this approach is that it is often very difficult to calculate Bayes factor as the necessary integrals might not be able to be solved analytically. In such a case, practitioners often resort to alternatives that try to avoid this issue – e.g., asymptotic approximation or sampling from the posterior (MCMC, Gibbs) [Kass and Raftery, 1995]. In the case of Markov chain models, we do not have this problem as the conjugate Dirichlet priors can be integrated analytically.

A further thing to note is that one common critique of Bayes factors is their high sensitivity on the choice of the prior as originally pointed out by Kass and Raftery [1995]. Contrary, posterior measures are more ignorant regarding the influence of the prior the more data one observes and incorporates [Vanpaemel, 2010]. Actually, when the number of observations becomes very large, the Bayesian posterior tends to the maximum likelihood estimation. Nonetheless, literature has pointed out that the high influence of the prior on the marginal likelihood and Bayes factors should not only be seen as a limitation, but also holds opportunities. As emphasized by Vanpaemel [2010], if *”models are quantitatively instantiated theories, the prior can be used to capture theory and should therefore be considered as* *an integral part of the model"*. Hence, Bayes factors can also be used as an instrument for making informed decisions about the plausibility of scientific theories by incorporating them as priors into Bayesian inference. However, the process of eliciting informative priors from theories is no trivial task and requires careful steps as discussed in [Oakley, 2010; Garthwaite et al., 2005]. Wolf Vanpaemel tackles this issue in follow-up work [Vanpaemel and Lee, 2012; Vanpaemel, 2011].

In this thesis, I utilize this sensitivity for the task of comparing hypotheses about human trails as presented in Section 3.6. In detail, I incorporate such hypotheses as informative Dirichlet priors for Bayesian Markov chain inference. Then, I use marginal likelihoods and Bayes factors for comparing hypotheses with each other. I present an adaption of the so-called *(trial) roulette method* for the task of eliciting proper Dirichlet priors from expressed hypotheses. This method was first introduced in [Gore, 1987] and discussed in [Oakley, 2010; Davidson-Pilon, 2014]. The general concept is to distribute a given number of chips to a grid that represents bins of a distribution. In my case, I see the grid as a matrix representing beliefs about transitions in a Markov chain model. Chips are then automatically distributed for determining pseudo counts for the hyperparameters of Dirichlet priors.

Bayesian model selection has further advantages which might make it superior to frequentist approaches. Compared models do not need to be nested and the main benefit is the inclusion of a natural *Occam's razor* which describes a penalty for too much complexity. The Occam's razor is a principle that states that in case of a set of competing hypotheses, the one with the fewest assumptions (i.e., parameters) should be selected. This allows researchers to get intuitive statements about the relative appropriateness of a set of models and overcomes the issue of overfitting [Kass and Raftery, 1995; MacKay, 1992; Murray and Ghahramani, 2005; MacKay, 2003].

Further methods for comparing models. Throughout this thesis, I also consider *information-theoretic* methods for comparing models. These methods are principled on concepts and ideas stemming from information theory with a focus on entropy measures. The two most well-known methods for model selection using information-theoretic approaches are (i) the *Akaike information criterion (AIC)* and (ii) the *Bayesian information criterion (BIC)*. For a thorough introduction into information theory please consult the works by Burnham and Anderson [2002, 2004]. AIC and BIC are shortly discussed next.

AIC was introduced by Akaike [1973] and is based on the Kullback-Leibler divergence [Kullback and Leibler, 1951] and the asymptotic properties of the likelihood ratio statistics discussed above. The idea is to balance the goodness of fit with the number of parameters needed by subtracting the maximum of the likelihood of a model from the number of parameters needed. For a set of competing models the approach is to minimize the AIC [Gates and Tong, 1976] and for Markov chain model selection a concrete approach was introduced by Tong [1975].

BIC follows a similar idea to AIC and was first introduced by Schwarz [1978] which is why it is also often called *Schwarz criterion*. It represents an approximation to the Bayes factor introduced above. It penalized higher order models more compared to the AIC by adding an additional penalization for the number of observations [Katz, 1981].

Mostly, AIC and BIC suggest the same model when both are applied. Both approaches have been criticized and praised in the past [Burnham and Anderson, 2004; Weakliem, 1999]. Katz [1981] emphasized that by using AIC, one may end up with too high of an order if used for determining the appropriate Markov chain order and hence, the work suggests to prefer BIC over AIC for Markov chain model selection. However, BIC does not perform too well for small datasets [Csiszár and Shields, 2000] which is why one might prefer AIC for small sized data [Baigorri et al., 2009]. These methods do not require the models to be nested which is why they may be preferred over the likelihood ratio tests for some use cases. BIC can also be used to approximate Bayes factors which might specifically come handy if the factors can not be calculated analytically as described above.

One further common way to evaluate models or a set of models is to use *cross validation* which is specifically applied in the machine learning area. The idea is to fit the model on a portion of the data and evaluate it on the remaining portion. This can also be applied for Markov chain order selection as suggested by Chierichetti et al. [2012] and Murphy [2002]. One would use the fitted Markov chain model and predict the sequences of the test set. The model that performs the best can then be preferred; but, complexity needs to be considered.

2.1.3. Computation of semantic relatedness between concepts

Semantic relatedness is a metric for indicating how semantically related two terms or documents are to each other. We can distinguish semantic relatedness from other similar metrics like *semantic similarity* or *semantic distance* [Resnik, 1998; Pedersen et al., 2007]. According to Harispe et al. [2013], we can define *semantic relatedness* as the *"strength of the semantic interactions between two elements without restriction regarding the types of semantic links considered."* Note that while similar, semantic similarity can be seen as a specialization of semantic relatedness and only considers taxonomical relationships. Semantic distance specifies the reverse of semantic relatedness in order to determine a semantic distance metric between terms or documents. While slightly different in their definitions, literature often refers to these metrics interchangeable. I focus on the more general notion of semantic relatedness in this thesis. Amongst others, it includes: similarity, meronymy, hypernymy or IS-A relationships. Next, with respect to the scope of this thesis, I start with a general overview of works calculating semantic relatedness on the Web in general. For a discussion about calculating semantic relatedness by leveraging human trails, please refer to Section 2.2.3.

Semantic relatedness on the Web. Computing semantic relatedness between concepts of the Web has been studied heavily in the past and is a fundamental approach for enabling a semantically-enabled Web. In their seminal work, Rubenstein and Goodenough [1965] stated that a positive relationship between the degree of semantic relatedness and the degree of similarity of their contexts exists. Later, psychological experiments such as by Tversky [1977] or Medin et al. [1993] demonstrated that semantic relatedness is both asymmetric and context dependent. Asymmetry refers to the observation that people provide different degree of semantic relatedness between two concepts if their positions are changed. Context dependency means that the degree of semantic relatedness is influenced by the context the concepts appear in. However, Aguilar and Medin [1999] and Medin et al. [1993] argued that asymmetry only occurs occasionally.

A large array of applications build upon the calculation of semantic relatedness between concepts. To give some examples: word sense disambiguation [Resnik, 1998], usage for word spelling errors [Budanitsky and Hirst, 2001, 2006], text segmentation using lexical cohesion [Kozima, 1993; Manabu and Takeo, 1994], image [Smeulders et al., 2000] and document [Srihari et al., 2000] retrieval or cognitive science [Talmi and Moscovitch, 2004]. For a detailed survey on this topic see the work by Zhang et al. [2012]. The work by Li et al. [2003] emphasizes that for calculating semantic relatedness between concepts we can distinguish *edge-counting and information-theory based methods*. According to it, edge-counting methods utilize *IS-A* relations only, while as pointed out by Resnik [1998], information-theoretic methods use information content for calculating semantic relatedness between concepts. Precisely, if both concepts share more content, they are more related to each other. A prominent method that combines both approaches is called *Jiang-Conrath distance* as introduced by Jiang and Conrath [1997]. This method can be applied to tree-shaped lexical taxonomies. For calculating semantic relatedness between concepts, the method uses an edge-counting scheme which it enhances with a node-based approach as known from information content methods.

Over time, several information sources suitable for calculating semantic relatedness between concepts have emerged. A very prominent example is the lexical database WordNet [Miller, 1995] which has been heavily studied in the past (e.g., see [Budanitsky and Hirst, 2001; Patwardhan, 2006; Banerjee and Pedersen, 2003; Pedersen et al., 2004; Navigli and Ponzetto, 2012]). Budanitsky and Hirst [2006] compared five distinct methods for calculating semantic relatedness between concepts using WordNet and concluded that the Jiang-Conrath distance works best.

Today, human-generated content is produced at massive scale on the Web. Hence, approaches have been developed that aim at leveraging such content for calculating semantic relatedness between concepts. For example, human-generated content generated in tagging systems (e.g., [Strohmaier et al., 2012] or [Helic et al., 2011]) has been shown to be a good content source for this task. But also content created on Wikipedia can be successfully leveraged (see e.g., [Gabrilovich and Markovitch, 2007]). Next, I will briefly cover some methods applied on Wikipedia as in this thesis also navigational trails through Wikipedia are leveraged for the task of calculating semantic relatedness between concepts. On Wikipedia, we can roughly distinguish between *content and link based semantic relatedness methods*. The former try to leverage the human-generated content while the latter mainly focus on links between concepts.

Semantic relatedness on Wikipedia. Maybe the most prominent content based method is the so-called *Explicit Semantic Analysis (ESA)* method by Gabrilovich and Markovitch [2007]. The main idea is to calculate a tfidf-weighted inverted index which can be used for calculating semantic relatedness between concepts by e.g., utilizing cosine similarity. The advantage of ESA is that it is not limited to word relatedness, but can also be used on arbitrary text. A further well-known method is *Latent Semantic Analysis (LSA)* [Landauer et al., 1998; Deerwester et al., 1990] which uses singular value decomposition on word count matrices of textual articles and then proceeds to determine similarity by calculating the angle between vectors. While both ESA and LSA work well on Wikipedia, they can be calculated on any other textual corpus.

Link-based methods can be partitioned into methods that focus on hyperlink information and methods that exploit trails through the underlying topological link network of Wikipedia. For example, Ito et al. [2008] used co-occurrence information between links that are present on a single page for deriving semantic relatedness between concepts; similar approaches have been proposed by Milne [2008] and Turdakov and Velikhov [2008]. Methods that are directly applied on the underlying link network have e.g., been suggested by Yeh et al. [2009] who presented an algorithm called WikiWalk conducting random walks through the network and then using these walks for specifying relatedness scores between concepts. Strube and Ponzetto [2006] investigated direct path-based measures and also studied a combination with WordNet for this approach.

2.2. Empirical Studies and Theories of Human Trails on the Web

This section gives a compressed overview of related work that studies human trails on the Web with a specific focus on emerging theories as well as on studies about human navigational trails on the Web as they are the main type of trails studied in this thesis. However, I also discuss studies on other types of human trails on the Web as well as on trails outside the Web realm for giving insights into the bigger picture. I highlight the corresponding main theory or observation at the beginning of each paragraph as communicated by related work.

2.2.1. Regularities and patterns

In this section, I am mainly concerned with discussing work that has studied regularities and patterns in human trails on the Web. This has been of interest for our research community for nearly two decades.

Human navigational trails on the Web exhibit regularities and patterns. One of the first studies on this topic is by Catledge and Pitkow [1995] who investigated actual human behavior as captured from client-side log files of NCSA's XMosaic. The experiments identified a series of navigation patterns such as serendipitous browsing. The article argues for the usefulness of the identification of navigational patterns for design and usability improvements for pages, sites or browsers. This steered further investigations in the late 1990s making the analysis of navigational human behavior on the Web a prominent research field. Subsequent seminal work by Huberman et al. [1998] studied regularities in World Wide Web surfing. The authors emphasized that navigating hyperlinks represents one of the most common modes of accessing information on the Web. By investigating various navigational log data such as a representative sample based on navigational trails by AOL WWW users, the authors found that surfing patterns on the Web reveal strong statistical regularities. Also, humans seem to follow hyperlinks as long as they find value in them and the probability distribution of the number of hits follows Zipf's law. In the same year, Huberman and Adamic [1998] demonstrated that recommendations play an important role in how humans choose websites to access. Thus, the article argues that social search mechanisms are manifested in the statistics of visits of websites on the Web. Actually, it seems to lead to a universal power law for the number of site visits where the exponent corresponds to the rate of new sites that humans discover.

Patterns and regularities can also be found in other types of human trails on the Web. Similar to the found regularities in human navigational trails, other types of human trails also exhibit certain regularities. Wang and Huberman [2012] studied randomness on two human trail datasets. The first one captures successive comments by humans on the who-trust-who consumer review site Epinions. The second dataset considers human trails from the location based social network Whrrl. Each trail consists of consecutive check-ins to places like restaurants, hotels or bookstores by users of the website. The authors looked at predictability of individual activities by calculating both entropy and mutual information on the trails at hand. Their results indicate the presence of regularities in these trails that can be used for prediction. However, it seems that the predictability is higher when humans act alone compared to group activities.

Structural patterns have also shown to play an important role in various other kinds of human trails on the Web. In [Archak et al., 2010], the authors studied trails of ads seen and clicked by humans. The authors proposed an approach called *adgraphs* which can formulate graphs for representing co-occurrences of events in given trails. Generated graphs can capture structural properties of human trails. The article introduces several scoring rules which are called *adfactors*. They are able to interpret the global role of ads in a graph. The approach can help practitioners to find and understand correlations in trails of ads seen and corresponding human actions. Yang et al. [2014] presented a model for finding progression stages in time-evolving trails. The method allows accurate prediction of future events and the determination of progression stages. These stages can be grouped based on similar patterns. This can not only give one the ability to model the data, but also to find behavioral aspects within such trails. Similar to human navigational trails, search trails capture subsequent search activities of humans. Trails begin with a search query and end with either another query, inactivity or termination of the browser [White and Drucker, 2007]. For example, Bilenko and White [2008] and White and Huang [2010] studied the value of search trail following. In this thesis, I am also interested in studying human edit trails in collaborative ontology engineering projects. Pöschko et al. [2012] and Walk et al. [2013] developed *PragmatiX* which is a tool for visualizing and analyzing aspects of the history of collaborative ontology engineering projects. Change logs of such projects have also been studied by Falconer et al. [2011], Strohmaier et al. [2013] and Wang et al. [2013].

Regularities in human trails on the Web might be based on inherent regularities of human behavior. Song et al. [2010] were interested in studying the degree of predictability of human behavior. The authors focused on studying human mobility trails collected from mobile phone carriers. By measuring entropy of given trails, the results indicate high predictability in human mobility. Additionally, only slight variability in predictability could be identified. This form of determining predictability of trails has its origins in studies on gene expressions such as the work of Steuer et al. [2002] discusses. The article suggests that such observations might be reasoned by inherent regularities of human behavior in general. Hence, this confirms the abovementioned studies that found regularities and patterns in all kinds of human trails on the Web and argues for the necessity of a better understanding of the production of these trails.

2.2.2. Behavioral strategies of humans

Based on these identified regularities and patterns, an array of work exists that has focused on studying the behavioral aspects of human navigational trails and specifically, emerging strategies.

Human trail behavior is guided by information scent. One of the most well-known theories about human navigation behavior is the so-called *information foraging theory* [Pirolli and Card, 1999]. It postulates that human behavior in an information environment on the Web is guided by information scent which is based on the cost and value of information with respect to the navigational goals that humans have in mind [Chi et al., 2001]. Consequently, humans estimate the value of information they gain on a given trail. After they have gathered enough information, they evaluate it by comparing it to what they expected and as soon as the information scent decreases, they switch to a different information source. The authors also presented two methods for modeling human needs which are based on the concept of information scent. Like many others, Chi et al. [2001] also highlighted that a better understanding of human navigational patterns and corresponding human behavior can have implications for a series of applications such as for personalizing Web environments, improving website design or also identifying parts of websites with bad design. Shortly after the establishment of the information foraging theory, Olston and Chi [2003] introduced a novel approach called *ScentTrails*. The approach highlights hyperlinks on websites in order to indicate trails to utilize for searching results using the concept of information scent. Furthermore, in [Teevan et al., 2004], the authors emphasized that indeed humans seem to prefer to navigate websites by leveraging their contextual knowledge instead of only using keyword based search. Downey et al. [2008] found that navigation is specifically useful when information needs are rare and that given a general result, navigating to more specific websites can be a fruitful way of satisfying these needs.

Humans follow certain strategies while navigating and produced trails differ from shortest paths. In [West and Leskovec, 2012a,b; West et al., 2009; Scaria et al., 2014], the authors studied human navigational trails derived from the online game *Wikispeedia*[1]. In Wikispeedia, players receive an arbitrary start and target Wikipedia page. The goal is to reach the target page by only clicking on hyperlinks of Wikipedia pages and navigating Wikipedia. The work by West and Leskovec [2012a] argues

[1] http://cs.mcgill.ca/~rwest/wikispeedia/

that no sophisticated background knowledge is necessary for efficiently navigating Wikipedia. Nonetheless, as pointed out in [West and Leskovec, 2012b], produced trails differ from shortest paths in several ways. Humans seem to prefer to navigate through high-degree hubs in the beginning of their navigational steps before their navigational behavior is guided by content features; this confirms the observations by Downey et al. [2008]. West and Leskovec [2012b] also demonstrated that complex solutions are rarer but more effective compared to simpler ones. Scaria et al. [2014] found that backtracking plays an important rule for humans navigating Wikipedia. They mostly use it for going back to high-degree hubs. Based on this analysis, the authors also introduced a model that predicts whether a human abandons or finishes a corresponding game in Wikispeedia.

Humans navigate over semantically similar concepts. While navigating Wikipedia and the Web, semantic relatedness between concepts seems to play a crucial role [West et al., 2009]. West and Leskovec [2012b] found similar behavior by demonstrating that the closer a human is to the final target they want to reach, the more similar the navigated over concepts are to the target. In similar context, Pierce et al. [1992] demonstrated that semantic relatedness and omission probability affect menu selection performance. The importance of semantic relatedness between consecutive items that humans navigate on the Web is also emphasized by the work of Chalmers et al. [1998].

2.2.3. Leveraging human trails

In this section, I want to shortly discuss the potential usefulness of human trails on the Web for inferring knowledge; specifically, for calculating semantic relatedness between concepts.

Human trails on the Web might be leveraged for calculating semantic relatedness between concepts. The methods for calculating semantic relatedness between concepts as described in Section 2.1.3 have focused on using Web content only. However, as suggested by earlier work [Chalmers et al., 1998] – or by also keeping the ideas by Bush [1945] in mind – it might be also reasonable to look into actual human behavior patterns (in this case navigational trails) for calculating semantic relatedness between concepts. In fact, Chalmers et al. [1998] studied trails over URLs requested by a hand full of humans. The authors used a co-occurrence method for recommending URLs and for visualizing trail components. The task of determining semantic relatedness between concepts using navigational trails by humans has also been investigated by West et al. [2009]. Their work studies human navigational trails derived from Wikispeedia. The authors have introduced a method for deriving semantic relatedness of concepts by looking at how humans navigate between them.

Both Chalmers et al. [1998] and West et al. [2009] have provided first insights into the potential usefulness of leveraging human navigational trails for the task of calculating semantic relatedness between concepts. However, their works are limited in some ways. Chalmers et al. [1998] only studied trails from a hand full of humans for recommendation and visualization purposes. Their work does not focus on the aspect of semantic relatedness between concepts and it is an open question how accurately this can be achieved as no evaluation for this task has been provided. While West et al. [2009] have focused on explicitly calculating semantic relatedness scores between concepts, their method only allows to calculate relatedness between concepts that at least once co-occur in a trail. Also, their work does not evaluate the results based on given gold-standards. The work presented in this thesis (see Section 3.5) provides an evolution and extension of the work by Chalmers et al. [1998] and West et al. [2009]. It applies a robust method that allows to calculate semantic relatedness between any arbitrary concept at interest. Additionally, the method is applied to large-scale data and rigorous evaluation based on several baseline corpora and gold-standards is provided.

By and large, in this thesis, I can indeed showcase the usefulness of using human navigational trails for the task of determining semantic relationships between concepts in Section 3.5. This confirms the early hypotheses by Chalmers et al. [1998] as well as the work by West et al. [2009]. Nonetheless, this thesis also highlights that not all trails are equally useful and selection strategies can improve results by a significant margin. While previous work as well as the article presented in this cumulative thesis have focused on calculating semantic relatedness between human navigational trails, it can be easily extended to other types of human trails which is an interesting aspect to investigate in future.

2.2.4. Modeling human trails on the Web with Markov chain models

In this section, I discuss related work that has applied the Markov chain model (see Section 2.1.1) to human trails on the Web, again I focus on human navigational trails.

Human navigational trails can be well modeled stochastically and with Markov chain models. In the early days of studying human navigational trails on the Web, several studies proposed that human navigation on the Web can best be modeled stochastically, but in a memoryless way [Huberman et al., 1998; Cunha and Jaccoud, 1997; Padmanabhan and Mogul, 1996]. This means that a state in the model is only dependent on the current one, and not on a series of preceding ones. With these observations and assumptions, the Markov chain model is a logic choice for efficiently modeling human navigation on the Web [Pirolli and Pitkow, 1999]. It is also the main model utilized in this thesis; for a thorough introduction please refer to Section 2.1.1.

The most prominent application utilizing a Markov chain model is Google's PageRank [Brin and Page, 1998]. This algorithm is responsible for ranking websites on the Web according their structure. The PageRank uses a random surfer model which postulates that such a surfer gets bored after a certain amount of clicks and switches to a completely random page (damping factor). The actual page values of the PageRank indicate the probability of a random surfer to end up at it. Actually, this can be understood as a memoryless (first-order) Markov chain model having a transition matrix with values corresponding to transitions of the PageRank.

First-order Markov chain models have also been applied in several other applications and studies. For example, Bestavros [1995] used them for a prefetching service with the intention to reduce server load. In [Sarukkai, 2000], the utility of (first-order) Markov chain models for modeling navigational trails on the Web is emphasized. It is stated that such models can not only be useful for modeling navigational click trails, but also for tasks like tour generation or hub / authority identification. The work further praises the generality and power of Markov chains as a tool for heuristically modeling Web trails. Further examples are the work by Zukerman et al. [1999] who leveraged first-order Markov chain models for predicting human navigation on the Web or Nicholson et al. [1998] who used these models for predicting which document humans request next. A mixture of first-order Markov chain models was also utilized by Cadez et al. [2003] for clustering and visualizing navigation patterns on a Web site.

The first-order Markov chain model is a practical model for human navigational trails. While, as above examples indicate and claim, memoryless Markov chain models are well-performing for modeling human navigation on the Web, several studies have questioned the memoryless property [Pirolli and Pitkow, 1999; Borges and Levene, 2000]. However, they have emphasized that the first-order indeed is a reasonable way to model human navigation on the Web. It was argued that while increasing the order of a Markov chain leads to a reduction in uncertainty, the higher complexity of such models might not compensate the additional benefit [Pirolli and Pitkow, 1999; Borges and Levene, 2000]. Similar observations were also presented by Sen and Hansen [2003] who studied the applicability of first-order and second-order Markov chain models of navigational log data within a single platform. Their results indicate that while a second-order model works reasonable well, the number of parameters necessary is enormous which is why the authors suggest to use finite mixtures of first-order models. This achieves a form of clustering of Web sites leading to a limited need of parameters. Additionally, the authors also used a Bayesian Markov chain modeling approach that incorporates prior knowledge about the link structure of the underlying topological network in order to enhance predictive accuracy.

The Markovian property might be wrong. In contrast, literature has also suggested that the memoryless model may not be suitable for modeling human navigation on the Web as e.g., the PageRank does. In [Gonçalves et al., 2009], the authors proposed a model that incorporates random effects by using an agent based model where each agent keeps a list of pages ranked by the number of previous visits that are then leveraged for determining future visits; the model was later extended in [Meiss et al., 2010]. Recently, Chierichetti et al. [2010] have picked up on the study of memory effects in Markov chain models. Their studies suggest that the Markovian memoryless assumption might not hold for human navigation on the Web. However, the authors also have pointed out that it is difficult to determine the appropriate Markov chain order unless given a significantly large amount of data. As inferred in this thesis in Section 2.1, simply choosing the model with the highest likelihood is not enough, as higher order Markov chain models are always better fits to the data [Murphy, 2002]. Hence, this warrants further investigations that specifically account for the higher complexity needed for these higher order models [Pirolli and Pitkow, 1999; Borges and Levene, 2000]. In this thesis, I tackle this issue and investigate memory effects in human navigational trails by utilizing the statistical tools described in Section 2.1.2. As a result, I present a framework for detecting the appropriate Markov chain order given data. In future, researchers can utilize this framework for detecting memory effects in human trails. For a more general approach of looking at memory in network flows please consult [Rosvall et al., 2014].

3. Papers

3.1. Contributions to the Publications

This section elaborates in detail *my* contributions to the main publications of this cumulative thesis.

- [Singer et al., 2014c] Singer, P., Helic, D., Taraghi, B., and Strohmaier, M. (2014c). Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PloS One*, 9(7):e102070

First an foremost, I developed the methodological concepts necessary for this work. These consist of several advanced statistical inference methods for Markov chain models and according statistical comparisons of Markov chain models of varying order. Subsequently, I implemented this approach in Python which is also made open-source[1] for offering researchers a general framework for determining the appropriate Markov chain order given human trail data. For showing the general applicability and mechanics of this framework, I conducted a series of experiments for detecting memory and structure in human navigational trails on the Web.

The idea for this paper stems from discussions between Markus Strohmaier, Denis Helic and myself. Denis Helic contributed to the design of the approach. Behnam Taraghi provided visualizations for the structural aspects studied. All authors contributed to interpreting and discussing the results as well as to the writing of the manuscript.

[1] https://github.com/psinger/PathTools

- [Walk et al., 2014b] Walk, S., Singer, P., Strohmaier, M., Tudorache, T., Musen, M. A., and Noy, N. F. (2014b). Discovering beaten paths in collaborative ontology-engineering projects using markov chains. *Journal of Biomedical Informatics*, 51:254–271

My main contribution for this work was the design of the approach. In detail, I developed and prepared the methodological Markov chain framework that is utilized throughout the experiments of this article.

The ideas for this article stem from various discussions between the authors of this article. Data preparation and the conduction of experiments were done by Simon Walk. The results were mainly interpreted by Simon Walk in coordination with all authors of this article. All authors contributed to the writing of this manuscript.

- [Walk et al., 2014a] Walk, S., Singer, P., and Strohmaier, M. (2014a). Sequential action patterns in collaborative ontology-engineering projects: A case-study in the biomedical domain. In *International Conference on Information and Knowledge Management*

For this article, I was responsible for the design of the approach. To that end, I provided the Markov chain framework which this article utilizes for detecting the appropriate Markov chain order given human edit trails in collaborative ontology engineering projects as well as for predicting human trails. Additionally, I developed a method for studying randomness in human trails. I make the implementations of this method available online[2]. Consequently, I also conducted the experiments for studying randomness and regularities given the human trail data at hand.

The main idea for this work stems from discussions between Simon Walk and myself in consultation with Markus Strohmaier who is the doctoral adviser of both. Simon Walk designed the approach for detecting structural patterns of various length in given data. Subsequently, he was also responsible for conducting the experiments that aimed at the detection of patterns as well as for predicting human trails on the Web. The results were mainly interpreted by Simon Walk and myself. All authors contributed to the writing of the manuscript.

[2] https://github.com/psinger/RunsTest

- [Singer et al., 2013a] Singer, P., Niebler, T., Strohmaier, M., and Hotho, A. (2013a). Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70

For this article, I focused on developing the methodological fundamentals that are utilized for calculating semantic relatedness by leveraging human navigational trails. I make an implementation of the method open-source and available online[3]. Additionally, I was involved into preparing the data (trails, gold-standards, baseline corpora and sampling). I conducted the experiments of this article by applying the method to the data at interest and evaluating the results on a set of gold standards.

The idea for this work stems from discussions between Markus Strohmaier and myself. Thomas Niebler generated parts of the baseline and sampling corpora and designed aspects of the evaluation. He was also responsible for producing the visualizations of this article. All authors were continuously involved in the refinements of the methodological development and experiments as well as the interpretation of the results. All authors contributed to the writing of the manuscript.

- [Singer et al., 2014b] Singer, P., Helic, D., Hotho, A., and Strohmaier, M. (2014b). Hyptrails: A bayesian approach for comparing hypotheses about human trails. *arXiv:1411.2844 [cs.SI]*

In this article, I developed HypTrails, an approach for comparing hypotheses about human trails on the Web. Again, I make an implementation of this approach open-source and available online[4]. Additionally, I prepared the data at hand and conducted all experiments of this article that aim at showing the general mechanics and applicability of HypTrails.

The ideas for this paper stem from regular discussions between the authors of this work. Additionally, the methodological fundaments, experimental setup and results were discussed by all authors of this article on a regular basis. Also, all authors contributed to the writing of this article.

[3] https://github.com/psinger/PathTools

[4] https://github.com/psinger/HypTrails

3.2. Detecting Memory and Structure in Human Navigation Patterns Using Markov Chain Models of Varying Order

This article tackles the first research question concerning the study of memory and structure in human trails and represents the fundament of this thesis. To that end, it presents a framework for detecting the appropriate Markov chain order given human trail data. For doing so, it deploys four different approaches that stem from distinct statistical fields: (i) likelihood, (ii) Bayesian, (iii) information-theoretic and (iv) cross validation methods. These methods evaluate whether higher order Markov chain models are statistically significant better fits to the data compared to lower order models. The article highlights the strengths and weaknesses of each method at hand and I provide an open-source implementation[5] of the framework. This should give researchers an easy-to-handle and comprehensive way to study memory effects in human trails.

For demonstrating the general mechanics and applicability of the framework, colleagues and I have applied it to three distinct human navigational trail corpora for detecting memory and structure in human navigation patterns as presented in this article. The results confirm what this thesis inferred from theory: It is difficult to make plausible statements about higher order Markov chain models given only a limited set of navigational trails. Hence, we argue that the memoryless (first-order) Markov chain model is a plausible model for human navigational data on a page level (i.e., humans navigating over websites) as also mostly indicated in literature. However, by reducing the state space by abstracting away from the page to a topical level, the results indicate memory effects at least on this topical level. This is coupled with the detection of several representative structural patterns found. This argues that regularities, patterns and memory play at least some role in human navigational trails. Hence, these observations warrant further more rigorous investigations which can be tackled by using the Markov chain framework presented in this thesis.

[5] https://github.com/psinger/PathTools

Detecting Memory and Structure in Human Navigation Patterns Using Markov Chain Models of Varying Order

Philipp Singer[1]*, **Denis Helic**[2], **Behnam Taraghi**[3], **Markus Strohmaier**[1,4]

1 GESIS - Leibniz Institute for the Social Sciences, Cologne, Germany, **2** Technical University of Graz, Knowledge Technologies Institute, Graz, Austria, **3** Technical University of Graz, Institute for Information Systems and Computer Media, Graz, Austria, **4** University Koblenz-Landau, Institute for Web Science and Technologies, Koblenz, Germany

Abstract

One of the most frequently used models for understanding human navigation on the Web is the Markov chain model, where Web pages are represented as states and hyperlinks as probabilities of navigating from one page to another. Predominantly, human navigation on the Web has been thought to satisfy the memoryless Markov property stating that the next page a user visits only depends on her current page and not on previously visited ones. This idea has found its way in numerous applications such as Google's PageRank algorithm and others. Recently, new studies suggested that human navigation may better be modeled using higher order Markov chain models, i.e., the next page depends on a longer history of past clicks. Yet, this finding is preliminary and does not account for the higher complexity of higher order Markov chain models which is why the memoryless model is still widely used. In this work we thoroughly present a diverse array of advanced inference methods for determining the appropriate Markov chain order. We highlight strengths and weaknesses of each method and apply them for investigating memory and structure of human navigation on the Web. Our experiments reveal that the complexity of higher order models grows faster than their utility, and thus we confirm that the memoryless model represents a quite practical model for human navigation on a page level. However, when we expand our analysis to a topical level, where we abstract away from specific page transitions to transitions between topics, we find that the memoryless assumption is violated and specific regularities can be observed. We report results from experiments with two types of navigational datasets (goal-oriented vs. free form) and observe interesting structural differences that make a strong argument for more contextual studies of human navigation in future work.

Citation: Singer P, Helic D, Taraghi B, Strohmaier M (2014) Detecting Memory and Structure in Human Navigation Patterns Using Markov Chain Models of Varying Order. PLoS ONE 9(7): e102070. doi:10.1371/journal.pone.0102070

Editor: Zhongxue Chen, Indiana University Bloomington, United States of America

Received January 28, 2014; **Accepted** June 13, 2014; **Published** July 11, 2014

Funding: This research was in part funded by the DFG German Science Fund research project "Pragmatics and Semantics in Social Tagging Systems II" (STR 1191/3-2) as well as the FWF Austrian Science Fund research project "Navigability of Decentralized Information Networks" (P24866). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* Email: philipp.singer@gesis.org

Introduction

Navigation represents a fundamental activity for users on the Web. Modeling this activity, i.e., understanding how predictable human navigation is and whether regularities can be detected has been of interest to researchers for nearly two decades – an example of early work would be work by Catledge and Pitkow [1]. Another example would be [2], who focused on trying to understand preferred user navigation patterns in order to reveal users' interests or preferences. Not only has our community been interested in gaining deeper insights into human behavior during navigation, but also in understanding how models of human navigation can improve user interfaces or information network structures [3]. Further work has focused on understanding whether models of human navigation can help to predict user clicks in order to prefetch Web sites (e.g., [4]) or enhance a site's interface or structure (e.g., [5]). More recently, such models have also been deployed in the field of recommender systems (e.g., [6]).

However, models of human navigation can only be useful to the extent human navigation itself exhibits regularities that can be exploited. An early study on user navigation in the Web by Huberman, Pirolli, Pitkow and Lukose [7], for example, already identified interesting regularities in the distributions of user page visits on a Web site. More recently, Wang and Huberman [8] confirmed these observations and Song, Qu, Blumm and Barabási [9] argued that the regularities in human activities might be based on the inherent regularities of human behavior in general.

The most prominent model for describing human navigation on the Web is the Markov chain model (e.g., [10]), where Web pages are represented as states and hyperlinks as probabilities of navigating from one page to another. Predominantly, the Markov chain model has been memoryless in a wide range of works (e.g., Google's PageRank [11]) indicating that the next state only depends on the current state of a user's Web trail. Recently, a study [12] suggested that human navigation might be better modeled with memory – i.e., the next page depends on a longer history of past clicks. However, this finding is preliminary and does not account for the higher complexity of higher order Markov chain models which is why the memoryless model is still widely used.

Research questions

In this paper, we are interested in shedding a deeper light on regularities in human navigation on the World Wide Web by studying memory and structure in human navigation patterns. We start by investigating memory of human navigational paths over Web sites by determining the order of corresponding Markov chains. We are specifically interested in detecting if the benefit of a larger memory (or higher order Markov chain) can compensate for the higher complexity of the model. In order to understand whether and to what extent human navigation exhibits memory on a topical level, we abstract away from specific page transitions and study memory effects on a topical level by representing click streams as sequences of topics (cf. Figure 1) – note that the terms "topic" and "category" should be seen as synonyms throughout this work. This enables us to (i) move up from the page to topical level and (ii) significantly reduce the complexity of higher order models and therefore (iii) gain deeper insights into memory and structure of human navigational patterns. Finally, we discuss our findings and demonstrate interesting differences between human navigation in free browsing vs. more goal-oriented settings.

Methods and Materials

We study memory and structure in human navigation patterns on three similarly structured datasets: WikiGame (a navigation dataset with known navigation goals), Wikispeedia (another goal-oriented navigation dataset) and MSNBC (a free navigation dataset). For analyzing memory, we use Markov chains to model human behavior and analyze the appropriate Markov chain order – i.e., we investigate whether human navigation is memoryless or not. For model selection – i.e., the process of finding the most appropriate Markov chain order – we resort to a highly diverse array of methods stemming from distinct statistical schools: (i) likelihood [13,14], (ii) Bayesian [15] and (iii) information-theoretic methods [14,16–19]. We supplement these with a (iv) cross validation approach for a prediction task [18]. We thoroughly elaborate each method, put them into relation to each other and also highlight strengths and weaknesses of each. Such detailed derivation of model parameters and the model comparison is, for example, missing in previous work [12], which prevents us from drawing definite conclusions. We apply these methods to our human navigational data in order to get an exhaustive picture about memory in human navigation. Finally, we identify structural aspects by analyzing transition matrices produced by our Markov chain analyses.

Contributions

The main contributions of this work are three-fold:

- First, we deploy four different, yet complementary, approaches for order selection of Markov chain models (likelihood, Bayesian, information-theoretic and cross validation methods) and elaborate their strengths and weaknesses. Hence, our work extends existing studies that model human navigation on the Web using Markov chain models [12]. By applying these methods on navigational Web data, our work presents – to the best of our knowledge – the most comprehensive and systematic evaluation of Markov model orders for human navigational sequences on the Web to date. Furthermore, we make our methods in the form of an open source framework available online (https://github.com/psinger/PathTools) to aid future work [20].
- Our empirical results confirm what we inferred from theory: It is difficult to make plausible statements about the appropriate Markov chain order having insufficient data but a vast amount of states, which is a common situation for Web page navigational paths. All evaluation approaches would favor a zero or first order because the number of parameters grows exponentially with the chain order and the available data is too sparse for proper parameter inferences. Thus, we show further evidence that the memoryless model seems to be a quite practical and legitimate model for human navigation on a page level.
- By abstracting away from the page level to a topical level, the results are different. By representing all datasets as navigational sequences of topics that describe underlying Web pages (cf. Figure 1), we find evidence that topical navigation of humans is not memoryless at all. On three rather different datasets of navigation – free navigation (MSNBC) and goal-oriented navigation (WikiGame and Wikispeedia) – we find mostly consistent memory regularities on a topical level: In all cases, Markov chain models of order two (respectively three) best explain the observed navigational sequences. We analyze the structure of such navigation, identify strategies and the most salient common sequences of human navigational patterns and provide visual depictions. Amongst other structural differences between goal-oriented and free form navigational patterns, users seem to stay in the same topic more frequently for our

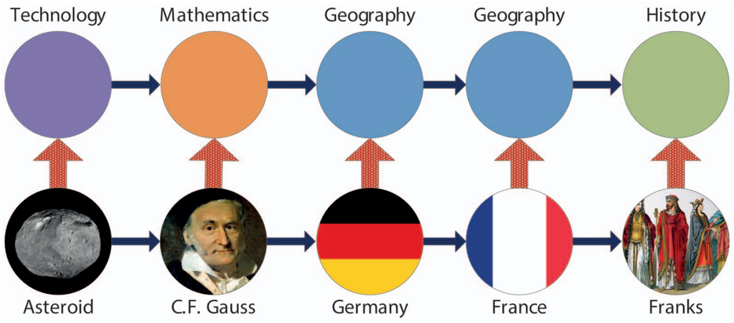


Figure 1. Example of a navigation sequence in the WikiGame dataset. Bottom row of nodes: A user navigates a series of Wikipedia articles, which can be represented as a sequence of Web pages. Top row of nodes: Each Wikipedia article can be mapped to a corresponding topic through Wikipedia's system of categories. This results in a sequence of topics.
doi:10.1371/journal.pone.0102070.g001

free form navigational dataset (MSNBC) compared to both of the goal oriented datasets (Wikigame and Wikispeedia). Our analysis thereby provides new insights into the memory and structure that users employ when navigating the Web that can e.g., be useful to improve recommendation algorithms, web site design or faceted browsing.

The paper is structured as follows: In the section entitled "Related Work" we review the state-of-the-art in this domain. Next, we present our methodology and experimental setup in the sections called "Methods" and "Materials". We present and discuss our results in the section named "Results". In the section called "Discussion we provide a final discussion and the section called "Conclusions" concludes our paper.

Related Work

In the late 1990s, the analysis of user navigational behavior on the Web became an important and wide-spread research topic. Prominent examples are models by Huberman and Adamic [21] that determine how users choose new sites while navigating, or the work by Huberman, Pirolli, Pitkow and Lukose [7] who have shown that strong regularities in human navigation behavior exist and that, for example, the length of navigational paths on the Web is distributed as an inverse Gaussian distribution. These first models of human navigation on the Web set a standard modeling framework for future research - the majority of navigation models have been stochastic henceforth. Common stochastic models of human navigation are Markov chains. For example, the Random Surfer model in Google's PageRank algorithm can be seen as a special case of a Markov chain [11]. Some further examples of the application of Markov chains as models of Web navigation can be found in [10,22–29].

In a Markov chain, Web pages are represented as states and links between the pages are modeled as probabilistic transitions between the states. The dynamics of a user's navigation session, in which she visits a number of pages by following the links between them, can thus be represented as a sequence of states. Specific configurations of model parameters – such as transition probabilities or model orders – have been used to reflect different assumptions about navigation behavior. One of the most influential assumptions in this field to date is the so-called Markovian property, which postulates that the next page that a user visits depends only on her current page, and not on any other page leading to the current one. This assumption is adopted in a number of prevalent models of human navigation in information networks, for example also in the Random Surfer model [11]. However, this property is neglecting the observations stated above that human navigation exhibits strong regularities which hints towards longer memory patterns in human navigation. We argue, that the more consistency human navigation in information networks displays the higher the appropriate Markov chain order should be.

The Markovian assumption might be wrong. The principle that human navigation might exhibit longer memory patterns than the first order Markov chain captures has been investigated in the past (see e.g., [3,10] or [30] for a more general approach of looking at memory in network flows). However, higher order Markov chains have been often disputed for modeling human navigation because the gain of a higher order model did not compensate for the additional complexity introduced by the model [10]. Therefore, it was a common practice to focus on a first order model since it was a reasonable but extremely simple approximation of user navigation behavior (e.g., [25,27,28,31]).

The discussion about the appropriate Markov chain order was just recently picked up again by Chierichetti, Kumar, Raghavan and Sarlos [12]. While the authors' results again show indicators that users on the World Wide Web are not Markovian, the study does not account for the higher complexity of such models and the possible lack of statistically significant gains of these models. Technically, the authors analyzed Markov chain models of different orders by measuring the likelihood of real navigational sequences given a particular model. In the next step, the authors compared the models by their likelihoods and found that the Markovian assumption does not hold for their given data and, thus, higher order Markov chain models seem to be more appropriate. As a result, the authors argue that users on the World Wide Web are not Markovian. However, their results come with certain limitations, such as the fact that choosing the model with the highest likelihood is biased towards models with more parameters. Because lower order models are always nested within higher order models and as higher order Markov chains have exponentially more parameters than lower order models (potential overfitting), they are always a better fit for the data [18]. Thus, higher order models are naturally favored by their improvements in likelihoods. A more comprehensive view on this issue shows that there exists a broad range of established model comparison techniques that also take into the account the complexity of a model in question [14–17,19,32,33].

Moreover, the principle objects of interest in the majority of the past studies are transitions between Web pages. Only a few studies [27,34,35] investigate navigation as transitions between Web page features, such as the content or context of those Web pages.

Methods

In the following, we briefly introduce Markov chains before discussing an expanded set of methods for order selection, including *likelihood*, *Bayesian*, *information-theoretic* and *cross validation* model selection techniques.

Markov Chains

Formally, a discrete (time and space) finite Markov chain is a stochastic process which amounts to a sequence of random variables $X_1, X_2, ..., X_n$. For a Markov chain of the first order, i.e., for a chain that satisfies the memoryless Markov property the following holds:

$$P(X_{n+1} = x_{n+1} | X_1 = x_1, X_2 = x_2, ..., X_n = x_n) =$$

$$P(X_{n+1} = x_{n+1} | X_n = x_n) \tag{1}$$

This classic first order Markov chain model is usually also called a *memoryless model* as we only use the current information for deriving the future and do not look into the past. For all our models we assume *time-homogeneity* – the probabilities do not change as a function of time. To simplify the notation we denote data as a sequence $D = (x_1, x_2, ..., x_n)$ with states from a finite set S. With this simplified notation we write the Markov property as:

$$p(x_{n+1} | x_1, x_2, ..., x_n) = p(x_{n+1} | x_n) \tag{2}$$

As we are also interested in higher order Markov chain models in this article – i.e., memory models – we now also define a

Markov chain for an arbitrary order k with $k\in\mathbb{N}$ – or a chain with memory k. In a Markov chain of k-th order the probability of the next state depends on k previous states. Formally, we write:

$$p(x_{n+1}|x_1,x_2,...,x_n)=p(x_{n+1}|x_n,x_{n-1},...,x_{n-k+1}) \qquad (3)$$

Markov chains of a higher order can be converted into Markov chains of order one in a straightforward manner – the set of states for a higher order Markov chain includes all sequences of length k (resulting in a state set of size $|S|^k|S|$). The transition probabilities are adjusted accordingly.

A Markov model is typically represented by a transition (stochastic) matrix P with elements $p_{ij}=p(x_j|x_i)$. Since P is a stochastic matrix it holds that for all i:

$$\sum_j p_{ij}=1 \qquad (4)$$

Please note, that for a Markov chain of order the current state x_i can be a compound state of length k – it is a sequence of past k states. Throughout this paper we use this simpler notation, but one should keep in mind that x_i differs for distinct orders k.

For the sake of completeness, we also allow k to be zero. In such a *zero order* Markov chain model the next state does not depend on any current or previous events, but simply can be seen as a *weighted random selection* – i.e., the probability of choosing a state is defined by how frequently it occurs in the navigational paths. This should serve as a baseline for our evaluations.

Next, we want to estimate the vector θ of parameters of a particular Markov chain that generated observed data D as well as determine the appropriate Markov chain order. For a Markov chain the model parameters are the elements p_{ij} of the transition matrix P, i.e., $\theta=P$.

Model Selection

In this article our main goal is to determine the appropriate order of a Markov chain – i.e., the appropriate length of the memory. For doing so, we resort to well established statistical methods. As we want to provide a preferably complete array of methods for doing so, we present and apply methods from distinct statistical schools: (i) likelihood, (ii) Bayesian and (iii) information-theoretic methods. Note that no official classification of statistical schools is available; some may also argue that there are only the two competing schools of frequentists (which we do not explicitly discuss in this article) and Bayesians. The categorization used here is motivated by a short blog post (see http://labstats.net/articles/overview.html). We also supplement the methods coming from these three schools by providing a model selection technique usually known from machine learning: (iv) cross validation. We provide an overall ample view of methods and discuss advantages and limitations of each in the following sections.

Likelihood Method

The term *likelihood* was coined and popularized by R. A. Fisher in the 1920's (see e.g, [13] for a historic recap of the developments). Likelihood can be seen as a central element of statistics and we will also see in the following sections that other methods also resort to the concept. The likelihood is a function of the parameters θ and it equals to the probability of observing the data given specific parameter values:

$$P(D|\theta) \ =p(x_n|x_{n-1})p(x_{n-1}|x_{n-2})...p(x_2|x_1)p(x_1)$$

$$=p(x_1)\prod_i\prod_j p_{ij}^{n_{ij}}, \qquad (5)$$

where n_{ij} is the number of transition from state x_i to state x_j in D.

Fisher also popularized the so-called *maximum likelihood estimate (MLE)* which has a very intuitive interpretation. This is the estimation of the parameters θ – i.e., transition probabilities – that most likely generated data D. Concretely, the maximum likelihood estimate $\hat{\theta}_{MLE}$ are the values of the parameters θ that maximize the likelihood function, i.e., $\hat{\theta}_{MLE}=\arg\max_\theta \ P(D|\theta)$ (a thorough introduction to MLE can be found in [36]).

The maximum likelihood estimation for Markov chains is an example of an optimization problem under constraints. Such optimization problems are typically solved by applying Lagrange multipliers. To simplify the calculus we will work with the log-likelihood function $\mathcal{L}(\mathcal{P}(\mathcal{D}|\theta))=logP(D|\theta)$. Because the *log* function is a monotonic function that preserves order, maximizing the log-likelihood is equivalent to maximizing the likelihood function. Thus, we have:

$$\mathcal{L}(\mathcal{P}(\mathcal{D}|\theta))= \ log\left(p(x_1)\prod_i\prod_j p_{ij}^{n_{ij}}\right)$$

$$= \ logp(x_1)+\sum_i\sum_j n_{ij}logp_{ij} \qquad (6)$$

Our constraints capture the fact that each transition matrix row sums to 1:

$$\sum_j p_{ij}=1 \qquad (7)$$

We have n rows and therefore we need n Lagrange multipliers $\lambda_1,\lambda_2,...,\lambda_n$. We can rewrite the constraints using Lagrange multipliers as:

$$\lambda_i\left(\sum_j p_{ij}-1\right)=0 \qquad (8)$$

Now, the new objective function is:

$$f(\lambda,\theta)=\mathcal{L}(P(D|\theta))-\sum_i\lambda_i\left(\sum_j p_{ij}-1\right) \qquad (9)$$

To maximize the objective function we set partial derivatives with respect to λ_i to 0, which gives back the original constraints.

Further, we set partial derivatives with respect to p_{ij} to 0 and solve the equation system for p_{ij}. This gives:

$$p_{ij} = \frac{n_{ij}}{\sum_j n_{ij}} \tag{10}$$

Thus, the maximum likelihood estimate for a specific p_{ij} is the number of transitions from state x_i to state x_j divided by the total number of transitions from state x_i to any other state. For example, in a navigation scenario the maximum likelihood estimate for a transition from page A to page B is the number of clicks on a link leading to page B from page A divided by the total number of clicks on page A.

Our concrete goal is to determine the appropriate order of a Markov chain. Using the log-likelihoods of the specific order models is not enough, as we will always get a better fit to our training data using higher order Markov chains. The reason for this is that lower order models are nested within higher order models. Also, the number of parameters increases exponentially with k which may result in overfitting [18] since we can always produce better fits to the data with more model parameters. To demonstrate this behavior, we produced a random navigational dataset by randomly (uniformly) picking a next click state out of a list of arbitrary states. One of these states determines that a path is finished and a new one begins. With this process we could generate a random path corpus that is close to one main dataset of this work (Wikigame topic dataset explained in the section called "Materials"). Concretely, we as well chose 26 states and the same number of total clicks. Purely from our intuition, such a process should produce navigational patterns with an appropriate Markov chain order of zero or at maximum one. However, if we look at the log-likelihoods depicted in Figure 2 we can observe that the higher the order the higher the corresponding log likelihoods are.

This strongly suggests that – as previously explained – looking at the log-likelihoods is not enough for finding the appropriate Markov chain order. Hence, we first resort to a well-known statistical likelihood tool for comparing two models – the so-called *likelihood ratio test*.

This test is suited for comparing the fit of two composite hypothesis where one model – the so-called *null model* k – is a special case of the *alternative model* m. The test is based on the log likelihood ratio, which expresses how much more likely the data is with the alternative model than with the null model. We follow the notation provided by Tong [14] and denote the ratio as ${}_k\eta_m$:

$${}_k\eta_m = -2\mathcal{L}(\mathcal{P}(\mathcal{D}|\theta_k)) - \mathcal{L}(\mathcal{P}(\mathcal{D}|\theta_m))) \tag{11}$$

To address the overfitting problem we perform a significance test on this ratio. The significance test recognizes whether a better fit to data comes only from the increased number of parameters. The test calculates the p-value of the likelihood ratio distribution. Whenever the null model is nested within the alternative model the likelihood ratio approximately follows a χ^2 distribution with degrees of freedom specified by $(|S|^m - |S|^k)(|S|-1)$. If the p-value is below a specific significance level we can reject the null hypothesis and prefer the alternative model [32] – note that this method also utilizes mechanisms usually known from the frequentist school; i.e., hypothesis testing.

Likelihood ratios and corresponding tests have been shown to be a very understandable approach of specifying evidence [37]. They also have the advantage of specifying a clear value (i.e., the likelihood ratio) with can give us intuitive meaning about the advantage of one model over the other. However, the likelihood-ratio test also has limitations like that it only works for nested models, which is fine for our approach but may be problematic for other use cases. It also requires us to use elements from frequentist approaches (i.e., the p-value) for deciding between two models

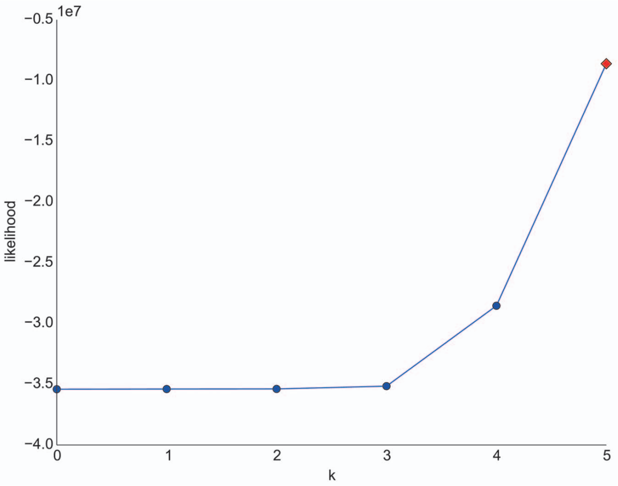


Figure 2. Log-likelihoods for random path dataset. Simple log-likelihoods of varying Markov chain orders would suggest higher orders as the higher the order the higher the corresponding log-likelihoods are. This suggests that looking at these log-likelihoods is not enough for finding the appropriate Markov chain order as methods are necessary that balance the goodness-of-fit against the number of model parameters.
doi:10.1371/journal.pone.0102070.g002

which have been criticized in the past (e.g., [38]). Furthermore, we are only able to compare two models with each other at a time. This makes it difficult to choose one single model as the most likely one as we may end up with several statistical significant improvements. Also, as we increase the number of hypothesis in our test, we as well increase the probability that we find at least one significant result (Type 1 error). We could tackle this problem by e.g., applying the *Bonferroni correction* which we leave open for future work.

Bayesian Method

Bayesian inference is a statistical method utilizing the Bayes' rule – Rev. Thomas Bayes started to talk about the Bayes theorem in 1764 – for updating prior believes with additional evidence derived from data. A general introduction to Bayesian inference can e.g., be found in [39]; in this article we focus on explaining the application for deriving the appropriate Markov chain order (see [15] for further details).

In Bayesian inference data and the model parameters are treated as random variables (cf. MLE where parameters are unknown constants). We start with a joint probability distribution of data D and parameters θ_k given a model M; that is given a Markov chain of a specified order k. Thus, we are interested in $P(D,\theta_k|M_k)$.

The joint distribution $P(D,\theta_k|M_k)$ can be written as the product of the conditional probability of data D given the parameters θ_k and the marginal distribution of the parameters, or we can write this joint distribution as the product of the conditional probability of the parameters given the data and the marginal distribution of the data.

Solving then for the posterior distribution of parameters given data and a model we obtain the famous Bayes rule:

$$P(\theta_k|D,M_k) = \frac{P(D|\theta_k,M_k)P(\theta_k|M_k)}{P(D|M_k)}, \tag{12}$$

where $P(\theta_k|M_k)$ is the prior probability of model parameters, $P(D|\theta_k,M_k)$ is the likelihood function; that is the probability of observing the data given the parameters, and $P(D|M_k)$ is the evidence (marginal likelihood). $P(\theta_k|D,M_k)$ is the posterior probability of the parameters, which we obtain after we update the prior with the data.

For a more detailed and an in-depth technical analysis of Bayesian inference of Markov chains we point to an excellent discussion of the topic in [15].

Likelihood. As previously, we have:

$$P(D|\theta_k,M_k) = p(x_1)\prod_i\prod_j p_{ij}^{n_{ij}} \tag{13}$$

Prior. The prior reflects our (subjective or objective) belief about the parameters before we see the data. In Bayesian inference, conjugate priors are of special interest. Conjugate priors result in posterior distributions from the same distribution family. In our case, each row of the transition matrix follows a categorical distribution. The conjugate prior for categorical distribution is the Dirichlet distribution. Further information on applying Dirichlet conjugate prior and dealing with Dirichlet process can be found in [40]. The Dirichlet distribution is defined as $Dir(\alpha)$:

$$Dir(\alpha) = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)}\prod_j x_j^{\alpha_j - 1}, \tag{14}$$

where Γ is the gamma function, $\alpha_j > 0$ for each j and $\sum_j x_j = 1$ is a probability simplex. The probability outside of the simplex is 0.

The *hyperparameters* α reflect our assumptions about the parameters θ before we have observed the data. We can think about the hyperparameters as fake counts in the transition matrix of a Markov chain. A standard uninformative selection for hyperparameters is a uniform prior – for example, we set $\alpha_j = 1$ for each j.

Thus, for row i of the transition matrix we have the following prior:

$$Dir(\alpha_i) = \frac{\Gamma(\sum_j \alpha_{ij})}{\prod_j \Gamma(\alpha_{ij})}\prod_j p_{ij}^{\alpha_{ij}-1} \tag{15}$$

As before, it holds that:

$$\sum_j p_{ij} = 1 \tag{16}$$

The prior for the complete transition matrix is the product of the Dirichlet distributions for each row:

$$P(\theta_k|M_k) = \prod_i \frac{\Gamma(\sum_j \alpha_{ij})}{\prod_j \Gamma(\alpha_{ij})}\prod_j p_{ij}^{\alpha_{ij}-1} \tag{17}$$

Evidence. To calculate the evidence we take a weighted average over all possible values of the parameters θ_k. Thus, we need to integrate out the parameters θ_k.

$$P(D|M_k) = \int P(D|\theta_k,M_k)P(\theta_k|M_k)d\theta_k \tag{18}$$

$$P(D|M_k) = \int P(D|\theta_k,M_k)P(\theta_k|M_k)d\theta_k$$

$$= \int p(x_1)\prod_i\prod_j p_{ij}^{n_{ij}}\prod_i \frac{\Gamma(\sum_j \alpha_{ij})}{\prod_j \Gamma(\alpha_{ij})}\prod_j p_{ij}^{\alpha_{ij}-1} d\theta_k$$

$$= p(x_1)\prod_i \frac{\Gamma(\sum_j \alpha_{ij})}{\prod_j \Gamma(\alpha_{ij})}\int \prod_j p_{ij}^{n_{ij}}\prod_j p_{ij}^{\alpha_{ij}-1} d\theta_k$$

$$= p(x_1)\prod_i \frac{\Gamma(\sum_j \alpha_{ij})}{\prod_j \Gamma(\alpha_{ij})}\int \prod_j p_{ij}^{n_{ij}+\alpha_{ij}-1} d\theta_k$$

Please note, that:

$$\int \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_j x_j^{\alpha_j - 1} dx = 1$$

$$\frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \int \prod_j x_j^{\alpha_j - 1} dx = 1$$

$$\int \prod_j x_j^{\alpha_j - 1} dx = \frac{\prod_j \Gamma(\alpha_j)}{\Gamma(\sum_j \alpha_j)}$$

Thus, we have

$$\int \prod_j p_{ij}^{n_{ij}+\alpha_{ij}-1} d\theta_k = \frac{\prod_j \Gamma(n_{ij}+\alpha_{ij})}{\Gamma(\sum_j (n_{ij}+\alpha_{ij}))} \tag{19}$$

And thus,

$$P(D|M_k) = p(x_1) \prod_i \frac{\Gamma(\sum_j \alpha_{ij})}{\prod_j \Gamma(x_{ij})} \frac{\prod_j \Gamma(n_{ij}+\alpha_{ij})}{\Gamma(\sum_j (n_{ij}+\alpha_{ij}))} \tag{20}$$

Posterior. For the posterior distribution over the parameters θ_k we obtain:

$$P(\theta_k|D,M_k) = \prod_i \prod_j p_{ij}^{n_{ij}} \prod_i \prod_j p_{ij}^{\alpha_{ij}-1} \frac{\Gamma(\sum_j (n_{ij}+\alpha_{ij}))}{\prod_j \Gamma(n_{ij}+\alpha_{ij})}$$

$$= \prod_i \prod_j p_{ij}^{n_{ij}+\alpha_{ij}-1} \frac{\Gamma(\sum_j (n_{ij}+\alpha_{ij}))}{\prod_j \Gamma(n_{ij}+\alpha_{ij})}$$

This equation is the product of the Dirichlet distributions for each row with parameters $n_j + \alpha_j$:

$$P(\theta_k|D,M_k) = \prod_i Dir(n_i + \alpha_i) \tag{21}$$

The posterior distribution is a combination of our prior belief and the data that we have observed. In fact, the expectation and the variance of the posterior distribution are:

$$E[p_{ij}] = \frac{n_{ij}+\alpha_{ij}}{\sum_j (n_{ij}+\alpha_{ij})} \tag{22}$$

$$Var[p_{ij}] = \frac{(n_{ij}+\alpha_{ij})(\sum_j (n_{ij}+\alpha_{ij}) - (n_{ij}+\alpha_{ij}))}{(\sum_j (n_{ij}+\alpha_{ij}))^2 (\sum_j (n_{ij}+\alpha_{ij}) + 1)} \tag{23}$$

We can rewrite the expectation as:

$$E[p_{ij}] = \frac{1}{\sum_j (n_{ij}+\alpha_{ij})} \left(\sum_j n_{ij} \frac{n_{ij}}{\sum_j n_{ij}} + \sum_j \alpha_{ij} \frac{\alpha_{ij}}{\sum_j \alpha_{ij}} \right) \tag{24}$$

Setting $c = \frac{\sum_j n_{ij}}{\sum_j (n_{ij}+\alpha_{ij})}$, we can rewrite the expectation of the posterior distribution as:

$$E[p_{ij}] = c \frac{n_{ij}}{\sum_j n_{ij}} + (1-c) \frac{\alpha_{ij}}{\sum_j \alpha_{ij}} \tag{25}$$

Thus, the posterior expectation is a *convex combination* of the MLE and the prior. When the number of the observation becomes large ($n_{ij} \gg \alpha_{ij}$) then c tends to 1, and the posterior expectation tends to the MLE.

By setting $\alpha_{ij} = 1$ for each i and j we effectively obtain Laplace's prior; that is we apply Laplace smoothing [18].

For model selection we adopt once more the Bayesian inference (again see [15] for a thorough discussion). We have a set M of models M_k with varying order k and are interested in deciding between several models (c.f. [41]). We are interested in the joint probability distribution $P(D,M_k)$ of data D and a model M_k. We can write the joint distribution as a product of a conditional probability (of data given a model, or of a model given the data) and a prior marginal distribution (of data or a model) and by solving for the posterior distribution of a model given the data we again obtain the Bayes rule:

$$P(M_k|D) = \frac{P(D|M_k)P(M_k)}{P(D)}, \tag{26}$$

where $P(D)$ is the weighted average over all models M_k:

$$P(D) = \sum_k P(D|M_k)P(M_k). \tag{27}$$

The likelihood of data D given a model M_k is the evidence $P(D|M_k)$ given by Equation 20, which is the weighted average over all possible model parameters θ_k given the model M_k.

Following Strelioff, Crutchfield and Hübler [15], we select two priors over the model set M – a uniform prior and a prior with an exponential penalty for the higher order models [15]. The uniform prior assigns the identical probability for each model:

$$P(M_k) = \frac{1}{|M|}. \tag{28}$$

With the uniform prior we obtain the following expression for the posterior probability of a model M_k given the data:

$$P(M_k|D) = \frac{P(D|M_k)}{\sum_k P(D|M_k)}. \tag{29}$$

The prior with the exponential penalty can be defined as:

$$P(M_k) = \frac{e^{-|S_k|}}{\sum_k e^{-|S_k|}}, \tag{30}$$

where $|S_k|$ is the number of states of the model M_k and can be calculated as:

$$|S_k| = |S|^k(|S|-1), \tag{31}$$

with $|S|$ being the number of states of the model of order 1.

After solving for the posterior distribution for the prior with the exponential penalty we obtain:

$$P(M_k|D) = \frac{P(D|M_k)e^{-|S_k|}}{\sum_k P(D|M_k)e^{-|S_k|}}. \tag{32}$$

The calculations are best implemented with log-evidence and logarithms of the gamma function to avoid underflow since the numbers are extremely small. To implement the sum for the normalizing constant in the denominator we apply the so-called *log-sum-exp trick* [42]. First, we calculate the log-evidence: $logP(D|M_k)$ and then calculate the logarithm of the normalizing constant $log(C)$:

$$log(C) = log(\sum_k e^{log(P(D|M_k))}). \tag{33}$$

A direct calculation of $e^{log(P(D|M_k))}$ results in an underflow, and thus we pull the largest log-evidence $E_{max} = max(log(P(D|M_k))$ out of the sum:

$$log(C) = E_{max} + log(\sum_k e^{log(P(D|M_k)) - E_{max}}). \tag{34}$$

One downside of using Bayesian model selection is that it is frequently difficult to calculate Bayes factors. Concretely, it is often complicated to calculate the necessary integral analytically and one needs to resort to various alternatives in order to avoid this problem. Nowadays, several such methods exist: e.g., asymptotic approximation or sampling from the posterior (MCMC, Gibbs) [41]. Also, we need to specify prior distributions for the parameters of each model. As elaborated by Kass and Raftery [41], one approach is to use the BIC (see the next section entitled "Information-theoretic Methods") which gives an appropriate approximation given one specific prior.

Compared to the likelihood ratio test (see section entitled Likelihood Method), the Bayesian model selection technique does not require the models to be nested. The main benefit of Bayesian model selection is that it includes a natural *Occam's razor* – i.e., a penalty for too much complexity – which helps us to avoid overfitting [41,43–45]. The Occam's razor is a principle that advises to prefer simpler theories over more complex ones. Based on this definition there is no need to include extra complexity control as we e.g., additionally did for our exponential penalty. We see this though as a nice further control mechanism for cautiously penalizing model complexity and for validating the natural Occam's razor.

Information-theoretic Methods

Information-theoretic methods are based on concepts and ideas derived from information theory with a specific focus on *entropy*. In the following we will provide a description of the two probably most well-known methods; i.e., AIC and BIC. A thorough overview of information-theoretic methods can e.g., be found in various work by K. P. Burnham [46,47].

Akaike information criterion (AIC). Akaike [16] introduced in 1973 a one dimensional statistic for determining the optimal model from a class of competing models. The criterion is based on Kullback-Leibler divergence [48] and the asymptotic properties of the likelihood ratio statistics described in the section entitled "Likelihood Method". The approach is based on minimization of AIC (minimum AIC estimate – MAICE) amongst several competing models [33] and has been first used for Markov chains by Tong [14]. Hence, we define the AIC based on the choice of a loss function proposed by Tong [14]:

$$AIC(k) = {}_k\eta_m - 2(|S|^m - |S|^k)(|S|-1) \tag{35}$$

The test represents an asymptotic version of the likelihood ratio test defined in Equation 11 for composite hypothesis. The idea is to choose m reasonably high and test lower order models until an optimal order is found. MAICE chooses the order k which exhibits the minimum AIC score and tries to balance between overfitting and underfitting [33].

Bayesian Information Criterion (BIC). In 1978 Schwarz [19] introduced this criterion which can be seen as an approximation of the Bayes factor for Bayesian model selection (see the previous section entitled "Bayesian Method"). It is similar to the AIC introduced above with the difference that it penalizes higher order models even more by adding an additional penalization for the number of observations [17]:

$$BIC(k) = {}_k\eta_m - (|S|^m - |S|^k)(|S|-1)ln(n) \tag{36}$$

Again we choose m reasonably high and test lower order models against it. The penalty function is the degree of freedom multiplied with the natural logarithm of the number of observations n. This function converges to infinity at a still slow enough rate and hence, grants a consistent estimator of the Markov chain order [17].

Frequently, both AIC and BIC suggest the same model. However, there are certain cases, where they might slightly disagree. In model selection literature there is a still ongoing debate of whether one should prefer AIC or BIC over each other – e.g., see [49] for a critique of the BIC for model selection. However, as pointed out by Burnham and Anderson [47], each has its strength and weaknesses in distinct domains. The authors emphasize that both can be seen as either frequentist or Bayesian procedures. In case of inequality, Katz [17] suggests to investigate the patterns further by simulating observations and investigate distinct sample sizes. In this paper we instead apply additional model comparison techniques to further analyze the data.

The performance of AIC and BIC has also been investigated in the terms of determining the appropriate Markov chain order which is the main goal of this article. R. W. Katz [17] pointed out that by using AIC there is the possibility of overestimating the true order independent of how large the data is. Hence, he points out that AIC is an inconsistent method. Contrary, he emphasizes that BIC is a consistent estimator – i.e., if there is a true underlying model BIC will select it with enough data. Alas, it does not perform well for small sample sizes (see also [50]). Nonetheless, AIC is the most used estimator for determining the appropriate order, maybe due to higher efficiency for smaller data samples, as elaborated by Baigorri, Gonçalves and Resende [51].

While both AIC and BIC seem at first to be very similar to the likelihood ratio test (see section entitled "Likelihood Method) there are some elementary differences. First and foremost, they can also be applied for non-nested models [46]. Moreover, they do not need to resort to hypothesis testing. BIC is also closely related to Bayesian model selection techniques; specifically to the Bayes factor (see section called "Bayesian Method"). Kass and Raftery [41] emphasize the advantages of BIC over the Bayes factor by pointing out that it can be applied even when the priors are hard to set. Also, it can be a rough approximation to the logarithm of the Bayes factor if the number of observations is large. BIC is also declared as being well suited for scientific reporting.

Finally, we want to point out that one could also see AIC as being best for prediction, while BIC might be better for explanation. Also, as pointed out by M. Stone [52], AIC is asymptotically equivalent to cross validation (see the section entitled "Cross Validation Method") if both use maximum likelihood estimation.

Cross Validation Method

Another – quite natural – way of determining the appropriate order of a Markov chain is cross-validation [12,18]. The basic idea is to estimate the parameters on a training set and validate the results on an independent testing set. In order to reduce variance we perform a stratified 10-fold cross-validation. In difference to a classic machine learning scenario, we refer to stratified as a way of keeping approximately the equal amount of observations in each fold. Thus, we keep approximately 10% of all clicks in a single fold.

With this method we focus on prediction of the next user click. Markov chains have been already used to prefetch the next page that the user most probably will visit on the next click. In the simplest scenario, this prefetched page is the page with the highest transition probability from the current page. To measure the prediction accuracy we measure the average rank of the actual page in sorted probabilities from the transition matrix. Thus, we determine the rank of the next page x_{n+1} in the sorted list of transition probabilities (expectations of the Bayesian posterior) of the current page x_n (see the section named "Markov Chains"). We then average the rank over all observations in the testing set. Hence, we can formally define the average rank $\overline{r(D_f)}$ of a fold D_f for some arbitrary model M_k the following way:

$$\overline{r(D_f)} = \frac{\sum_i \sum_j n_{ij} r_{ij}}{\sum_i \sum_j n_{ij}}, \tag{37}$$

where n_{ij} is the number of transition from state x_i to state x_j in D_f and r_{ij} denotes the rank of x_j in the i-th row of the transition matrix.

For ranking the states in a row of the matrix, we resort to *modified competition ranking*. This means that if there is a tie between two or more values, we assign the maximum rank of all ties to each corresponding one; i.e., we leave the gaps before a set of ties (e.g., "14445" ranking). By doing so, we assign the worst possible ranks to ties. One important implication of this methodology is that we include a natural penalty (a natural Occam's razor) for higher order Markov chains. The reason for this is that the transition matrices generally become sparser the higher the order. Hence, we come up with many more ties and the chance is higher that we assign higher ranks for observed transitions in the testing data. The most extreme case happens when we do not have any information available for observations in the testing set (which frequently happens for higher orders); then we assign the maximum rank (i.e., the number of states) to all states. We finally average the ranks over all folds for a given order and suggest the model with the lowest average rank. In order to confirm our findings we also applied an additional way of determining the accuracy which is motivated by a typical evaluation technique known from link predictors [53]. Concretely, it counts how frequently the true next click is present in the TopK (k = 5) states determined by the probabilities of the transition matrix. In case of ties in the TopK elements we randomly draw from the ties. By applying this method to our data we can mirror the evaluation results obtained by using the described and used ranking technique. Note that we do not explicitly report the additional results of this evaluation method throughout the paper.

This method requires priors (i.e., fake counts; see the section named "Bayesian Method") – otherwise prediction of unseen states is not possible. It also resorts to the maximum likelihood estimate for calculating the parameters of the models as described in the section entitled "Likelihood Method". Also, as shown in the previous section called "Information-theoretic Methods" cross validation has asymptotic equivalence to AIC.

One disadvantage of cross validation methods usually is that the results are dependent on how one splits the data. However, by using our stratified k-fold cross validation approach, we counteract this problem as it matters less of how the data is divided. Yet, by doing so we need to rerun the complete evaluation k times, which leads to high computational expenses compared to the other model selection techniques described earlier and we have to manually decide of which k to use. One main advantage of this method is that eventually each observation is used for both training and testing.

Materials

In this paper, we perform experiments on three datasets. While the first two datasets (WikiGame and Wikispeedia) are representatives of goal-oriented navigation scenarios (where the target node for each navigation sequence is known beforehand), the third dataset (MSNBC) is representative of free navigation on the Web (where we have no knowledge about the targets of navigation).

Wikigame dataset

This dataset is based on the online game *TheWikiGame* (http://thewikigame.com/). The game platform offers a multiplayer game, where users navigate from a randomly selected Wikipedia page (the start page) to another randomly selected Wikipedia page (the target page). All pairs of start and target pages are connected through Wikipedia's underlying network. The users are only allowed to click on Wikipedia links or on the browser back button to reach the target page, but they are not allowed to use search functionality.

In this study, we only considered click paths of length two or more going through the main article namespace in Wikipedia. Table 1 shows some main characteristics of our Wikigame dataset.

As motivated in Section "Introduction", we will represent the navigational paths through Wikipedia twofold: (a) each node in a path is represented by the corresponding Wikipedia page ID – we

Table 1. Dataset statistics.

	Wikigame	Wikispeedia	MSNBC
#Page Ids	360,417	n/a	n/a
#Topics	25	15	17
#Paths	1,799,015	43,772	624,383
#Visited nodes	10,758,242	259,019	4,333,359

doi:10.1371/journal.pone.0102070.t001

refer to this as the *Wikigame page* dataset – and (b) each node in a path is represented by a corresponding Wikipedia category (representing a specific topic) – we call this the *Wikigame topic* dataset. For the latter dataset we determine a corresponding top level Wikipedia category (http://en.wikipedia.org/wiki/Category:Main_topic_classifications) in the following way. The majority of Wikipedia pages belongs to one or more Wikipedia categories. For each of these categories we find a shortest path to the top level categories and select a top level category with the shortest distance. In the case of a tie we pick a top level category uniformly at random. Finally, we replace all appearances of that page with the chosen top level category. Thus, in this new dataset we replaced each navigational step over a page with an appropriate Wikipedia category (topic) and the dataset contains paths of topics which users visited during navigation (see Figure 1). Figure 3 illustrates the distinct topics and their corresponding occurrence frequency (A).

Wikispeedia dataset

This dataset is based on a similar online game as the Wikigame dataset called *Wikispeedia* (http://www.cs.mcgill.ca/~rwest/wikispeedia/). Again, the players are presented with two randomly chosen Wikipedia pages and they are as well connected via the underlying link structure of Wikipedia. Furthermore, users can also select their own start and target page instead of getting randomly chosen ones. Contrary to the Wikigame, this game is no multiplayer game and you do not have a time limit. Again, we only look at navigational paths with at least two nodes in the path. The main difference to the Wikigame dataset is that Wikispeedia is played on a limited version of Wikipedia (Wikipedia for schools http://schools-wikipedia.org/) with around 4,600 articles. Some main characteristics are presented in Table 1. Conducted research and further explanations of the dataset can be found in [35,54–56].

As we want to look at transitions between topics we determine a corresponding top level category (topic) for each page in the dataset. We do this in similar fashion as for our Wikigame dataset, but the Wikipedia version used for Wikispeedia has distinct top level categories compared to the full Wikipedia. Figure 3 illustrates the distinct categories and their corresponding occurrence frequency (B).

MSNBC dataset

This dataset (http://kdd.ics.uci.edu/databases/msnbc/msnbc.html) consists of Web navigational paths from MSNBC (http://msnbc.com) for a complete day. Each single path is a sequence of page categories visited by a user within a time frame of 24 hours. The categories are available through the structure of the site and include categories such as *news*, *tech*, *weather*, *health*, *sports*, etc. In this dataset we also eliminate all paths with just a single click. Table 1 shows the basic statistics for this dataset and in Figure 3 the frequency of all categories of this dataset are depicted (C).

Data preparation

Each dataset D consists of a set of paths $\mathbb{P}$. A single path contains a single game in the Wikigame and Wikispeedia dataset or a single navigation session in the MSNBC dataset. A path p is defined as a n-tuple $(v_1,\ldots,v_n)$ with $v_i \in V, 1 \leq i \leq n$ and $(v_i, v_{i+1}) \in E, 1 \leq i \leq n-1$ where V is the set of all nodes in $\mathbb{P}$ and E is the set of all observed transitions in $\mathbb{P}$. We also define the length of a path $len(p)$ as the length of the corresponding tuple $(v_1,\ldots,v_n)$. Additionally, we want to define $\mathbf{p} = \{v_k | k = 1 \ldots n\}$ as the set of nodes in a path p. Note that $|\mathbf{p}| \leq n$. The finite state set S needed for Markov chain modeling is originally the set of vertices V in a set of paths $\mathbb{P}$ given a specific dataset D. To prepare the paths for estimation of parameters of a Markov chain of order k, we separate single paths by prepending a sequence of k generic *RESET* states to each path, and also by appending one *RESET* state at the end of each path. This enables us to connect independent paths and – through the addition of the *RESET* state – to forget the history between different paths. Hence, we end up with an ergodic Markov chain (see [12]). With this artificial *RESET* state, the final number of states is $|S|+1$.

Results

In this section we present the results obtained from analyzing human navigation patterns based on our datasets at hand introduced in Section "Materials". We begin by presenting the results of our investigations of memory – i.e., appropriate Markov chain order using the Markov chain methods thoroughly explained in the section called "Methods" – of user navigation patterns in the section entitled "Memory". Based on these calculations and observations we dig deeper into the structure of human navigation and try to find consistent patterns – i.e., specific sequences of navigated states – in the section named "Structure".

Memory

We start by analyzing human navigation over Wikipedia pages on the Wikigame page dataset. Afterwards, we will focus on our topic datasets for getting insights on a topical level.

Page navigation

Wikigame page dataset. The initial Markov chain model selection results (see Figure 4) obtained from experiments on the Wikigame page dataset confirm our theoretical considerations. We observe that the likelihoods are rising with higher Markov chain orders (confirming what [12] found) which intuitively would indicate a better fit to the data using higher order models. However, the likelihood grows per definition with increasing order and number of model parameters and therefore, the likelihood based methods for model selection fail to penalize the increasing model complexity (c.f. Section "Likelihood Method"). All other applied methods take the model complexity into account.

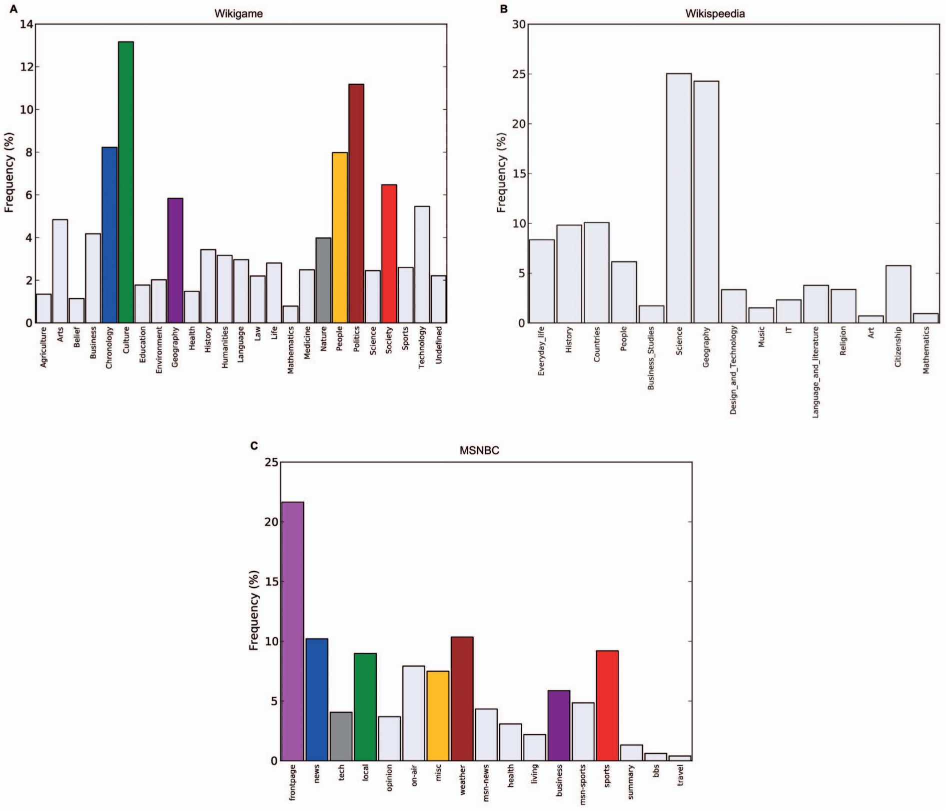


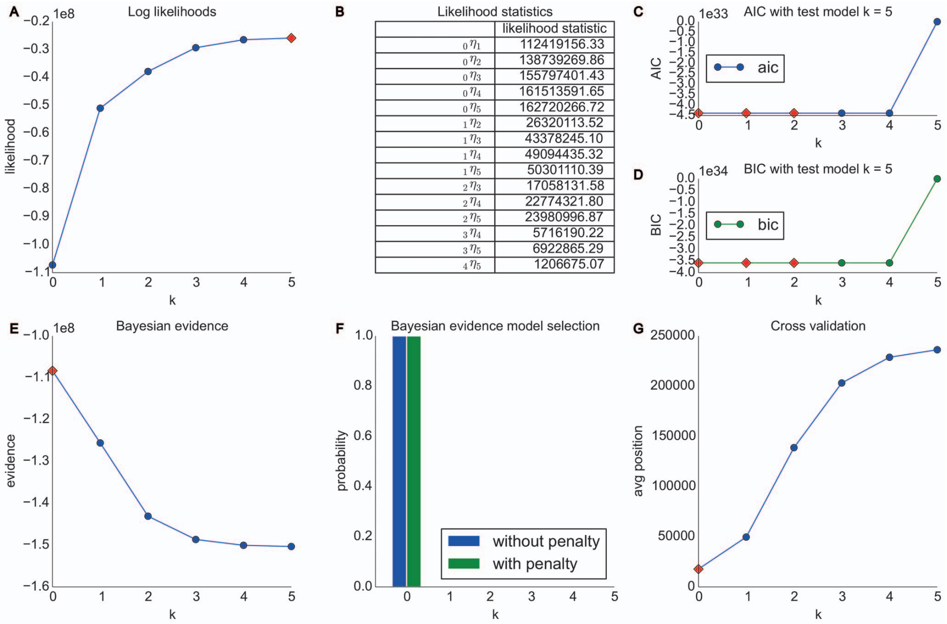
Figure 3. Topic frequencies. Frequency of categories (in percent) of all paths in (A) the Wikigame topic dataset (B) the Wikispeedia dataset and (C) the MSNBC dataset. The colors indicate the categories we will investigate in detail later and are representative for a single dataset – this means that the same color in the datasets does not represent the same topic. The Wikigame topic dataset consists of more distinct categories than the Wikispeedia and MSNBC dataset. Furthermore, the most frequently occuring topic in the Wikigame topic dataset is Culture with around 13%. The Wikispeedia dataset is dominated by the two categories the most Science and Geography each making up for almost 25% of all clicks. Finally, the most frequent topic in the MSNBC dataset is the frontpage with a frequency of around 22%.
doi:10.1371/journal.pone.0102070.g003

First, we can imply already from the likelihood statistics (B) that there might be no improvement over the most basic zero order Markov chain model as we can not find any statistically significant improvements of higher orders. Both AIC (C) and BIC (D) results confirm these observations and also agree with each other. Even though we can see equally low values for a zero, first and second order Markov chain, we would most likely prefer the most simple model in such a case – further following the ideas of the Occam's razor.

In order to extend these primary observations we used a uniform Laplace prior and Bayesian inference and henceforth, we obtain the results illustrated in the first two figures of the bottom row in Figure 4. The Bayesian inference results again suggest a zero order Markov chain model as the most appropriate as indicated by the highest evidence (E) and the highest probability obtained using Bayesian model selection with and without a further exponential penalty for the number of parameters (F).

The observations and preference of using a zero order model are finally confirmed by the results obtained from using 10-fold cross-validation and a prediction task (G). We can see that the average position is the lowest for a zero order model approving our observations made above.

Summary. Our analysis of the Wikigame page dataset thereby reveals a clear trend towards a zero order Markov chain model. This is imminent when looking at all distinct model selection techniques introduced and applied in this article, as they all agree on the choice of weighted random selection as the statistically significant most approvable model. This is a strong approval of our initial hypothesis stating it is highly difficult to make plausible statements about the appropriate Markov chain



	likelihood statistic
${}_0\eta_1$	112419156.33
${}_0\eta_2$	138739269.86
${}_0\eta_3$	155797401.43
${}_0\eta_4$	161513591.65
${}_0\eta_5$	162720266.72
${}_1\eta_2$	26320113.52
${}_1\eta_3$	43378245.10
${}_1\eta_4$	49094435.32
${}_1\eta_5$	50301110.39
${}_2\eta_3$	17058131.58
${}_2\eta_4$	22774321.80
${}_2\eta_5$	23980996.87
${}_3\eta_4$	5716190.22
${}_3\eta_5$	6922865.29
${}_4\eta_5$	1206675.07

Figure 4. Model selection results for the Wikigame page dataset. The top row shows results obtained using likelihood and information theoretic results: (A) likelihoods, (B) likelihood ratio statistics (* statistically significant at the 1% level; ** statistically significant at the 0.1% level) as well as AIC (C) and BIC (D) statistics. The bottom row illustrates results obtained from Bayesian Inference: (E) evidence and (F) Bayesian model selection. Finally, the figure presents the results from (G) cross validation. The overall results suggest a zero order Markov chain model.
doi:10.1371/journal.pone.0102070.g004

order having insufficient data but a vast amount of states. The higher performance of higher order chains can not compensate the necessary additional complexity in terms of statistically significant improvements. However, this may be purely an effect of the data sparsity in our investigation (i.e., the limited number of observations compared to the huge amount of distinct states). One can argue that real human navigation always can be better modeled by at least an order of one, because – as soon as we have enough data – links play a vital role in human navigation as humans by definition follow links when they navigate – except for teleportation which we do not model in this work. Consequently, we believe that the memoryless Markov chain model is a plausible model for human navigation on a page level. Yet, further detailed studies are necessary to confirm this.

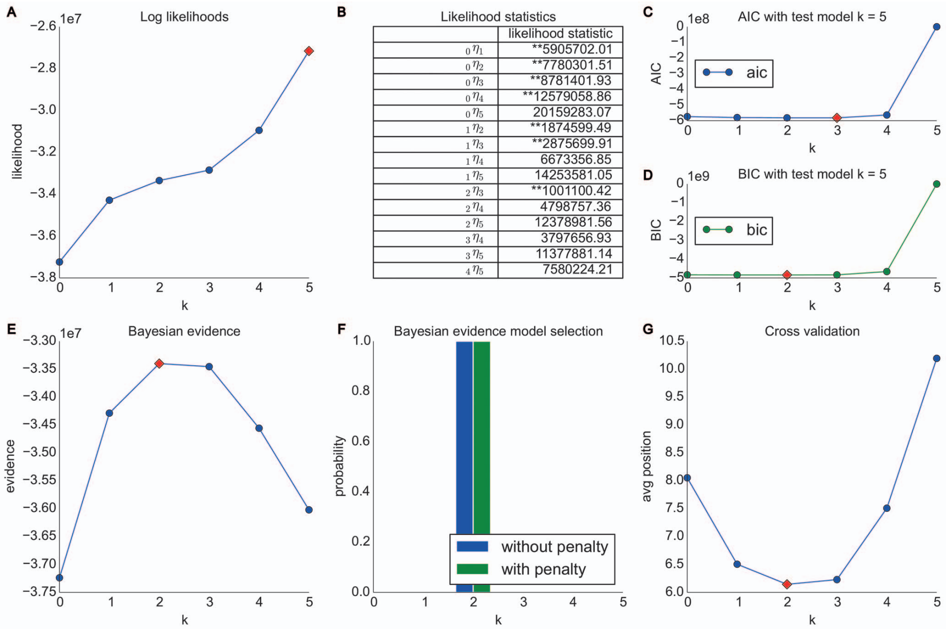
At the same time, one could argue that memory is best studied on a topical level, where pages are represented by topics. Consequently, we focus on studying transitions between topics next, which yields a reduced state space that allows analysis of the memory and structure of human navigation patterns on a topical level.

Topics navigation

Wikigame topic dataset. Performing our analyses by representing Wikipedia pages by their topical categories shows a much clearer and more interesting picture as one can see in Figure 5. Similar to above we can see (A) that the log likelihoods are rising with higher orders. However, in contrast to the Wikigame page dataset, we can now see (B) that several higher order Markov chain models are significantly better than lower orders. In detail, we can see that the appropriate Markov chain order is at least of order one and we can also observe a trend towards an order of two or three. Nevertheless, as pointed out in the section entitled "Likelihood Method", it is hard to concretely suggest one specific Markov chain order from these pairwise comparisons which is why we resort to this extended repertoire of model selection techniques described next.

The AIC (C) and BIC (D) statistics show further indicators – even though they are disagreeing – that the appropriate model is of higher order. Concretely, the suggest an order of three or two respectively by exhibiting the lowest values at these points. Not surprisingly, AIC suggests a higher order compared to BIC as the latter model selection method additionally penalized higher orders by the number of observations as stated in the section called "Information-theoretic Methods".

The Bayesian inference investigations (E, F) exhibit a clear trend towards a Markov chain of order two. The results in (F) nicely illustrate the inherent Occam's razor of the Bayesian model selection method as both priors – (a) no penalty and (b) exponential penalty for higher orders – suggest the same order (both priors agree throughout all our investigations in this article). Finally, the cross validation results (G) confirm that a second order



	likelihood statistic
${}_0\eta_1$	**5905702.01
${}_0\eta_2$	**7780301.51
${}_0\eta_3$	**8781401.93
${}_0\eta_4$	**12579058.86
${}_0\eta_5$	20159283.07
${}_1\eta_2$	**1874599.49
${}_1\eta_3$	**2875699.91
${}_1\eta_4$	6673356.85
${}_1\eta_5$	14253581.05
${}_2\eta_3$	**1001100.42
${}_2\eta_4$	4798757.36
${}_2\eta_5$	12378981.56
${}_3\eta_4$	3797656.93
${}_3\eta_5$	11377881.14
${}_4\eta_5$	7580224.21

Figure 5. Model selection results for the Wikigame topic dataset. The top row shows results obtained using likelihood and information theoretic results: (A) likelihoods, (B) likelihood ratio statistics (* statistically significant at the 1% level; ** statistically significant at the 0.1% level) as well as AIC (C) and BIC (D) statistics. The bottom row illustrates results obtained from Bayesian Inference: (E) shows evidence and (F) Bayesian model selection. (G) presents the results from cross validation. The overall results suggest that higher order chains seem to be more appropriate for our navigation paths consisting of topics. In detail, we find that a second order Markov chain model for our Wikigame topic dataset best explains the data.
doi:10.1371/journal.pone.0102070.g005

Markov chain produces the best results, while a third order model is nearly as good.

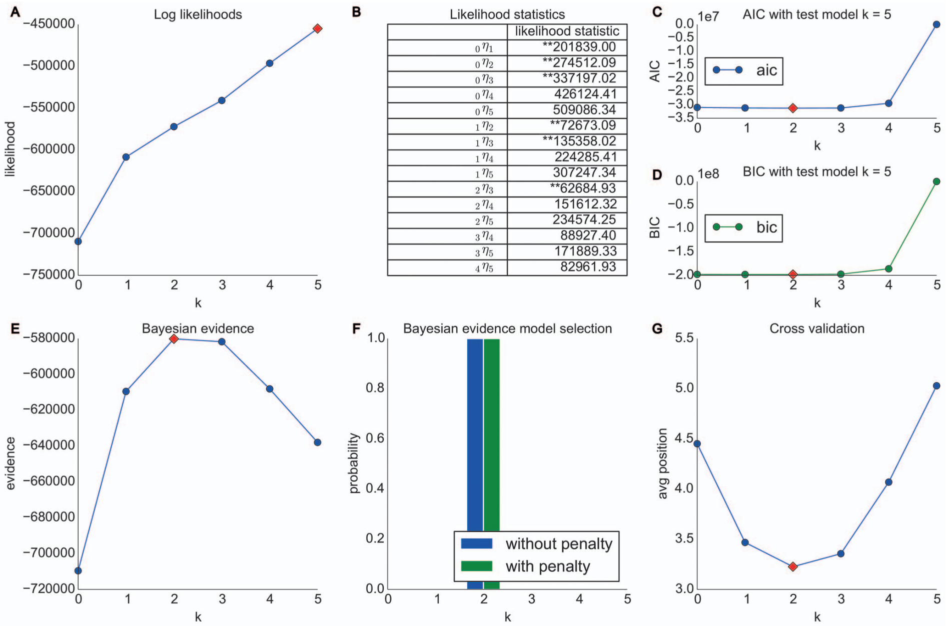
Summary. Overall, we can see that representing Wikigame paths as navigational sequences of corresponding topics leads to more interesting results: Higher order Markov chains exhibit statistically significant improvements, thereby suggesting that memory effects are at play. Overall, we can suggest that a second order Markov chain model seems to be the most appropriate for modeling the corresponding data as it gets suggested by all methods except for AIC which is known for slightly overestimating the order. This means, that humans remember their topical browsing patterns – in other words, the next click in navigational trails is dependent on the previous two clicks on a topical level.

Wikispeedia dataset. This section presents the results obtained from the Wikispeedia dataset introduced in the section entitled "Materials". Similar to the Wikigame topic dataset we look at navigational paths over topical categories in Wikipedia and present the results in Figure 6. Again we can observe that the likelihood statistics suggest higher order Markov chains to be appropriate (B). Yet, further analyses are necessary for a clear choice of the appropriate order. The AIC (C) and BIC (D) statistics agree to prefer a second order model; however, we need to note that all orders from zero to four have similarly low values. The Bayesian inference investigations (E, F) show a much clearer trend towards a second order model. The prediction results (G) agree on these observations by also showing the best results for a second order model. This time we can also observe a clear consilience between the cross validation and AIC results which are – as described in the section called "Information-theoretic Methods" – asymptotically equivalent.

Summary. This dataset is similar to the Wikigame topic dataset and the results are comparable to the previous results on the first goal-oriented dataset (Wikigame topic). Hence, even though the game is played on a much smaller set of Wikipedia articles and also the dataset consists of distinct categories, we can see the exact same behavior which strongly indicates that human navigation is not memoryless on a topical level and can be best modeled by a second order Markov chain model. This strongly suggests that humans follow common topical strategies while navigating in a goal-oriented scenario.

MSNBC dataset. In this section we present the results obtained from the MSNBC dataset introduced in the section called "Materials". Again we look at navigational paths over topical categories and henceforth, we only look at categorical information of nodes and present the results in Figure 7.

Similar to the experiments conducted for the Wikigame and Wikispeedia topic datasets we can again see, based on the likelihood ratio statistics (B), that a higher order Markov chain



	likelihood statistic
${}_0\eta_1$	**201839.00
${}_0\eta_2$	**274512.09
${}_0\eta_3$	**337197.02
${}_0\eta_4$	426124.41
${}_0\eta_5$	509086.34
${}_1\eta_2$	**72673.09
${}_1\eta_3$	**135358.02
${}_1\eta_4$	224285.41
${}_1\eta_5$	307247.34
${}_2\eta_3$	**62684.93
${}_2\eta_4$	151612.32
${}_2\eta_5$	234574.25
${}_3\eta_4$	88927.40
${}_3\eta_5$	171889.33
${}_4\eta_5$	82961.93

Figure 6. Model selection results for the Wikispeedia dataset. The top row shows results obtained using likelihood and information theoretic results: (A) likelihoods, (B) likelihood ratio statistics (* statistically significant at the 1% level; ** statistically significant at the 0.1% level) as well as AIC (C) and BIC (D) statistics. The bottom row illustrates results obtained from Bayesian Inference: (E) shows evidence and (F) Bayesian model selection. (G) presents the results from cross validation. The overall results suggest that higher order chains seem to be more appropriate for our navigation paths consisting of topics. Concretely, we find that a second order Markov chain model for our Wikispeedia topic dataset best explains the data.
doi:10.1371/journal.pone.0102070.g006

seems to be appropriate. The AIC (C) and BIC (D) statistics suggest an order of three and two respectively. To further investigate the behavior we illustrate the Bayesian inference results (E, F) that clearly suggest a third order Markov chain model. Finally, this is also confirmed by the cross validation prediction results (G) which again is in accordance with the AIC.

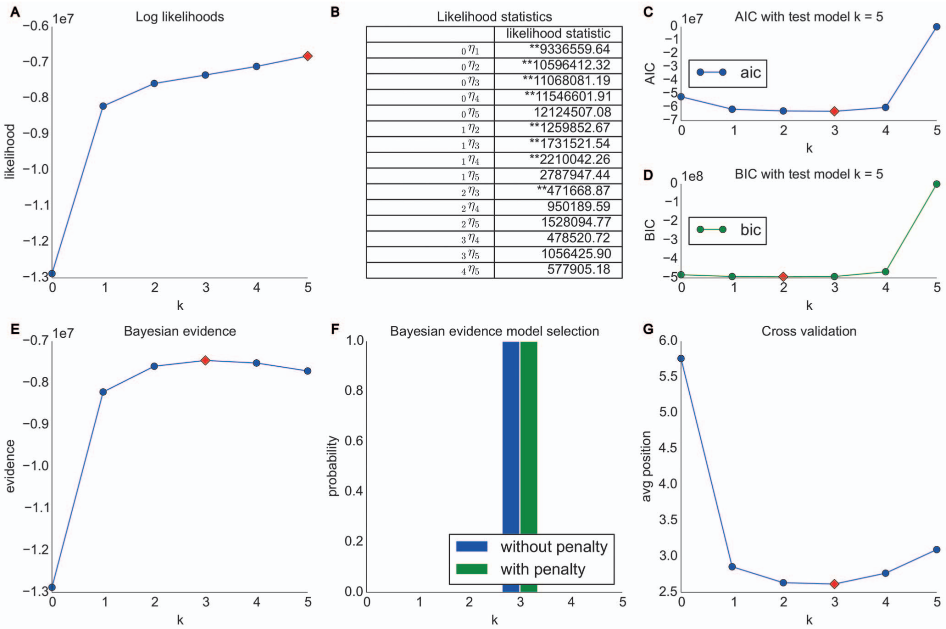
Summary. By and large, almost all methods for order selection suggest a Markov chain of order three for the topic sequence in the MSNBC dataset. Again, we can observe that the navigational patterns are not memoryless. Even though this dataset is not a goal-oriented navigation dataset, but is based on free navigation on MSNBC, we can identify similar memory effects as above.

Structure

In the previous section we observed memory patterns in human navigation over topics in information networks. We are now interested in digging deeper into the structure of human navigational patterns on a topical level. Concretely, we are interested in detecting common navigational sequences and in investigating structural differences between goal-oriented and free form navigation.

First, we want to get a global picture of common transition patterns for each of the datasets. We start with the Markov chain transition matrices, but instead of normalizing the row vectors, we normalize each cell by the complete number of transitions in the dataset. We illustrate these matrices as heatmaps to get insights into the most common transitions in the complete datasets. Due to tractability, we focus on a first order analysis and will focus on higher order patterns later on.

The heatmaps are illustrated in Figure 8. Predominantly, we can observe that self transitions seem to be very common as we can see from the high transition counts in the diagonals of the matrices. This means, that users regularly seem to stay in the same topic while they navigate the Web. Consequently, we might get better representations of the data by using Markov chain models that, instead modeling state transitions in equal time steps, additionally stochastically model the duration times in states (e.g., semi Markov or Markov renewal models). However, we leave these investigations open for future work. For the Wikigame (A) we can observe that the categories *Culture* and *Politics* are the most visited topics throughout the navigational paths. Most of the time the navigational paths start with a page belonging to the *People* topic which is visible by the dark red cell from *RESET* to *People* (remember that the *RESET* state marks both the start and end of a path - see Section "Materials"). However, as this is a game-based goal-oriented navigation scenario, the start node is always predefined. In our second goal-oriented navigation dataset (B) we can see that the paths are dominated by transitions from and to the categories *Science* and *Geography* and there are fewer transitions



	likelihood statistic
${}_0\eta_1$	**9336559.64
${}_0\eta_2$	**10596412.32
${}_0\eta_3$	**11068081.19
${}_0\eta_4$	**11546601.91
${}_0\eta_5$	12124507.08
${}_1\eta_2$	**1259852.67
${}_1\eta_3$	**1731521.54
${}_1\eta_4$	**2210042.26
${}_1\eta_5$	2787947.44
${}_2\eta_3$	**471668.87
${}_2\eta_4$	950189.59
${}_2\eta_5$	1528094.77
${}_3\eta_4$	478520.72
${}_3\eta_5$	1056425.90
${}_4\eta_5$	577905.18

Figure 7. Model selection results for the MSNBC dataset. The top row shows results obtained using likelihood and information theoretic results: (A) likelihoods, (B) likelihood ratio statistics (* statistically significant at the 1% level; ** statistically significant at the 0.1% level) as well as AIC (C) and BIC (D) statistics. The bottom row illustrates results obtained from Bayesian Inference: (E) shows evidence and (F) Bayesian model selection. (G) presents the results from cross validation. The overall results suggest that higher order chains seem to be more appropriate for our navigation paths consisting of topics. Specifically, the results suggest a third order Markov chain model.
doi:10.1371/journal.pone.0102070.g007

between other topics. In our MSNBC dataset (C) we can observe that most of the time users remain in the same topic while they navigate and globally no topic changes are dominant. This may be an artifact of the free navigation users practice on MSNBC. Perhaps unsurprisingly, users start with the frontpage most of the time while navigating but do not necessarily come back to it in the end.

As we have now identified global navigational patterns on the first order transition matrices we turn our attention to models of higher order. Furthermore, we are now interested in investigating

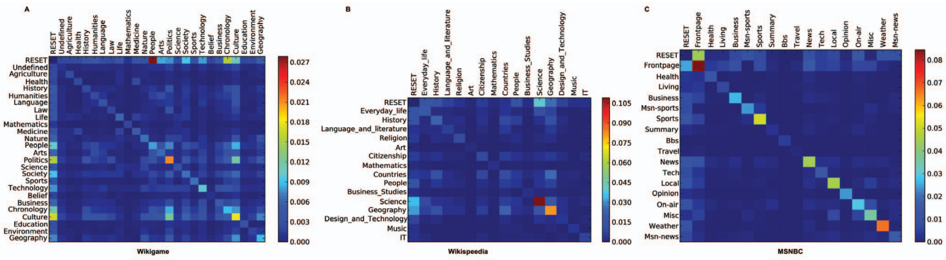


Figure 8. Global structure of human navigation. Common transition patterns of navigational behavior on all three topics datasets (Wikigame, Wikispeedia and MSNBC). Patterns are illustrated by heatmaps calculated on the first order transition matrices. Each cell is normalized by the total number of transitions in the dataset. The vertical lines depict starting states and the horizontal lines depict target states. A main observation is that self transitions – e.g., a transition from *Culture* to *Culture* – are dominating all datasets. However, the goal-oriented datasets (Wikigame and Wikispeedia) exhibit more transitions between distinct categories than the free navigation dataset (MSNBC).
doi:10.1371/journal.pone.0102070.g008

local transition probabilities – e.g., being at topic *Science*, what are the transition probabilities to other states. The transition weights directly correspond to the transition probabilities from the source to the target state determined by the MLE (see the section called "Likelihood Method"). We illustrate these local transitional patterns for our Wikigame dataset in Figure 9 (the investigations on the other goal-oriented Wikispeedia dataset exhibit similar patterns, but are omitted due to space limitations). Similar to the observations in Figure 8 we can observe that *Culture* is the most visited topic in our Wikigame dataset. We can now also identify specific prominent topical transition trails. For example, users seem to navigate between *Culture* and *Politics* quite frequently and also vice versa. Contrary, there seem to be specific unidirectional patterns too, e.g., users frequently navigate from *People* to *Politics* but not vice versa. Higher order chains also show similar structure, but on a more detailed level. As previously, the figure also depicts that the vast amount of transitions is between same categories. However, we can now observe that this is also the case for higher order Markov chains – this suggests, that the probability that users stay in the same topic increases with each new click on that topic.

To further look into this structural pattern, we illustrate the number of times users stay within the same topic vs. the number of times they change the topic during navigation in Figure 10. We can see that the longer the history – i.e., the higher the order of the Markov chain – the more likely people tend to stay in the same topic instead of switching to another topic. We can also see differences regarding this behavior between distinct categories; e.g., users are more likely to stay in the topic *Chronology* than in the topic *Politics* the higher the order is. For our Wikispeedia dataset we can observe similar patterns – i.e., the higher the order the higher the chance to stay in the same topic.

In order to contrast goal-oriented and free-form navigation, we also depict state transitions in similar fashion derived from the MSNBC dataset in Figure 11. In this figure we can see that the

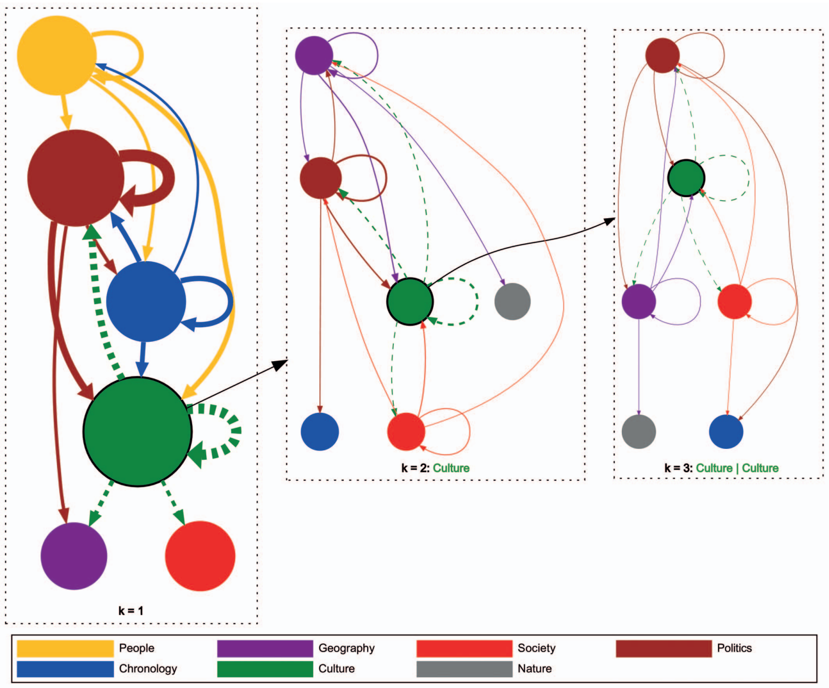


Figure 9. Local structure of navigation for the Wikigame topic dataset. The graphs above illustrate selected state transitions from the Wikigame topic dataset for different k values. The nodes represent categories and the links illustrate transitions between categories. The link weight corresponds to the transition probability from the source to the target node determined by MLE. The node size corresponds to the sum of the incoming transition probabilities from all other nodes to that source node. In the left figure the top four categories with the highest incoming transition probabilities are illustrated for an order of $k = 1$. For those nodes we draw the four highest outgoing transition probabilities to other nodes. In the middle figure we visualize the Markov chain of order $k = 2$ by setting the top topic (*Culture*) as the first click; this diagram shows transition probabilities from top four categories given that users first visited the *Culture* topic. For example, the links from the red node (*Society*) in the bottom-right part of the diagram represent the transition probabilities from the sequence (*Culture, Society*). Similarly, we visualize order $k = 3$ in the right figure by selecting a node with the highest incoming probability (*Culture, Culture*) of order $k = 2$. We then show transition probabilities from other nodes given that users already visited (*Culture, Culture*). For example, the links from the brown node (*Politics*) at the top represent the transition probabilities from the sequence (*Culture, Culture, Politics*).
doi:10.1371/journal.pone.0102070.g009

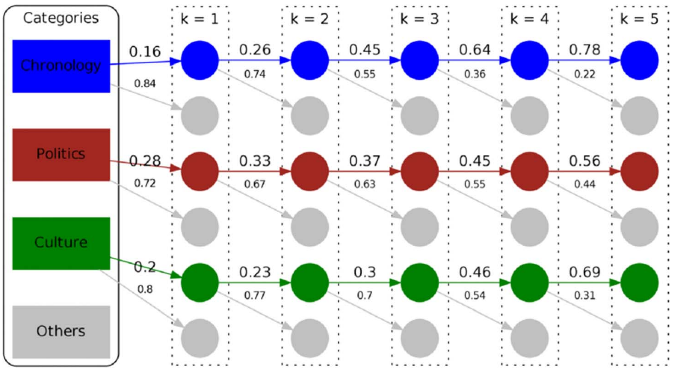


Figure 10. Self transition structure of navigation for the Wikigame topic dataset. The number of times users stay within the same topic vs. the number of times they change the topic during navigation for different orders k for our Wikigame dataset. Only the top three categories with the highest transition probabilities are shown. With high consistency, the transition probabilities to the same topic increase while those to other categories decrease with ascending order k.
doi:10.1371/journal.pone.0102070.g010

topic *business* is the most used. To give a navigational example: users frequently navigate from *business* to *news* and vice versa. However, there are also navigational patterns just going one direction. For example, users seem to frequently navigate from *business* to *sports* but not in the opposite direction. Again, higher order chains show similar patterns. Like in the Wikigame topic dataset we can as well observe that most of the transitions seem to be between similar categories. In Figure 12 we depict the number of times a user stays in the same topic vs. the number of times she switches the topic for the categories with the highest transition probabilities. We can again observe that the higher the Markov chain the more likely people tend to stay in the same topic while navigating. Nevertheless, an interesting difference to the Wikigame topic dataset can be observed. Concretely, we can see that the probability of staying in the same topic is much higher for the MSNBC dataset. Especially, the topic *weather* exhibits a very high probability of staying in the same topic (0.9 for $k=1$). A possible explanation is that users navigate on a semantically more narrow path on MSNBC. If you are interested about the weather you just check the specific pages on MSNBC while on Wikipedia you might get distracted by different categories at a higher probability. So these concrete observations seem to be very specific for the Web site and domains of the site users navigate on while the general patterns seem to be applicable for both of our datasets at hand.

Discussion

Our findings and observations in this article show that simple likelihood investigations (see e.g., [12]) may not be sufficient to select the appropriate order of Markov chains and to prove or falsify whether human navigation is memoryless or not. To ultimately answer this, we think it is inevitable to look deeper into the results obtained and to investigate them with a broader spectrum of model selection methods starting with the ones presented in this work.

By applying these methods to human navigational data, the results suggest that on the Wikigame page dataset a zero order model should be preferred. This is due to the rising complexity of higher order models and indicates that it is difficult to derive the appropriate order for finite datasets with a huge amount of distinct pages having only limited observations of human navigational behavior. In this article we presented and applied a variety of distinct model selection that all include (necessary) ways of penalizing the large number of parameters needed for higher order models. Yet, we do not necessarily know what would happen if we would apply the models to a much larger number of navigational paths over pages. Perhaps higher order models would then outperform lower ones. As it is unlikely to get hands on such an amount of data for large websites, a starting point to further test this could be to analyze a sub-domain with rich data; i.e., a large number of observations over just a very limited number of distinct pages. However, due to no current access to such data, we leave this open for future work.

On the other hand, the results on a topical level are intriguing and show a much clearer picture: They suggest that the navigational patterns are not memoryless. Higher order Markov chains – i.e., second or third order – seem to be the most appropriate. Henceforth, the navigation history of users seem to span at least two or three states on a topical level. This gives high indications that common strategies (at least on a topical level) exist among users navigating information networks on the Web. It is certainly intriguing to see similar memory patterns in both goal-oriented navigation (Wikigame and Wikispeedia) and free form navigation (MSNBC), and different kinds of systems (encylopedia vs. news portal).

In order to confirm that these observed memory effects are based on the actual human navigation patterns we again look at our random path dataset introduced in the section entitled "Likelihood Method" with the log-likelihoods visualized in Figure 2. We can recapitalize, that these simple log-likelihoods would suggest a higher order model for the randomly produced navigational patterns. However, if we apply our various model selection techniques the results suggest a zero or at maximum a first order Markov chain model which is the logic conclusion for this random process. Hence, this confirms that our observations on the real nature navigational data are based on human navigational memory patterns and would not be present in a random process.

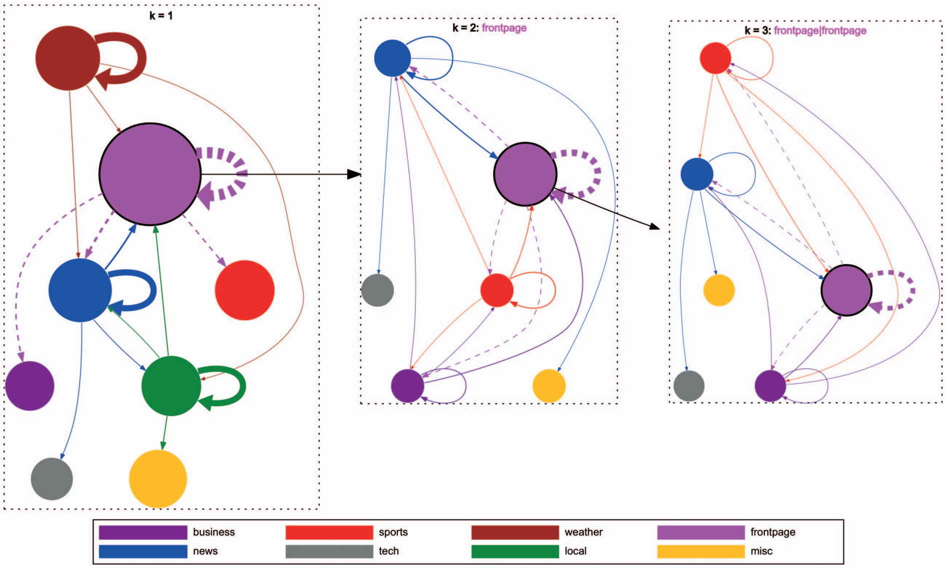


Figure 11. Local structure of navigation for the MSNBC dataset. The graphs above illustrate selected state transitions from the MSNBC dataset for different k values. The nodes represent categories and the links illustrate transitions between categories. The link weight corresponds to the transition probability from the source to the target node determined by MLE. The node size represents the global importance of a node in the whole dataset and corresponds to the sum of the outgoing transition probabilities from that node to all other nodes. For visualization reasons we primarily focus on the top four categories with the highest sum of outgoing transition probabilities – i.e., those with the largest node sizes – for an order of $k = 1$. For those nodes we draw the four highest outgoing transition probabilities to other nodes. In the middle figure we visualize the Markov chain of order $k = 2$ by setting the top topic (frontpage) from order $k = 1$ as the first click; this diagram shows transition probabilities from top four categories given that users first visited the frontpage topic (represented by the dashed transitions in the left figure representing $k = 1$). For example, the links from the blue node (news) in the top-left corner of the diagram represent the transition probabilities from the sequence (frontpage, news) to other nodes. Similarly, we visualize order $k = 3$ in the right figure by selecting a node with the highest sum of outgoing transition probabilities (frontpage, frontpage) and its four highest outgoing transition probabilities from order $k = 2$ (represented by the dashed transitions in the middle figure representing $k = 2$). We then show transition probabilities from other nodes given that users already visited (frontpage, frontpage). For example, the links from the red node (sports) at the top represent the transition probabilities from the sequence (frontpage, frontpage, sports) to other nodes.
doi:10.1371/journal.pone.0102070.g011

Finally, we showed in the section called "Structure" that common structure in the navigational trails exist among many users – i.e., common sequences of navigational transitions. First of all, we could observe that transitions between the same topic are common among all three datasets. However, they occur more frequently in our free form navigational data (MSNBC) than in the goal-oriented navigation datasets (Wikigame and Wikispeedia). Furthermore, users also seem to be more likely to stay longer in the same topic while navigating MSNBC while they seem to switch categories more frequently in both the Wikigame and Wikispeedia datasets. A possible explanation for this user behavior might be that users on MSNBC are more driven by specific information needs regarding one topic. For example, a user might visit the website to get information about the weather only. Contrary, exact information goals on Wikipedia might not always be in the same topic. Suppose, you are located on *Seoul* which belongs to the *Geography* topic and you want to know more about important inventions made in *Seoul*. A possible path then could be that you navigate over a *People* topic page and finally reach a *Science* topic page. However, we need to keep in mind that our goal-oriented datasets are based on game data with predefined start and target nodes. This means, that if the target nodes regularly lie in distinct categories, the user might be forced to switch categories more frequently. To rule this out, we illustrate the heatmap of our Wikigame dataset (cf. Figure 8) again by splitting the path corpus into two parts (see Figure 13): (A) only considering paths where the start and target node lie in the same topic and (B) only taking paths with distinct start and target categories. If the bias of given start and target nodes would influence our observations for specific structural properties of goal-oriented navigational patterns, Figure 13 would show strong dissimilarities between both illustrations which is not the case. Hence, we can state with strong confidence that the differences between goal-oriented and free form navigation stated in this section are truly based on the distinct strategies and navigational scenarios. Nevertheless, we also need to keep in mind that the website design and inherent link structure (Wikipedia vs. MSNBC) might also influence this behavior. For example, a reason could be that Wikipedia has more direct links between distinct categories in comparison to MSNBC or that Wikipedia's historical coverage steers user

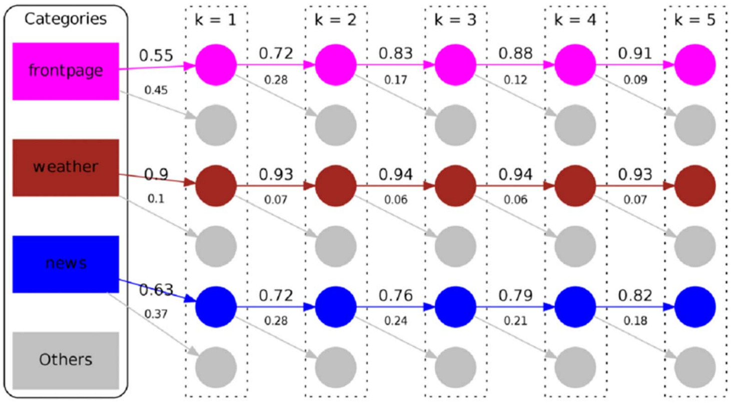


Figure 12. Self transition structure of navigation for the MSNBC dataset. The number of times users stay within the same topic vs. the number of times they change the topic during navigation for different values of k. Only the top three categories with the highest transition probabilities are shown. With high consistency, the transition probabilities to the same topic increase while those to other categories decrease with ascending order k.
doi:10.1371/journal.pone.0102070.g012

behavior to specific kinds of navigational patterns. To explicitly rule this possibility out, we would need to investigate the underlying link networks in greater detail, which we leave open for future work. We also plan on looking at data capturing navigational paths over distinct platforms of the Web (e.g., from toolbar data) which may allow us to make even more generic statements about human navigation on the Web.

Conclusions

This work presented an extensive view on detecting memory and structure in human navigational patterns. We leveraged Markov chain models of varying order for detecting memory of human navigation and took a thorough look at structural properties of human navigation by investigating Markov chain transition matrices.

We developed an open source framework (https://github.com/psinger/PathTools) [20] for detecting memory of human navigational patterns by calculating the appropriate Markov chain order using four different, yet complementary, approaches (likelihood, Bayesian, information-theoretic and cross validation methods). In this article we thoroughly present each method and emphasize strengths, weaknesses and relations between them. By applying this framework to actual human navigational data we find that it is indeed difficult to make plausible statements about the appropriate

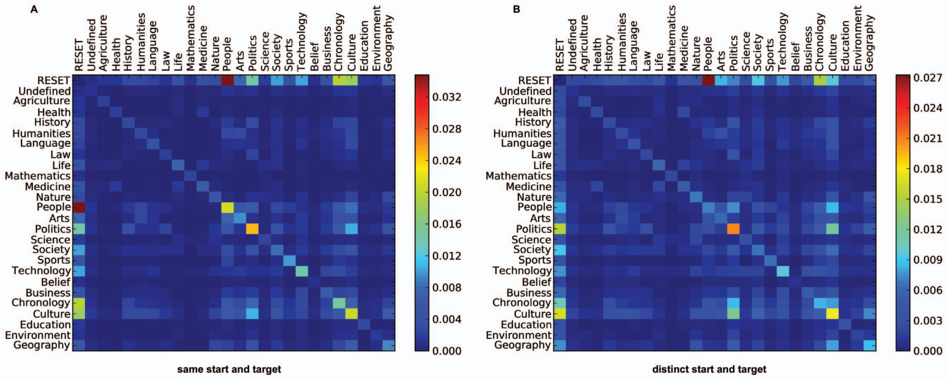


Figure 13. Common global transition patterns of navigational behavior on the Wikigame topic dataset. The results should be compare with Figure 8. The results are split by only looking at a corpus of paths where each path starts with the same topic as it ends (A) and by looking at a corpus with distinct start and target categories (B).
doi:10.1371/journal.pone.0102070.g013

order of a Markov chain having insufficient data but a vast amount of states which results in too complex models. However, by representing pages by their corresponding topic we could identify that navigation on a topical level is not memoryless – an order of two and respectively three best explain the observed data, independent whether the navigation is goal-oriented or free-form. Finally, our structural investigations illustrated that users tend to stay in the same topic while navigating. However, this is much more frequent for our free form navigational dataset (MSNBC) as compared to both of the goal-oriented datasets (Wikigame and Wikispeedia).

Future attempts of modeling human behavior in the Web can benefit from the methodological framework presented in this work to thoroughly investigate such behavior. If one wants to resort to a single model selection technique, we would recommend to use the Bayesian approach if computationally feasible.

Our work strongly indicates memory effects of human navigational patterns on a topical level. Such observations as well as detailed insights into structural regularities in human navigation patterns can e.g., be useful for improving recommendation systems, web site design as well as faceted browsing. In future work, we want to extend our ideas of representing Web pages with categories by looking at further features for representation. We also plan on tapping into the usefulness of further Markov models like the hidden Markov model, varying order Markov model or semi Markov model. Also, we want to improve recommendation algorithms by the insights generated in this work and explore the implications higher order Markov chain models may have on ranking algorithms like PageRank.

Acknowledgments

We want to thank Alex Clemesha (Wikigame) and Robert West (Wikispeedia) for granting us access to their navigational datasets as well as the anonymous reviewers for the highly valuable inputs.

Author Contributions

Conceived and designed the experiments: PS DH BT MS. Performed the experiments: PS DH BT MS. Analyzed the data: PS DH BT MS. Contributed reagents/materials/analysis tools: PS DH BT MS. Wrote the paper: PS DH BT MS.

References

1. Catledge LD, Pitkow JE (1995) Characterizing browsing strategies in the world-wide web. Computer Networks and ISDN systems 27: 1065–1073.
2. Xing D, Shen J (2004) Efficient data mining for web navigation patterns. Information and Software Technology 46: 55–63.
3. Borges J, Levene M (2000) Data mining of user navigation patterns. In: Web usage analysis and user profiling, Springer. pp. 92–112.
4. Bestavros A (1995) Using speculation to reduce server load and service time on the www. In: Proceedings of the fourth international conference on Information and knowledge management New York, NY, USAACM, CIKM ′95, pp. 403–410.
5. Perkowitz M, Etzioni O (1997) Adaptive web sites: an ai challenge. In: Proceedings of the 15th international joint conference on Artifical intelligence - Volume 1. San Francisco, CA, USAMorgan Kaufmann Publishers Inc., IJCAI ′97, pp. 16–21.
6. Rendle S, Freudenthaler C, Schmidt-Thieme L (2010) Factorizing personalized markov chains for nextbasket recommendation. In: Proceedings of the 19th international conference on World wide web New York, NY, USAACM, WWW ′10, pp. 811–820.
7. Huberman BA, Pirolli PL, Pitkow JE, Lukose RM (1998) Strong regularities in world wide web surfing. Science 280: 95–97.
8. Wang C, Huberman BA (2012) How random are online social interactions? Scientific reports 2.
9. Song C, Qu Z, Blumm N, Barabási AL (2010) Limits of predictability in human mobility. Science 327: 1018–1021.
10. Pirolli PLT, Pitkow JE (1999) Distributions of surfers' paths through the world wide web: Empirical characterizations. World Wide Web 2: 29–45.
11. Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. Computer networks and ISDN systems 30: 107–117.
12. Chierichetti F, Kumar R, Raghavan P, Sarlos T (2012) Are web users really markovian? In: Proceedings of the 21st international conference on World Wide Web New York, NY, USAACM, WWW ′12, pp. 09–618.
13. Stigler SM (2002) Statistics on the table: The history of statistical concepts and methods. Harvard University Press.
14. Tong H (1975) Determination of the order of a markov chain by akaike's information criterion. Journal of Applied Probability 12: 488–497.
15. Strelioff CC, Crutchfield JP, Hübler AW (2007) Inferring markov chains: Bayesian estimation, model comparison, entropy rate, and out-of-class modeling. Physical Review E 76: 011106.
16. Akaike H (1973) Information theory and an extension of the maximum likelihood principle. In: Second international symposium on information theory Akademinai Kiado, pp.267–281.
17. Katz RW (1981) On some criteria for estimating the order of a markov chain. Technometrics 23: 243–249.
18. Murphy KP (2002) Learning markov processes. The Encyclopedia of Cognitive Science.
19. Schwarz G (1978) Estimating the dimension of a model. The Annals of Statistics 6: 461–464.
20. Singer P (2014) Path tools. doi:10.5281/zenodo.10229. Available: http://dx.doi.org/10.5281/zenodo.10229
21. Huberman BA, Adamic LA (1998) Novelty and social search in the world wide web. CoRR cs.MA/9809025.
22. Borges J, Levene M (2007) Evaluating variable-length markov chain models for analysis of user web navigation sessions. IEEE Transactions on Knowledge and Data Engineering 19: 441–452.
23. Deshpande M, Karypis G (2004) Selective markov models for predicting web page accesses. ACM Transactions on Internet Technology 4: 163–184.
24. Lempel R, Moran S (2000) The stochastic approach for link-structure analysis (salsa) and the tkc effect. Computer Networks 33: 387–401.
25. Sen R, Hansen M (2003) Predicting a web user's next access based on log data. Journal of Computational Graphics and Statistics 12: 143–155.
26. Anderson CR, Domingos P,Weld DS (2001) Adaptive web navigation for wireless devices. In: Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence. pp. 79–884.
27. Cadez I, Heckerman D, Meek C, Smyth P, White S (2003) Model-based clustering and visualization of navigation patterns on a web site. Data Mining and Knowledge Discovery 7: 399–424.
28. Zukerman I, Albrecht DW, Nicholson AE (1999) Predicting users' requests on the www. Courses and Lectures-International Centre for Mechanical Sciences: 275–284.
29. Pitkow J, Pirolli P (1999) Mining longest repeating subsequences to predict world wide web surfing. In: Proceedings of the 2nd conference on USENIX Symposium on Internet Technologies and Systems – Volume 2. Berkeley, CA, USA: USENIX Association.
30. Rosvall M, Esquivel AV, Lancichinetti A, West JD, Lambiotte R (2013) Networks with memory. arXiv preprint arXiv:13054807.
31. Sarukkai RR (2000) Link prediction and path analysis using markov chains. Computer Networks 33: 377–386.
32. Bartlett M (1951) The frequency goodness of fit test for probability chains. In: Proceedings of the Cambridge Philosophical Society Cambridge Univ Press-volume 47, pp. 86–95.
33. Gates P, Tong H (1976) On markov chain modeling to some weather data. Journal of Applied Meteorology and Climatology 15: 1145–1151.
34. Kumar R, Tomkins A (2010) A characterization of online browsing behavior. In: Proceedings of the 19th international conference on World wide web New York, NY, USAACM, WWW ′10, pp. 561–570.
35. West R, Leskovec J (2012) Human Wayfinding in Information Networks. In: Proceedings of the 21st International Conference on World Wide Web New York, NY, USAACM, WWW ′12, pp. 619–628.
36. Royall R (1997) Statistical evidence: a likelihood paradigm, volume 71. CRC press.
37. Perneger TV, Courvoisier DS.
38. Morrison DE, Henkel RE (2006) The significance test controversy: A reader. Transaction Publishers.
39. Box GE, Tiao GC (2011) Bayesian inference in statistical analysis, volume 40. John Wiley & Sons.
40. Huelsenbeck J, Andolfatto P (2007) Inference of population structure under a dirichlet process model. Genetics 175: 1787–1802.
41. Kass RE, Raftery AE (1995) Bayes factors. Journal of the american statistical association 90: 773–795.
42. Durbin R, Eddy SR, Krogh A, Mitchison G (1998) Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press.
43. MacKay DJ (1992) Bayesian methods for adaptive models. Ph.D. thesis, California Institute of Technology.

44. Murray I, Ghahramani Z (2005) A note on the evidence and bayesian occam's razor.
45. MacKay DJ (2003) Information theory, inference and learning algorithms. Cambridge university press.
46. Burnham KP, Anderson DR (2002) Model selection and multimodel inference: a practical informationtheoretic approach. Springer.
47. Burnham KP, Anderson DR (2004) Multimodel inference understanding aic and bic in model selection. Sociological methods & research 33: 261–304.
48. Kullback S, Leibler RA (1951) On information and sufficiency. The Annals of Mathematical Statistics 22: 79–86.
49. Weakliem DL (1999) A critique of the bayesian information criterion for model selection. Sociological Methods & Research 27: 359–397.
50. Csiszár I, Shields PC (2000) The consistency of the bic markov order estimator. The Annals of Statistics 28: 1601–1619.
51. Baigorri A, Gonçalves C, Resende P (2009) Markov chain order estimation and relative entropy. arXiv preprint arXiv:09100264.
52. Stone M (1977) An asymptotic equivalence of choice of model by cross-validation and akaike's criterion. Journal of the Royal Statistical Society Series B (Methodological): 44–47.
53. Liben-Nowell D, Kleinberg J (2003) The link prediction problem for social networks. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management New York, NY, USA: ACM, CIKM '03, pp. 556–559. doi:10.1145/956863.956972. Available: http://doi.acm.org/10.1145/956863.956972
54. West R, Leskovec J (2012) Automatic versus human navigation in information networks. In: Breslin JG, Ellison NB, Shanahan JG, Tufekci Z, editors, ICWSM. The AAAI Press.
55. West R, Pineau J, Precup D (2009) Wikispeedia: An Online Game for Inferring Semantic Distances between Concepts. In: Proceedings of the 21st International Joint Conference on Artifical Intelligence San Francisco, CA, USAMorgan Kaufmann Publishers Inc., IJCAI '09, pp. 1598–1603.
56. Scaria AT, Philip RM, West R, Leskovec J (2014) The last click: Why users give up information network navigation.

3.3. Discovering Beaten Paths in Collaborative Ontology-Engineering Projects using Markov Chains

This article provides further answers to the first research question of this thesis. To that end, it demonstrates the general applicability of the Markov chain framework presented in Section 3.2 as well as highlights new mechanics offered. Specifically, in this article, colleagues and I have been interested in studying structural patterns in human trails on the Web. To that end, we have utilized transition matrices of fitted Markov chain models for detecting common patterns. This allows to gain insights into emerging behavioral patterns on the Web.

In detail, we have focused on first-order Markov chain models and corresponding structural patterns as well as on human edit trails in collaborative ontology engineering projects stemming from the biomedical domain. The framework could successfully elicit dominant structural patterns which further argues that certain aspects and strategies guide human behavior on the Web. While just an example, this demonstrates that the application of the Markov chain framework is not limited to human navigational trails. It can successfully be utilized for studying memory and structure in all kinds of human trails on the Web.

Discovering Beaten Paths in Collaborative Ontology-Engineering Projects using Markov Chains

Simon Walk[a,*], Philipp Singer[b], Markus Strohmaier[b,c], Tania Tudorache[d], Mark A. Musen[d], Natalya F. Noy[d]

[a] *Institute for Information Systems and Computer Media, Graz University of Technology, Austria*
[b] *GESIS - Leibniz Institute for the Social Sciences, Cologne, Germany*
[c] *Dept. of Computer Science, University of Koblenz-Landau, Germany*
[d] *Stanford Center for Biomedical Informatics Research, Stanford University, USA*

Abstract

Biomedical taxonomies, thesauri and ontologies in the form of the International Classification of Diseases as a taxonomy or the National Cancer Institute Thesaurus as an OWL-based ontology, play a critical role in acquiring, representing and processing information about human health. With increasing adoption and relevance, biomedical ontologies have also significantly increased in size. For example, the 11th revision of the International Classification of Diseases, which is currently under active development by the World Health Organization contains nearly 50,000 classes representing a vast variety of different diseases and causes of death. This evolution in terms of size was accompanied by an evolution in the way ontologies are engineered. Because no single individual has the expertise to develop such large-scale ontologies, ontology-engineering projects have evolved from small-scale efforts involving just a few domain experts to large-scale projects that require effective collaboration between dozens or even hundreds of experts, practitioners and other stakeholders. Understanding the way these different stakeholders collaborate will enable us to improve editing environments that support such collaborations. In this paper, we uncover how large ontology-engineering projects, such as the International Classification of Diseases in its 11th revision, unfold by analyzing usage logs of five different biomedical ontology-engineering projects of varying sizes and scopes using Markov chains. We discover intriguing interaction patterns (e.g., which properties users frequently change after specific given ones) that suggest that large collaborative ontology-engineering projects are governed by a few general principles that determine and drive development. From our analysis, we identify commonalities and differences between different projects that have implications for project managers, ontology editors, developers and contributors working on collaborative ontology-engineering projects and tools in the biomedical domain.

Keywords: Collaborative ontology engineering; Markov chains; sequential patterns; collaboration; ontology-engineering tool; user interface

1. Introduction

Today, biomedical ontologies play a critical role in acquiring, representing and processing information about human health. For example, the International Classification of Diseases (ICD) is a taxonomy that is used in more than 100 countries to encode patient diseases, to compile health-related statistics and to collect health-related spending statistics. Similarly, the National Cancer Institute's Thesaurus (NCIt) represents an important OWL-based vocabulary for classifying cancer and cancer-related terms.

With their increase in relevance, biomedical taxonomies, thesauri and ontologies have also significantly increased in size to cover new findings and to extend and complement their original areas of application. For example, the 11th revision of the International Classification of Diseases (ICD-11), currently under active development by the World Health Organization (WHO), consists of nearly 50,000 classes representing a vast variety of different diseases and causes of death. In contrast to previous revisions, the foundation component of ICD-11 is implemented as an OWL ontology with a broader scope than previous ICD revisions.

This growth was accompanied by a need to adapt the way these ontologies are engineered as no single individual or small group of domain experts have the expertise to develop such large-scale ontologies. New tools and processes have to be developed in order to coordinate, augment and manage collaboration between the dozens or hundreds of experts, practitioners and stakeholders when engineering an ontology.

Understanding the ways in which such a large number of participants – e.g., more than 100 experts contribute to ICD-11 – collaborate with one another when creating a structured knowledge representation is a prerequisite for quality control and effective tool support.

Objectives: Consequently, we aim at understanding how large collaborative ontology-engineering projects such as ICD-

[*] Corresponding author (`simon.walk@tugraz.at`)

11 unfold. In particular, we want to investigate if we can identify usage patterns in the change-logs of collaborative ontology-engineering projects? We approach this problem by analyzing patterns in usage logs of five biomedical ontology-engineering projects of varying sizes and scopes. For this analysis we employ Markov chain models for investigating and modeling sequential interaction paths (c.f. Section 3.2). Such paths are represented by chronologically ordered lists of interactions within the underlying ontology for (a) a single user or (b) a single class (see Figure 2). For example, we study sequences of properties that were either changed by (a) *a single user* on any class or (b) *a single class* by any user in an ontology over time. For example, as depicted in Figure 2, a sequential property path for a single user (user-based) consists of a chronologically ordered list of all properties (e.g., *title*, *definition* etc.), which have been changed by that user on any class, while a sequential property path for a single class (class-based) consists of a chronologically ordered list of properties that were changed on that class by any user. Instead of only modeling sequences for single users or classes, our data contains a set of paths; e.g., each path in the dataset consists of sequences of properties whose value has been changed by a single user over time. This allows us to tap into accumulated patterns. Concretely, we are interested in studying emerging patterns of subsequent steps in such sequential paths – e.g., which properties do users frequently change after a specific given property.

The analyzed datasets range from large-scale datasets such as ICD-11 to smaller ones such as the Ontology for Parasite Lifecycle (OPL). Given the differences of our datasets in a number of salient characteristics, we investigate if specific patterns can be found across all or only in certain biomedical ontology-engineering projects. Furthermore, we investigate and discuss features of these projects that potentially affect observed patterns, which can only be found in specific datasets. This analysis can be seen as a stepping stone for collaborative ontology-engineering project managers to devise infrastructures and tool support to augment collaborative ontology engineering.

Contributions: We present new insights on social interactions and editing patterns that suggest that large collaborative ontology-engineering projects are governed by a few general principles that determine and drive development. Specifically, our results indicate that general edit patterns can be found in all investigated datasets, even though they (i) represent different projects with different goals, (ii) use variations of the same ontology-editors and tools for the engineering process and (iii) differ in the way the projects are coordinated.

To the best of our knowledge, the work presented in this paper represents the most fine-grained and comprehensive study of patterns in large-scale collaborative ontology-engineering projects in the domain of biomedicine. In addition, our analysis is conducted across five datasets of different sizes, which have been developed using different versions of Collaborative Protégé (Table 1).

2. Collaborative ontology engineering

According to Gruber [1], Borst [2], Studer et al. [3] an ontology is an explicit specification of a shared conceptualization. In particular, this definition refers to a machine-readable construct (the formalization) that represents an abstraction of the real world (the shared conceptualization), which is especially important in the field of computer science as it allows a computer (among other things) to "understand" relationships between entities and objects that are modeled in an ontology.

Collaborative ontology engineering is a new field of research with many new problems, risks and challenges that we must first identify and then address. In general, contributors of collaborative ontology-engineering projects, similar to traditional collaborative online production systems[1] (e.g., Wikipedia), engage remotely (e.g., via the internet or a client–server architecture) in the development process to create and maintain an ontology. As an ontology represents a formalized and abstract representation of a specific domain, disagreements between authors on certain subjects can occur. Similar to face-to-face meetings, these collaborative ontology-engineering projects need tools that augment collaboration and help contributors in reaching consensus when modeling topics of the real world.

Indeed, the majority of the literature about collaborative ontology engineering sets its focus on surveying, finding and defining requirements for the tools used in these projects [4, 5].

The Semantic Web community has developed a number of tools aimed at supporting the collaborative development of ontologies. For example, Semantic MediaWikis [6] and its derivatives [7, 8, 9] add semantic, ontology modeling and collaborative features to traditional MediaWiki systems.

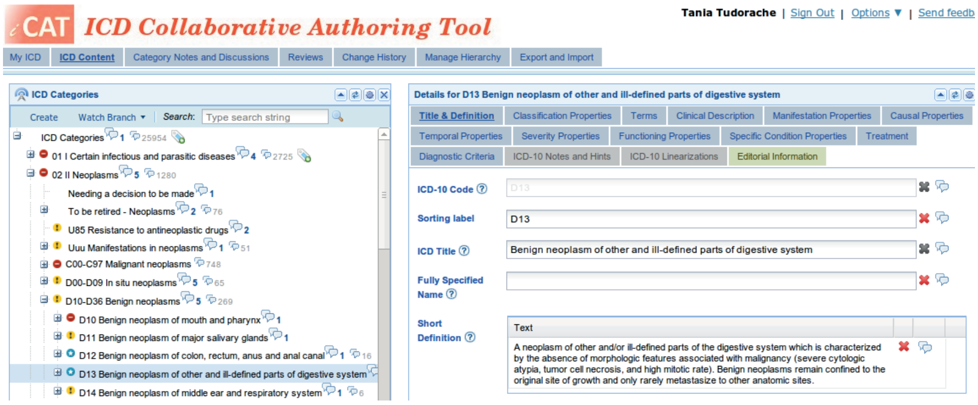
Protégé, and its extensions for collaborative development, such as WebProtégé and iCAT [10] (see Figure 1 for a screenshot of the iCAT ontology-editor interface) are prominent standalone tools that are used by a large community worldwide to develop ontologies in a variety of different projects. Both WebProtégé and Collaborative Protégé provide a robust and scalable environment for collaboration and are used in several large-scale projects, including the development of ICD-11 [11].

Pöschko et al. [12] and Walk et al. [13] have created *PragmatiX*, a tool to visualize and analyze a collaboratively engineered ontology and aspects of its history and the engineering process, providing quantitative insights into the ongoing collaborative development processes.

Falconer et al. [14] investigated the change-logs of collaborative ontology-engineering projects, showing that users exhibit specific roles, which can be used to group and classify users, when contributing to the ontology. Pesquita and Couto [15] investigated whether the location and specific structural features can be used to determine if and where the next change is going to occur in the Gene Ontology[2].

[1] Note that the term traditional online production systems refers to online platforms that have users collaborate in engineering digital goods, opposed to a structured knowledge base that is the result of collaborative ontology-engineering.

[2] http://www.geneontology.org



ICD Categories 1 25954
01 I Certain infectious and parasitic diseases 4 2725
02 II Neoplasms 5 1280
Needing a decision to be made 1
To be retired - Neoplasms 2 76
U85 Resistance to antineoplastic drugs 2
Uuu Manifestations in neoplasms 1 51
C00-C97 Malignant neoplasms 748
D00-D09 In situ neoplasms 5 65
D10-D36 Benign neoplasms 5 269
D10 Benign neoplasm of mouth and pharynx 1
D11 Benign neoplasm of major salivary glands 1
D12 Benign neoplasm of colon, rectum, anus and anal canal 1 16
D13 Benign neoplasm of other and ill-defined parts of digestive system
D14 Benign neoplasm of middle ear and respiratory system 1 6

Details for D13 Benign neoplasm of other and ill-defined parts of digestive system

ICD-10 Code: D13

Sorting label: D13

ICD Title: Benign neoplasm of other and ill-defined parts of digestive system

Fully Specified Name:

Short Definition:

Text: A neoplasm of other and/or ill-defined parts of the digestive system which is characterized by the absence of morphologic features associated with malignancy (severe cytologic atypia, tumor cell necrosis, and high mitotic rate). Benign neoplasms remain confined to the original site of growth and only rarely metastasize to other anatomic sites.

Figure 1: A screenshot of iCAT, a custom tailored, web-based version of WebProtégé, developed for the collaborative engineering of ICD-11. The left part of the interface visualizes the ICD-11 class hierarchy, the class titles, the number of annotations each class has received (speech bubbles) and its overall progress (color and symbol before the class title). The right part of the interface shows the different user-interface sections (e.g, *Title & Definition* or *Classification Properties*), listing all properties and property values for each class.

Goncalves et. al [16, 17, 18] performed an analysis of different versions of ontologies by applying and categorizing *Diff* algorithms, with the goal of categorizing the differences between consecutive and chronologically ordered versions of the ontologies. Furthermore, they conducted reasoner performance tests and identified factors that potentially increase reasoner performance. For the analysis presented in this paper we were able to rely on ChAO [19], which is a change-log provided by Protégé and its derivatives that already provides us with detailed and unambiguous logs of changes for the investigated ontologies.

In a similar context Grau et al. [20, 21] proposed a logical framework for modularity of ontologies and a definition of what is to be considered as an ontology module. In general, an ontology module can be used to extract the meaning of a specified set of terms from an ontology. Extracting the right amount of information is especially important for the topic of ontology reuse. According to Grau et al. modularity also represents a crucial factor in collaborative ontology-engineering environments as modular representations of ontologies are easier to understand, to extend and to reuse, similar to modularity in software engineering projects.

Mikroyannidi et al. [22] investigated the detection and use of (design) patterns in the content of an ontology, using a clustering approach. In contrast to Mikroyannidi et al., our analysis focuses on the detection of sequential patterns in interaction data rather than content.

Strohmaier et al. [23] investigated the hidden social dynamics that take place in collaborative ontology-engineering projects from the biomedical domain and provides new metrics to quantify various aspects of the collaborative engineering processes. Wang et al. [24] have used association-rule mining to analyze user editing patterns in collaborative ontology-engineering projects. The approach presented in this paper uses Markov chains to extract much more fine grained user-interaction patterns incorporating a variable number of historic editing information.

The only requirement to perform the pattern analysis that we present in this paper is the availability of a structured log of changes that can be mapped to the underlying ontology. The majority of the discussed collaborative ontology-engineering environments provide such a log, allowing for a similar analysis. For example, the Semantic MediaWikis store all the changes to the articles, and thus the ontology, allowing to expand the application of Markov chains to analyze sequential patterns as shown in this paper.

3. Materials & methods

For the analysis conducted in this paper we concentrated our efforts on five ontology-engineering projects in the biomedical domain. Each of the projects (i) has at least two users who contributed to the project, (ii) provides a structured log of changes and (iii) represents knowledge from the biomedical domain. In Section 3.1 we provide a brief history for each dataset and in Section 3.2 we describe the sequential path analysis. To aid readers in understanding the analyses conducted in this paper and its implications we provide a very brief overview of Markov chains and the involved model selection methodology in Section 3.3.

3.1. Datasets

Table 1 lists the detailed features and observation periods for the following five datasets that we used in our analysis. All datasets have been created either with WebProtégé or special

Table 1: Detailed information of the datasets used for the sequential pattern analysis to extract beaten paths in collaborative ontology-engineering projects.

		ICD-11	ICTM	NCIt	BRO	OPL
Ontology	classes	48,771	1,506	102,865	528	393
	changes	439,229	67,522	294,471	2,507	1,993
	DL expressivity	$\mathcal{SHOIN}(\mathbf{D})$	$\mathcal{SHOIN}(\mathbf{D})$	$\mathcal{SH}$	$\mathcal{SHIF}(\mathbf{D})$	$\mathcal{SHOIF}$
Editor	tool	iCAT	iCAT-TM	Collaborative Protégé	WebProtégé	Collaborative Protégé
Users	users	109	27	17	5	3
	bots (changes)	1 (935)	1 (1)	0 (0)	0 (0)	0 (0)
Duration	first change	18.11.2009	02.02.2011	01.06.2010	12.02.2010	09.06.2011
	last change	29.08.2013	17.7.2013	19.08.2013	06.03.2010	23.09.2011
	observation period (ca.)	4 years	2.5 years	3 years	1 month	3 months

versions of WebProtégé. To be able to conduct the pattern detection analysis for a different dataset, there is only one requirement that needs to be satisfied: The availability of a change-log that can be mapped onto the ontology so that changes can be associated with users and classes without ambiguity.

The DL expressivity [25, 26] of the five datasets is added to Table 1 to highlight that the investigated ontologies exhibit different strategies regarding their OWL-DL expressivity. As all levels of expressivity shown in Table 1 allow for the definition and assignment of properties and classes, they do not influence the conducted pattern detection analyses. Also, in the case of WebProtégé and its derivatives, the data used for the pattern detection analysis can be extracted from the change-logs, allowing us to prevent parsing and extracting values from OWL directly.

The International Classification of Diseases (ICD)[3] is the international standard for diagnostic classification used to encode information relevant to epidemiology, health management, and clinical use in over 100 United Nations countries. The World Health Organization (WHO) develops ICD, and publishes new revisions of the classification every decade or more. The current revision in use is ICD-10, a taxonomy that contains over 15,000 classes. The 11th revision of ICD,[4] **ICD-11**, is currently taking place and brings two major changes with respect to previous revisions. First, ICD-11's foundation component is developed as an OWL ontology using a much richer representation formalism than previous revisions. ICD-11 contains very detailed descriptions of several aspects of diseases, mostly represented as properties in the ontology. Second, the development of ICD-11 takes place in a Web-based collaborative environment, called iCAT (see Figure 1), which allows domain experts around the world to contribute and review the ontology online. ICD-11 is planned to be finalized in May 2017.

The International Classification of Traditional Medicine (ICTM) is a WHO led project that aimed to produce an international standard terminology and classification for diagnoses and interventions in Traditional Medicine.[5] ICTM, similarly to ICD-11, is implements an OWL based ontology as foundation component, which tries to unify the knowledge from the traditional medicine practices from China, Japan and Korea. Its content is authored in 4 languages: English, Chinese, Japanese and Korean. More than 20 domain experts from the three countries developed ICTM using a customized version of the iCAT system, called iCAT-TM. The development of ICTM was stopped in 2012, and a subset of ICTM is also included as a branch in the ICD-11 ontology.[6]

The National Cancer Institute's Thesaurus (NCIt) [27] has over 100,000 classes and has been in development for more than a decade. It is a reference vocabulary covering areas for clinical care, translational, basic research, and cancer biology. A multidisciplinary team of editors works to edit and update the terminology based on their respective areas of expertise, following a well-defined workflow. A lead editor reviews all changes made by the editors. The lead editor accepts or rejects the changes and publishes a new version of the NCI Thesaurus. The NCI Thesaurus is , at its core, an OWL ontology, which uses many OWL primitives such as defined classes and restrictions. It was named thesaurus due to historical reasons, however fully conforms to OWL semantics, thus represents an actual ontology.

The Biomedical Resource Ontology (BRO) originated in the Biositemaps project,[7] an initiative of the Biositemaps Working Group of the NIH National Centers for Biomedical Computing [28]. Biositemaps is a mechanism for researchers working in biomedicine to publish metadata about biomedical data, tools, and services. Applications can then aggregate this information for tasks such as semantic search. BRO is the enabling technology used in Biositemaps; a controlled terminology for describing the resource types, areas of research, and activity of a biomedical related resource. BRO was developed by a small group of editors, who use a Web-based interface (WebProtégé) to modify the ontology and to carry out discussions to reach consensus on their modeling choices.

The Ontology for Parasite Lifecycle (OPL) models the life cycle of the *T.cruzi*, a protozoan parasite, which is responsible for a number of human diseases. OPL is an OWL ontology that extends several other OWL ontologies. It uses many OWL constructs such as restrictions and defined classes. Several users from different institutions collaborate on OPL development. This ontology is much smaller and has far fewer users than NCIt, ICD-11, or ICTM.

[3] http://www.who.int/classifications/icd/en/

[4] http://www.who.int/classifications/icd/ICDRevision/

[5] http://tinyurl.com/ictmbulletin

[6] The ICD-11 dataset used in our analysis did not include the ICTM branch.

[7] http://biositemaps.ncbcs.org

3.2. Sequential interaction paths

For our sequential pattern analysis we analyze three different kinds of paths, which all represent interactions with the underlying ontology. A sequential path is represented by the chronologically ordered list of extracted interactions for either a single user or a single class (see Figure 2). For example, a sequential property path for a single user (user-based) consists of a chronologically ordered list of all properties (e.g., *title*, *definition* etc.), which have been changed by that user on any class, while a sequential property path for a single class (class-based) consists of a chronologically ordered list of properties that were changed on that class by any user.

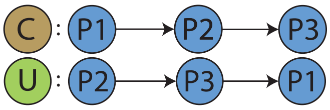


Figure 2: The top row of the figure depicts an exemplary **class-based** sequential property path ($P1$ to $P3$) for class C. This means that for class C the property $P1$ was changed first, then property $P2$ and most recently changed was property $P3$. The bottom row of the figure depicts the sequential property path ($P1$ to $P3$), however this time for a user U (**user-based**). Analogously, user U has first changed $P2$, continued to change property $P3$ and most recently changed $P1$.

User-sequence paths: First, we analyze activity patterns within the collaborative ontology-engineering project. This means that we analyze sequences of users who change a class. We want to detect and describe the different sequential patterns (the structure) that can be extracted from the change-logs of the investigated collaborative ontology-engineering projects.

Structural paths: Analogously to the User-Sequence Paths, we investigate edit-strategies, such as *bottom-up* or *top-down* development, that users follow. Is it possible to detect common patterns of which depth level a user frequently contributes to after a given current depth level? In addition to development-strategies, we look at the relationships (e.g., parent, child, sibling, etc.) between the current and the next class a user is going to contribute to.

Property paths: On a content-based level, we investigate the series of property-changes users perform on. In particular, we want to identify common successive property-changes – i.e., which properties *users* (user-based) regularly change consecutively and which properties are changed back-to-back for *classes* (class-based).

3.3. Markov chain models

For the analysis conducted in this paper we are adopting the methodology presented by Singer et al. [29] and mapped to collaborative ontology-engineering change logs by Walk et al. [30] to detect sequential patterns identified in and extracted from change-logs of collaborative ontology-engineering projects.

For a better understanding of the collected results, we will provide a short description of Markov chains. For an in-depth description of our methodology we point to Singer et al. [29], Walk et al. [30].

In general, Markov chain models are used for stochastically modeling transitions between states on a given state space. In our case, a Markov chain consists of a finite *state-space* (e.g., properties that a user edits over time; see Section 3.2) and the corresponding *transition probabilities* (e.g., the probability of changing property j after property i) between these states. Markov chain models are usually described as memoryless which means that the next state in a sequences only depends on the current one and not on a sequence of preceding ones (also known as Markovian property). Hence, this property defines serial dependence between adjacent nodes in trajectories – this is where the term "chain" comes from. Such a model is usually called a *first-order* or *memoryless* model.

As we are interested in modeling sequential interaction paths of collaborative ontology-engineering projects (see Section 3.2), we fit a Markov chain model on such sequences $D = (x_1, x_2, ..., x_n)$ with states from a finite set S. Then, we can write the Markovian property as:

$$P(x_{n+1}|x_1, x_2, ..., x_n) = P(x_{n+1}|x_n) \quad (1)$$

After the model fitting on the data, a Markov chain model is usually represented via a stochastic transition matrix P with elements $p_{ij} = P(x_j|x_i)$ where it holds that for all i:

$$\sum_j p_{ij} = 1 \quad (2)$$

For our analysis, we will make use of these transition probabilities to identify likely transitions for a variety of different states.[8] For example, if we fit the Markov chain model on sequential property paths for users (see Section 3.2), element p_{ij} of the transition matrix would tell us the probability that users change property j right after i (e.g., in 60% of all cases). By now, e.g., looking for the highest transition probabilities from state i to all other states of S, we can identify potential high-frequent patterns in our data.

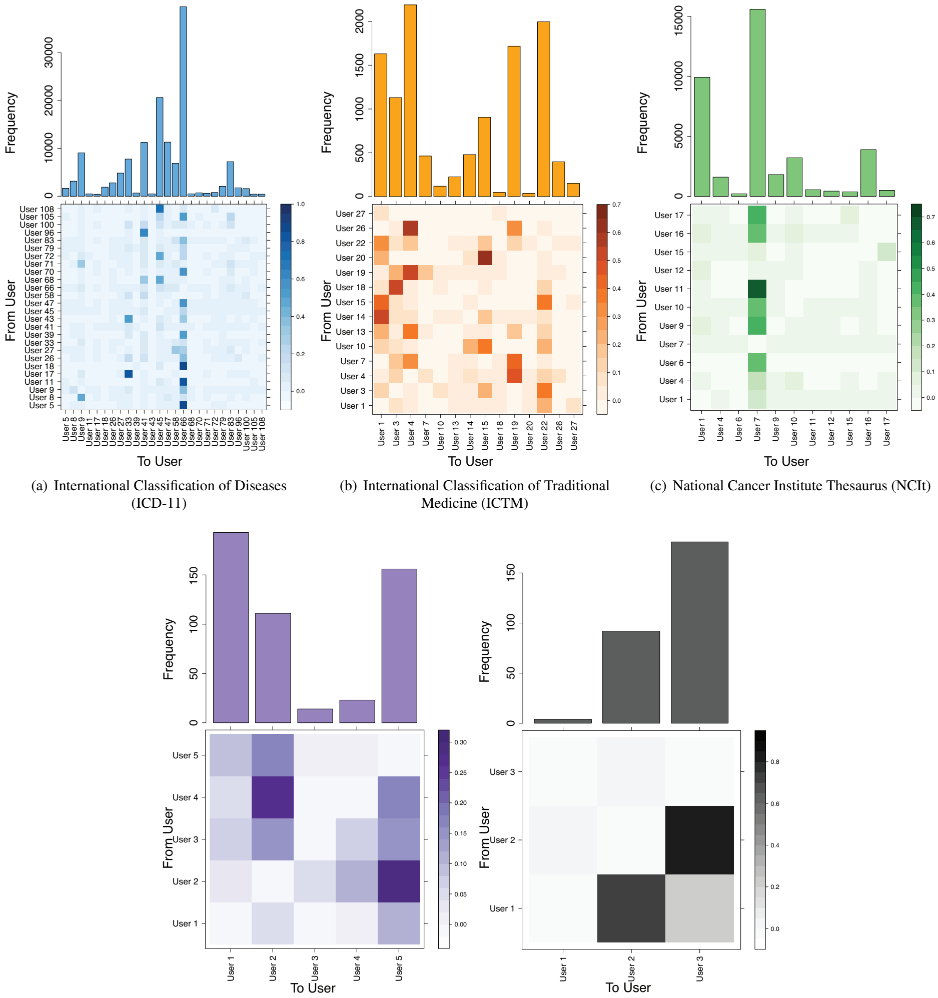
4. Results

4.1. User-sequence paths

In the *User-Sequence Paths* analysis we investigate patterns emerging when looking at sequences of users who contribute to a class of an ontology. Hence, given a sequence of n contributors for a class over time, we identify consecutive users who edit the class (e.g., user Y frequently contribute to a class after user X).

Analyzing the chronologically ordered list of contributors for each class of the five investigated datasets provides the necessary information to identify users who perform changes on classes after (or before) other users. Note that this analysis on its own, without regarding additional factors, such as the

[8] Note that throughout this article we usually refer to the entities modeled (i.e., interactions) instead of states. However, we speak about transition probabilities between these entities as we derive them directly from the resulting model transition matrix.



(a) International Classification of Diseases (ICD-11)

(b) International Classification of Traditional Medicine (ICTM)

(c) National Cancer Institute Thesaurus (NCIt)

(d) Biomedical Resource Ontology (BRO)

(e) Ontology for Parasite Lifecycle (OPL)

Figure 3: **Results for the *User-Sequence Paths* analysis:** The columns and rows of the transition maps (**bottom area** of Figures 3(a) to 3(e)) represent the transition-probabilities between the users of each dataset for a first-order Markov chain, where rows are *source users* and columns are *target users*. A sequence (or transition-probability) is always read *from row to column*. Darker colors represent higher transition-probabilities while lighter colors indicate lesser transition-probabilities. Absolute probability values are dependent on the number of investigated rows and columns, hence relative differences are of greater importance. Darker colored columns identify gardeners, a contributor focused on pruning ontology classes and fixing syntactical errors. The histograms (**top area** of Figures 3(a) to 3(e)) show the number of changes performed by each user (again for a first-order Markov chain) within the five ontologies in alphabetical order. Note, that the *y*-axes for all histograms are scaled differently for each dataset. All datasets have a few users who contributed the majority of changes, while the rest of the users (the long-tail) only contributed a very small number of changes. Note that the transition-probabilities depicted in the transition maps are relative numbers for each column and row individually. The sum of all transition probabilities for one row in the transition maps is 1. For example, if *User 1* exhibits a transition probability of 0.30 to another *User 2* it means that *User 2* has a 30% probability of changing a class after *User 1*. Thus, an inspection of the transition maps **and** histograms is necessary for proper interpretation. To increase readability we have removed users from the plots who have contributed only a very limited number of changes for ICD-11, ICTM and NCIt.

changed property or the performed change-action, does not provide information about actual collaboration. The results of this analysis could be used to potentially identify users who work on the same classes, however, we do not know if they actually collaborate with or just clean up (i.e., a *gardener*, a contributor focused on pruning ontology classes and fixing syntactical errors) after other users.

Path & model description: To analyze user sequences, we iterated over each class of our datasets and extracted a chronologically ordered list of contributors. For example, a given path for a given class can look like the following: *User A, User B, User B, User C*. As we are interested in uncovering patterns of distinct users, we merged multiple consecutive changes by the same user into a single change – our previous example would then unfold into: *User A, User B, User C*. By doing so we remove biases emerging when one single user consecutively changes the same class over and over as this may result in unreasonable high transition probabilities between equal users.

We fit a first-order Markov chain model on this set of paths, where each path represents a single class of the ontology and each element of a path constitutes a change by a single user on the class. The resulting transition probabilities between users then e.g., tell us the probability that *User B* changed a class after *User A*. Hence, they give us thorough insights into frequent consecutive user patterns that emerge when looking at which users contribute to classes in an ontology. Due to reasons of privacy we obfuscated the usernames and replaced them with generic names.

Results: When investigating the transition probabilities (representing a Markov chain of first order) between contributors (see bottom area of Figures 3(a) to 3(e)) we can identify very active users by looking at darker colored columns of the transition maps. Note that these darker colored columns can also be used to identify gardeners, a contributor focused on pruning ontology classes and fixing syntactical errors. As we have merged all consecutive changes of the same user into one single change, the diagonal, representing the transition probabilities between the same users, is 0. The absolute transition probabilities, depicted next to each transition map, are dependent on the absolute amount of observations and users, thus are to be interpreted relatively to each other for each row individually. When looking at the probabilities between the three most active users (being users 66, 45 and 47), and all corresponding target users in ICD-11 we can see that the probabilities are very evenly distributed among them. Meaning that, when investigating the rows (*From User*) that correspond to the top three most active users, probabilities to all target users (*To User*) are very evenly distributed, with very minor exceptions. This indicates that users who contribute many changes to ICD-11 are not followed by specific other contributors, but exhibit an even distribution of users that edited a class after them. Nonetheless, we can clearly identify *User 66* to be the most likely user that edits a class after nearly all other users. This suggests, that *User 66* may represent a gardener, a contributor focused on pruning ontology classes and fixing syntactical errors, in ICD-11.

For NCIt we can clearly observe that *User 7* appears to be a *gardener*, who is checking all the changes contributed by all other users. For BRO *Users 2* and *5* are prominent target users, evident in the high transition probabilities as *To User* (dark columns) – i.e., they frequently edit a class after other users do. Interestingly, the user with the highest number of changes (*User 1*) exhibits very low and evenly distributed transition probabilities (row) and is not necessarily the user that most likely changes a class after another users. This shows us that there does not need to be a necessary connection between the overall activity of users and their activity as a gardener. This could also mean that *User 1* is possibly working independently from the other users in BRO, or that *User 1* is a domain specialist and all other users only change concepts that have not been worked on by that specialist. However, further investigations in future work are required to confirm this observation as our Markov chain analysis is not able to determine this kind of distinction. For OPL we can observe that *User 3* frequently changes the same classes after *User 2*. A similar observation can be made for *Users 1* and *2*. However, one has to keep in mind that *User 1* has contributed a limited number of changes, rendering the observed transition probabilities less useful as they rarely occur.

The histograms (see top area of Figures 3(a) to 3(e)) indicate that a small number of users contribute the majority of changes (similar to a long-tail distribution). However, this appears to be more dominant for specific ontologies compared to others. In order to measure the inequality among contributions of changes to a specific ontology by users, we analyzed the *Normalized Entropy*[9], which is determined by calculating the *Shannon Entropy* and normalizing the entropy by dividing by the logarithm of the length (i.e., number of users) of a distribution. This coefficient measures the statistical dispersion of a distribution – i.e., the coefficient is one if all users contributed equally to the ontology, while it is zero in case of total inequality where a single user conducts all changes. The results indicate that ICD-11 (0.55) exhibits a low entropy value, i.e., the changes are dominated by only a few users. For NCIt (0.61), OPL (0.64) and ICTM (0.68) we receive medium normalized entropies indicating a more democratic contribution to the ontology by users. A high entropy can be observed for BRO (0.81), which indicates that it is a demographically edited ontology – even though there are only five users.[10]

Interpretation & practical implications: The transition probabilities for a first-order Markov chain unveil the roles of certain users and can help to identify users or even groups of users who frequently change the same classes. Users that frequently change classes after other users (i.e., exhibit high transition probabilities in their columns) were identified by us as actual gardeners, curators and administrators of the corresponding projects. If certain users always change the same classes after specific other users, it could be worthwhile for project administrators to investigate if these users are actually collaborating, for example by looking at the changed properties and property

[9] Additionally, we calculated the *Gini Coefficient* for each distribution confirming the results presented here.

[10] Note that we do not necessarily know whether the differences between these distributions are statistically significant as we are mainly interested in the behavior of single distributions.

values, or if a single user is always cleaning up after the other user. In all datasets we were able to observe at least one user who contributed a high number of changes, with evenly distributed transition probabilities to all remaining users. This observation indicates that in all projects, gardeners, curators and administrators are assigned (directly or indirectly) certain parts of the ontology; otherwise the transition probabilities between the very active users would be higher.

The ability of understanding who is most likely going to change a specific class next, as well as the classes that a user is most likely to change next could be used by project administrators to help users in finding and identifying classes (and thus work) of interest. On the other hand, the information about the next, most probable contributor for a class, can even be used to create automatic class recommender systems to suggest work to users, which could help to increase participation. However, these two analyses are beyond the scope of this paper and are therefore subject to future work. In particular for projects the size of ICD-11 and NCIt, mechanisms to automatically identify and assign work are highly useful as it is still very time-consuming to find pending work and users with the necessary knowledge to address the identified work-tasks.

4.2. Structural paths

The investigation of *Structural Paths* involves an analysis of different aspects regarding how and where users contribute to the ontology, such as the depth level of the class that users contribute to next (Section 4.2.1) as well as looking at the relationship distances between consecutively changed classes (Section 4.2.2).

4.2.1. Depth-level paths

In this analysis, we investigate if users concentrate their efforts on specific depth levels of the ontology and if there are certain depth levels that are frequently consecutively changed and receive less concentrated workflows. The gathered results provide the necessary information to implement prefetching mechanisms, potentially helping to minimize the loading and waiting times for contributors. Furthermore, we can determine whether users move along the structure of the underlying ontology when editing classes.

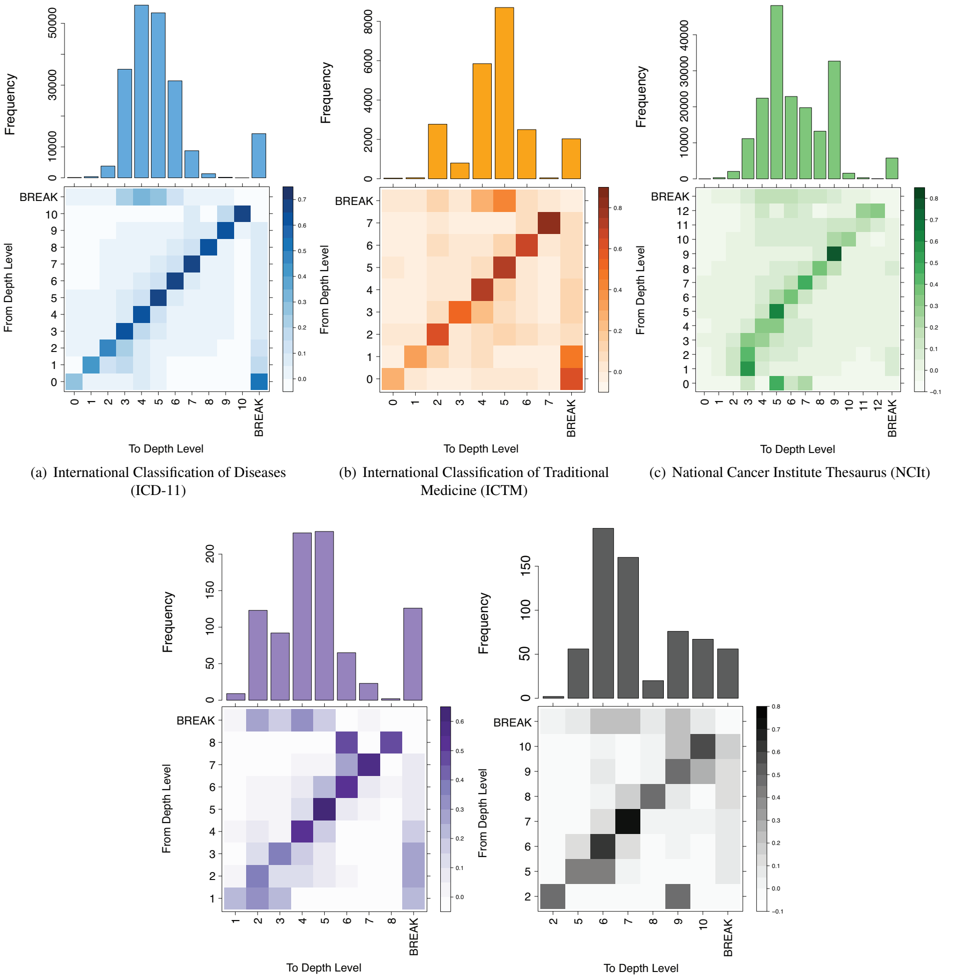
Path & model description: For this analysis, we stored the chronologically ordered depth levels of each changed class for each user (user-based). The depth level of a class is the length of the shortest path between the *root node* of the ontology and the corresponding class. For example, a given path for a given user can look like the following: *Depth 3 (for class A), Depth 3 (for class A), Depth 3 (for class A), Depth 3 (for class B), Depth 4 (for class C).* We merged consecutive changes that were conducted by the same user on the same class into one single sequent change between the same depth levels. Hence, for our previous example we would merge the three successive changes of class A into just two consecutive ones which results in the following final depth-level path: *Depth 3, Depth 3, Depth 3, Depth 4*. This approach helps us to investigate patterns of changing distinct depth levels while still retaining the notion of users consecutively editing the same classes.

Consequently, we fit a first-order Markov chain model on these paths – each path represents a single user and each element of a path represents a corresponding depth level of a class the user has changed. The final transition probabilities give us information about consecutive depth levels that users change over time. For example, they might tell us the probability that users change a class belonging to the third depth level of the ontology after one that has a depth level of 2.

Results: First, the histograms (see top area of Figures 4(a) to 4(e)) show that work is concentrated on certain depth levels of the ontology, with the highest and lowest levels not receiving as much attention as the levels in-between.

As depicted in the transition maps (bottom area of Figures 4(a) to 4(e)), users have a high tendency to edit classes in the same depth levels, visible in the darker colored diagonal. In ICD-11, for the first five depth levels, users appear to have a tendency towards *top-down* editing, evident in the darker immediately right of the diagonal, while this tendency turns around into a *bottom-up* editing behavior, evident in the darker colored squares immediately left of the diagonal, at a depth level of 6 and higher, and appears to be strictly limited to surrounding depth levels. For ICTM (see Figure 4(b)), we can observe a similar trend, again with the tendency towards *top-down* editing appearing to be minimally more dominant. For NCIt, when only looking at the transition map, we can identify a trend towards *bottom-up* editing, evident in the squares directly left of the diagonal being darker than the ones right of the diagonal. However, when also considering the absolute number of changes, depicted in the histogram of Figure 4(c), we can infer that the levels with a higher frequency of occurrence, even though their transition probabilities are more evenly distributed, have a greater impact on the editing strategy. This means that while we can see a *bottom-up* editing behavior for levels 8 to 5 and a *top-down* editing behavior for levels 1 to 4, classes on levels 1 to 4 are more frequently changed than classes on the other levels, hence a tendency towards *top-down* editing can be observed. Thus, when users are not changing the same classes, they still exhibit a preference towards *top-down* editing. Given the short observation periods for BRO and OPL it is hard to infer edit strategies. However, similar to the other projects, we can observe a concentration on the same depth levels with alternating preferences towards higher and lower depth levels. Similar to ICD-11, all datasets exhibit higher transition probabilities between the immediately surrounding depth levels.

Furthermore, we investigate whether the total number of classes as well as the total number of links to the immediate higher (children; edges to classes one level further away from root) and lower (parents; edges to classes one level closer to root) depth level correlate with our findings (Figures 5(f) to 5(j)). For example, the transition map for ICD-11 (see Figure 4(a)) shows that contributors exhibit a *top-down* editing behavior for the first five depth levels, with level 5 exhibiting first signs of *bottom-up* editing. Figure 5(f) shows a higher number of possible transitions from children than parents, indicating that users are in general likelier to follow *top-down* editing-strategies when changing classes, following relationships by chance, of the first four levels. This changes for ICD-11 at level



(a) International Classification of Diseases (ICD-11)

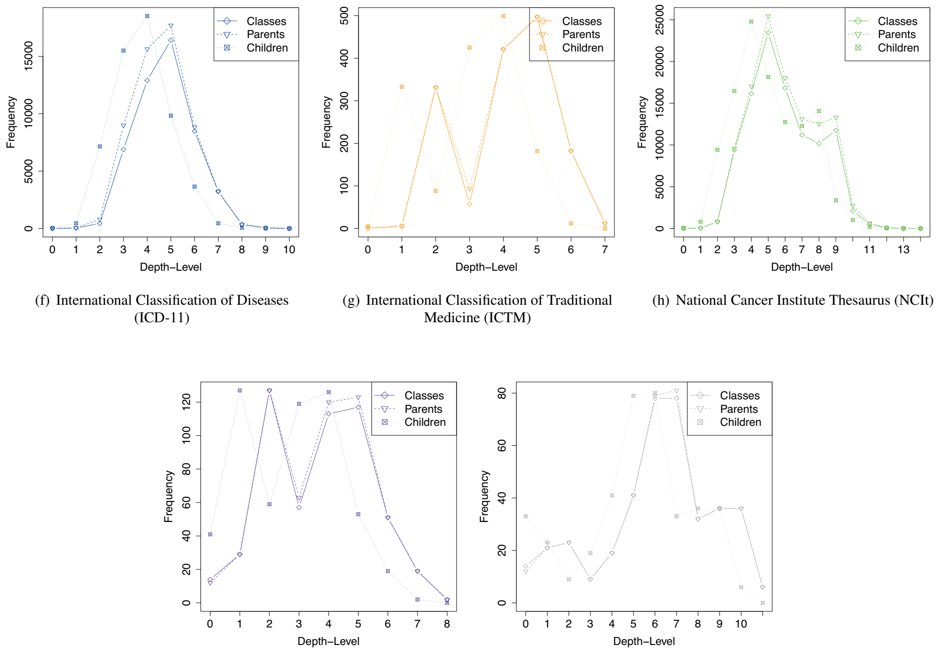
(b) International Classification of Traditional Medicine (ICTM)

(c) National Cancer Institute Thesaurus (NCIt)

(d) Biomedical Resource Ontology (BRO)

(e) Ontology for Parasite Lifecycle (OPL)

Figure 4: **Results for the *Depth-Level Paths* analysis:** The columns and rows of the transition maps (**bottom area** of Figures 4(a) to 4(e)) represent the transition probabilities of a first-order Markov chain between depth levels, where rows are *source depth levels* and columns are *target depth levels*. A sequence (or transition probability) is always read *from row to column*. Darker colors represent higher transition probabilities while lighter colors indicate lesser transition-probabilities. Absolute probability values are dependent on the number of investigated rows and columns, hence relative differences are of greater importance. For classes closer to root a *top-down* editing manner can be observed, while this is reversed for classes further away from root. The sum of all transition probabilities for one row in the transition maps is 1. For example, if *Depth-Level 6* exhibits a transition probability of 0.30 to another *Depth-Level 5* it means that a class on *Depth-Level 5* has a 30% probability of being changed after a class on *Depth-Level 6*. The histograms (**top area** of Figures 4(a) to 4(e)) show the number of changes performed in each depth level aggregated over all users of the respective projects (again for a first-order Markov chain). Throughout all projects, classes located between the first and last few depth levels (in the middle) are changed substantially more frequently than others, suggesting that work is concentrated on some depth levels while others receive none to very few changes at all. Note, that the *y*-axes for all histograms are scaled differently for each dataset. For the *x*-axes (and column/rows of the transition maps) we only display depth levels which exhibit at least one change, thus, the depth level sequences are not necessarily continuous from lowest to highest depth level.



(f) International Classification of Diseases (ICD-11)

(g) International Classification of Traditional Medicine (ICTM)

(h) National Cancer Institute Thesaurus (NCIt)

(i) Biomedical Resource Ontology (BRO)

(j) Ontology for Parasite Lifecycle (OPL)

Figure 5: The **Figures 5(f) to 5(j)** depict the absolute numbers (y-axis; Frequency) of classes as well as the number of edges (*isKindOf*) to classes on the immediate higher (*parents*; closer to root) and lower (*children*; further away from root) depth level for all depth levels (x-axis; Depth-Level). According to Figures 5(f) to 5(j) the transition probabilities depicted in the transition maps correlate with the total number of edges to children and parents for each depth level across all datasets.

5, with a higher number of transitions to parents than to children, and continues until level 10. Resulting in a higher probability of users performing *bottom-up* editing-strategies when changing classes from levels 6 to 10. The same observations can be made for all other datasets, indicating that the class hierarchy influences the edit behavior of contributors.

In all datasets, after taking a *BREAK* (representing an artificially introduced session break when two consecutive changes of the same user are more than 5 minutes apart; for more information see Section 5.4), users exhibit a clear tendency towards changing classes on certain depth levels (e.g., levels 3 to 5 for ICD-11, levels 4 to 5 for ICTM, levels 4 to 7 for NCIt, levels 2 to 4 for BRO and levels 6 to 9 for OPL).

Interpretation & practical implications: The results of this analysis show if, to what extent and where (limited to locality being determined by *isKindOf* relationships) work is conducted and concentrated within the ontology. This information can potentially be used in a variety of ways, for example by ontology-engineering tool developers to adapt the interface of the ontology-engineering tool dynamically to display specific classes after users return from a *BREAK*. Project managers can adapt milestones and project progress reports to reflect the underlying editing strategies (e.g., *top-down* editing), for example by aligning progress with created branches (opposed to complete coverage). Another potential use-case for the results of this analysis involves the prefetching of content in certain environments (e.g., mobile or embedded systems) to minimize waiting times. Across all projects we can observe that classes close to and very far away from the *root* of the ontology are not edited as frequently as other classes. One explanation for this observation could be that classes in lower depth levels (closer to *root*) are mainly used as content dividers and are usually created in the beginning of a project. Thus, they may be more stable and less frequently updated. Classes at the higher depth levels (further away from *root*) on the other hand most likely require extensive expert knowledge. Hence, only a small number of users have the necessary expertise to contribute to these classes. Additionally, the absolute number of classes in the higher and lower depth levels is much lower in all investigated datasets. Note that absolute values of depth levels are less important for the interpretation of the results than their relative position (i.e., closest to root, furthest away from root, etc.). For example, a class at level 6 can exhibit different behaviors in ontologies with 6 or 10 levels.

In all projects, except for NCIt, the depth levels where users start to edit the ontology after they return from a *BREAK* are similar to the ones where they stop editing before taking a *BREAK*. To be able to make that observation we have to take the absolute numbers of changes on each depth level (bottom area of Figure 4) into account when looking at the transition probabilities (top area of Figure 4). NCIt is the only dataset where users appear to be similarly likely to take a *BREAK* after changing classes across all depth levels, except for 0 and 12.

When we combine the results of this analysis with the results of the *User-Sequence Paths* (Section 4.1) we may be able to develop automatic mechanisms to curate and delegate work to users. For example, if we know that a specific user is most probably going to contribute to a class on level 3 and we have a set of classes on that level where that specific user is the most probable next user to contribute to, determined by the *User-Sequence Paths* analysis, we may combine these two observations to create class (and thus work) suggestions for users.

4.2.2. Hierarchical relationship paths

Given the high number of observed transitions between the same depth levels in the *Depth-Level Paths* analyses (Section 4.2.1; bottom area of Figure 4), we conducted an additional analysis investigating the relationships between the changed classes for all users. Hence, we wanted to know if all worked-on classes on the same depth-levels are siblings, cousins or any other kind of close relative? And in general, can we determine if users follow these hierarchical orders of an ontology when contributing to classes on the same depth level? To further strengthen our observation that users are actually moving along the ontological hierarchy when contributing to an ontology (see Section 4.2.1), we analyzed the relationships between the changed classes for each user. Note that whenever we talk about relationships for this analysis, we refer to the hierarchical *isKindOf* relationships between two classes, e.g., parent, child, sibling or cousin. For example, when traversing the shortest-path distance of 2, multiple different nodes can be reached, such as a grandparent (i.e., 2 times up), a grandchild (i.e., 2 times down), a sibling (i.e., 1 time up, 1 time down) or even some other relationship (e.g., 1 time down, 1 time up).

Path & model description: By combining the information from the *Depth-Level Paths* and the relative movement between depth levels, we inferred the hierarchical relationships between two consecutively changed classes of a single user (user-based). For example, if the difference between the depth levels of the investigated classes would be exactly the size of the shortest-path between them (with the shortest-path being > 0), the latter-changed class could either be a *Child*, a *Parent*, an *Ancestor* or a *Descendent* of the first-changed class. Given a relative *DOWN* movement (to a lower depth level) value, depending on the shortest-path value, the second class could be classified as *Child* (shortest-path of 1) or *Descendent* (shortest-path > 1). Analogously follows the definition of a *Parent* and *Ancestor* with a relative *UP* movement. A *Sibling* is defined as the two classes being (i) connected via the same parent with (ii) a shortest-path distance of 2 and (iii) both classes are located on the *SAME* depth level. A *Cousin* is used when two classes on the *SAME* depth level are connected by the same grand parent while exhibiting a shortest-path distance of 4. Every other possible combination of depth level and shortest-path was classified as *Other*. *Self* indicates that the same class that was changed last time was changed again. For example, a consecutive change of *Sibling* and *Self* means that a change was first performed on a class that is a sibling of the previous class (not displayed in this example) and then another change was performed on the same class, however now the relationship changed to *Self* as no new class was involved.

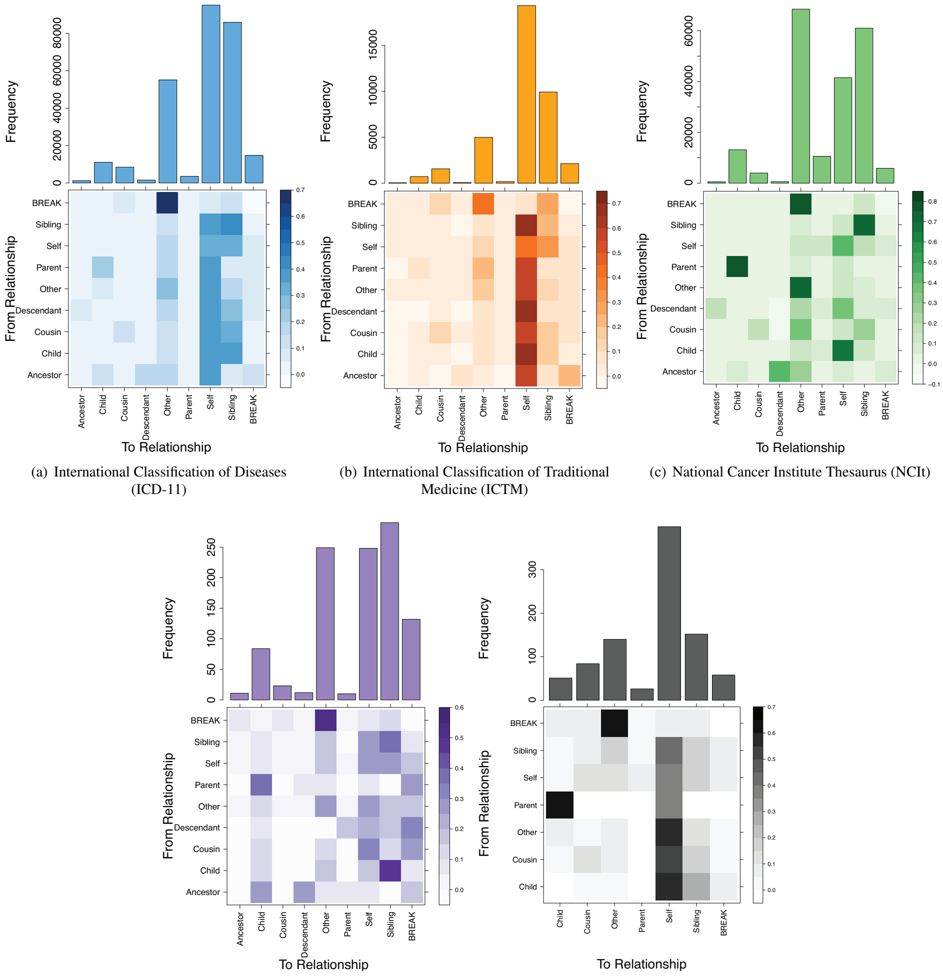
Again, consecutive changes on the same class by the same user have been merged into one single sequent change (c.f. Section 4.2.1), meaning that multiple (more than 2) consecutive changes of the same user on the same class have been merged into *Self* to *Self*. Hence, a given path for a single user can, e.g., look like the following: Sibling, Self, Self, Child.

We fit a first-order Markov chain model to the data – each path represents a single user and each element represents a hierarchical relationship between the classes changed by the user. The resulting transition probabilities of the fitted model can then give us insights into common emerging patterns. E.g., we can identify how probable it is that users change a *Sibling* after a *Child*.

Results: When looking at the histograms (see top area of Figures 6(a) to 6(e)), we can observe that the relationships *Self*, *Sibling* and *Other* are highly represented across all datasets. The transition maps (bottom area of Figures 6(a) to 6(e)) show that after a *BREAK*, across all five datasets, users tend to change classes “somewhere els” in the ontology, evident in the high transition probability from *BREAK* towards *Other*, and are likely not to resume work in the same area of the ontology that they stopped working on. For ICD-11, ICTM and OPL, no matter which relationship type occurs, users tend to edit the same class consecutively (dark colors in the *Self* column). From this *Self* relationship, which is also the one that occurs the most often in ICD-11, ICTM and OPL, users are very likely either to change the same class again (*Self*) or to change a *Sibling* of the current class.

For NCIt, BRO and OPL we can observe that users, when changing a *Parent* are very likely to change a *Child* of that parent afterwards. Note, that this *Child* does not necessarily have to be the same class that was changed prior to the traversal to *Parent*. In all datasets, except for OPL, very high transition probabilities towards *Other* can be observed for all not so frequently present relationships. In particular for NCIt we can observe that *Other* is the most frequently observed transition, even before *Self* and *Sibling*.

Interpretation & practical implications: By combining the results of this analysis with the results of the *Depth-Level Paths* analysis, we can infer that users exhibit a tendency towards *top-down* editing while contributing to the ontology, when only considering changes that occur on different depth levels. If they concentrate their efforts on the same depth levels, users



(a) International Classification of Diseases (ICD-11)

(b) International Classification of Traditional Medicine (ICTM)

(c) National Cancer Institute Thesaurus (NCIt)

(d) Biomedical Resource Ontology (BRO)

(e) Ontology for Parasite Lifecycle (OPL)

Figure 6: **Results for the *Hierarchical-Relationship Paths* analysis:** The columns and rows of the transition maps (**bottom area** of Figures 6(a) to 6(e)) represent the transition-probabilities of a first-order Markov chain between hierarchical-relationship levels, where rows are *source relationships* and columns are *target relationships*. A sequence (or transition-probability) is always read *from row to column*. Darker colors represent higher transition-probabilities while lighter colors indicate lesser transition-probabilities. Absolute probability values are dependent on the number of investigated rows and columns, hence relative differences are of greater importance. Across all datasets, aside from *Self*, a very clear trend towards editing the ontology along *Siblings* can be observed. The histograms (**top area** of Figures 6(a) to 6(e)) show the total number of occurrences of each relationship in the corresponding datasets aggregated over all users (again for a first-order Markov chain). Note, that the *y*-axes for all histograms are scaled differently for each dataset. For the *x*-axes (and column/rows of the transition maps) we only relationships that occur at least once in the corresponding paths, thus the *x*-axes could be different from project to project. Given the very high amount of *Self* and *Sibling* transitions we can concur that users, when they contribute to classes on the same depth level follow a *breadth-first* strategy, meaning that they first concentrate their work on closely related classes (*Siblings*) on the same depth-level before switching to a different branch on the same or any other depth-level.

exhibit a *breadth-first* editing behavior, meaning that they first concentrate their work on closely related classes (*Siblings*) on the same depth-level before switching to a different branch on the same or any other depth-level, either changing the same class multiple times or traversing along siblings of the current class. We can leverage this information not only to refine the previously suggested pre-fetching of classes but also to enhance possible class recommendations. Similarly, it is possible for ontology-engineering tool developers to minimize the necessary efforts of users to contribute to the ontology by implementing, for example, guided workflows that take the underlying edit strategies of the contributors into account.

As classes in ICD-11 and ICTM have a large number of properties and for ICTM certain properties have to be added in multiple languages, the high transition probabilities towards *Self* (dark colors in the *Self* column) are not surprising. One possible explanation for this observation for ICD-11 could be the special functionality available in iCAT (for ICD-11) that allows users to export parts of the ontology as spreadsheets for local editing and adding property values. Once contributors finished editing the spreadsheet they have to enter the data into the system manually, as no automatic import functionality is present. In the iCAT interface, users are simultaneously presented with the ontology tree for navigating through the classes and the corresponding properties and property values. When users select a property they can easily switch between classes, with the selected property staying selected, thus allowing to quickly enter the same properties for different classes.

A similar, yet not as dominant as in ICD-11 and ICTM, behavior can be observed for NCIt and BRO and even to some extent in OPL, which all do not use the export functionality. According to our observations, users travel along the underlying hierarchy when contributing to the ontology. Given the observations made for ICD-11 this behavior can be enforced by providing certain functionalities in the user-interface especially when they compliment the workflows of the contributors.

The results of this analysis have also shown that users are likely to pursue a certain strategy or intermediate goal for their edit sessions, for example changing all classes in a specific (narrow) area of the ontology. This is evident in the observation that after returning from a *BREAK*, users have a very high tendency to change the ontology "somewhere else" (see the transition probabilities from *BREAK* towards *Other* in the top-row of Figure 6), rather than picking up the work, where they left off. This discovery is very important for developing class-recommender, as we may use the results of this analysis to suggest closely related classes to the current class a user is working on, however when that user stays inactive for the duration defined for introducing *BREAK*s the recommendation strategy has to be changed.

4.3. Property paths

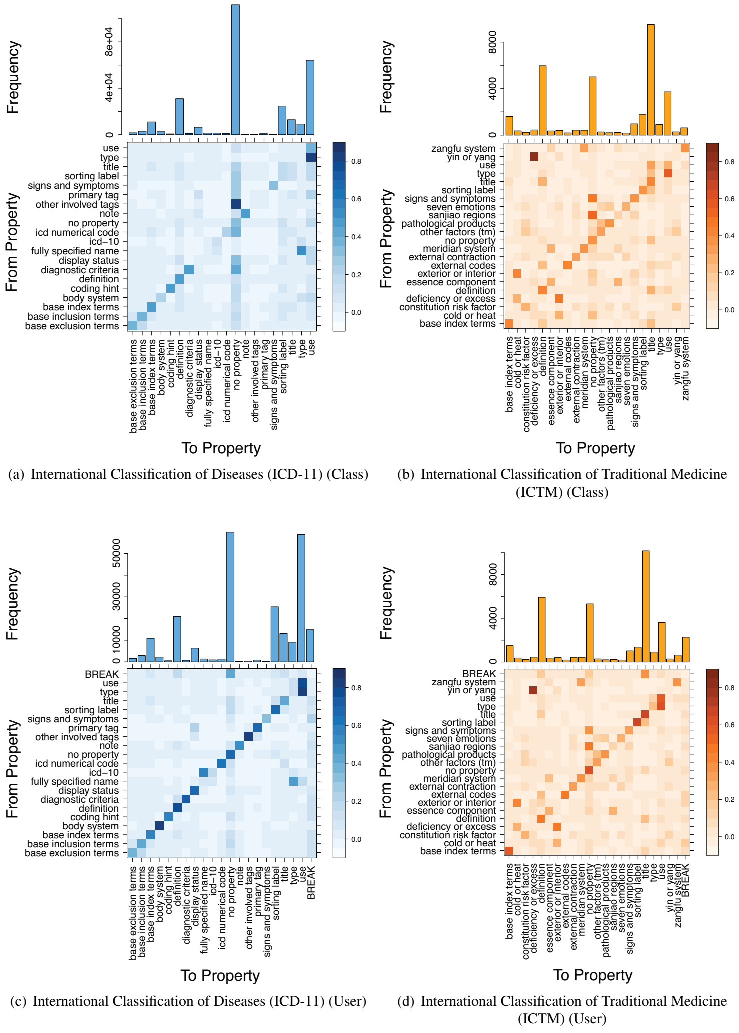
Aside from analyzing different aspects of activity (Section 4.1) and the correlation between contribution patterns and the structure of an ontology (Section 4.2), we can use Markov chains to perform an analysis on the properties that are consecutively change by users in an ontology. This means that, for example, if a property value was edited by a user, we extracted the property (not the value) and created chronologically ordered lists of properties, whose values were changed by the corresponding users. For example, if a user changed the title of a specific class, we would extract *title*, rather than the value inserted into the title property. Now, we provide insights into emerging patterns from different viewing angles for the observations. Thus, we look at property sequences for (a) single users (user-based) and for (b) single classes (class-based) – see Section 3.2. We were not able to perform the *Property Paths* analysis on OPL and BRO as these datasets contain only a very limited number of unique property value changes during our observation periods. We also had to discard the results from NCIt, as the ontology-editing environment for NCIt provides a unique change-queuing mechanism that allows for multiple property values to be changed at the same time, making it impossible to extract chronologically ordered sequential property patterns.

Path & model description: First, we extracted the properties whose values were changed in ICD-11 and ICTM, sorted either by user and timestamp or by class and timestamp. Finally, two different types of chronologically ordered property lists were extracted, one ordered per user and one ordered per class (for both datasets). The properties in *Property Paths* represent the ones which can be assigned a value for each class in ICD-11 and ICTM. Whenever a change did not modify a property (e.g., because the change action dealt with moving or creating a class) we added the element *no property* to the corresponding path. A potential path for a single user or class then may look like: *title, title, title, use*. Similar to previous analyses, if the same user has consecutively changed the same property (e.g., in the previous example *title*) on the same class, we merged these multiple changes into one successive change. Analogously, however without the restriction of the same user, if the same property was changed on the same class, we merged these changes into one sequent change. For previous example, if changes would have been performed editing the referenced properties for a single class, we would end up with the path: *title, title, use*.

Consequently, we fit a first-order Markov chain model on this set of paths (for users or classes). The final transition probabilities of the model then give us information about the probability of changing a value of one property Y after another property X either for users or for classes. For instance, we can find the property Y that most frequently has been changed after property X for classes.

Results: When looking at the histograms (top area in Figures 7(a) to 7(d)) we can see that even after removing not very frequently used properties,[11] both datasets exhibit a few properties which have received a high number of changes, while the remaining majority of properties only received a very limited number of changes. For both datasets, aside from *no property*, the properties *use*, *title* and *definition* appear to be the most frequently used properties. As can be seen in the top area

[11] All properties which where rarely edited have been removed from Figure 7 as they do not hold information but their removal increased the readability of the plots dramatically.



(a) International Classification of Diseases (ICD-11) (Class)

(b) International Classification of Traditional Medicine (ICTM) (Class)

(c) International Classification of Diseases (ICD-11) (User)

(d) International Classification of Traditional Medicine (ICTM) (User)

Figure 7: **Results for the *Property Paths* analysis:** The columns and rows of the transition maps (**bottom area** of Figures 7(a) to 7(d)) represent the transition-probabilities of a first-order Markov chain between consecutively changed properties, where rows are *source properties* and columns are *target properties*. Figures 7(a) and 7(c) represent class-based patterns while Figures 7(b) and 7(d) visualize user-based patterns. A sequence (or transition-probability) is always read *from row to column*. Darker colors represent higher transition-probabilities while lighter colors indicate lesser transition-probabilities. Absolute probability values are dependent on the number of investigated rows and columns, hence relative differences are of greater importance. Across all datasets a very clear trend towards consecutively editing the same properties can be observed. The histograms (**top area** of Figures 7(a) to 7(d)) show the total edits of each property in the corresponding datasets aggregated over all users and classes (again for a first-order Markov chain). Note, that the *y*-axes for all histograms are scaled differently for each dataset. As ICTM and ICD-11 only share a limited amount of properties the *x*-axes (and column/rows of the transition maps) are different from project to project. In both projects and across all 4 different approaches the *title*, *definition* and *use* properties are frequently used. Due to reasons of readability we were forced to remove properties from the plots, which exhibited only a very limited number of changes, thus did not provide substantial information for the purpose of this analysis.

of Figures 7(a) and 7(b), multiple consecutive changes of the same property appear to be fairly common for both datasets. In contrast, when looking at Figures 7(c) and 7(d), which depict the transition probabilities between the sequences of properties changed by each user, we can see an even stronger trend towards consecutively changing the same properties across different classes, especially *definition*, *title* and *use*. For ICD-11 Figures 7(a) and 7(c) show that the class-based approach is less focused on consecutively changing the same property, evident in the brighter diagonal, when compared to the user-based approach. This is due to the export functionality available in iCAT combined with the manual process of inserting the same property for different classes by users of ICD-11. In contrast, such functionality is absent in ICTM, thus leading to similar behaviors for the class and user-based approaches for ICTM. The fact that a large portion of successive changes are conducted on the same property for both approaches analyzed for ICTM could also be due to the multilingual nature of the project, meaning that certain properties, such as *title* and *definition*, have to be entered multiple times in multiple languages. Similar results have been presented by Wang et al. [24], who used association rule mining techniques to analyze the change-logs of ICD-11 and ICTM.

Contributors in ICD-11 have a high tendency of performing *no property* changes after they return from a *BREAK* followed by *use*, *title* and *definition*. In ICTM, users resume their work primarily by changing the *title* property, the *definition* property followed by *no property* changes.

Interpretation & practical implications: One of the main benefits of this analysis is the identification of commonly and consecutively changed properties for classes and users. In turn, this information might potentially be used to suggest work (e.g., prompting a user to check a certain property by combining the *User-Sequence Paths* analysis and the *Property Paths* analysis), or by ontology-engineering tool developers to potentially anticipate the property a user is most likely to change next. The fact that classes appear to exhibit more diverse property-contribution patterns when being changed than users could be a direct result of the multi-lingual nature of ICTM and the already mentioned export functionality present in iCAT. This means that given the most recent property of a class that was edited, we may predict which property is most likely to be changed next. Similarly, we can predict the property a user is going to edit next.

5. Findings and discussion

In this section we first summarize our findings in Section 5.1 before we shortly discuss the potential applicability of higher order Markov chain models in Section 5.2. Next, we discuss differences between the investigated projects in Section 5.3 and finally, point out potential limitations of this work in Section 5.4.

5.1. Summary of findings

We will now discuss our main findings (Table 2) and explore their consequences.

Emergence of micro-workflows: By investigating whether sequential user-contribution patterns (see Section 4.1) can be identified in five different collaborative ontology-engineering projects, we have shown that users appear to work in micro-workflows, indicating that for all investigated projects, each user contains predictive information about the user, who is going to contribute to a specific class next.

Additionally, however not presented in this paper due to reasons of space, we have also conducted an analysis to determine the change type (e.g., adding a property value, moving a class, replacing a property value, etc.) a user is most likely to perform next (as shown in Walk et al. [30] for ICD-11). In this analysis we were able to extract a first-order Markov chain for all datasets presented in this paper, meaning that the last change type that a user performed contains information about the next change type of that user. When combining the information about the user who is most likely to contribute to a class next and the specific change action that this user is most likely to conduct (or the change action that is most likely conducted on a class next), we can create specific tasks for contributors, asking them to perform a certain change on a specific class.

Our results could be used by project managers and ontology-engineering tool developers to identify classes for users and users for classes, helping editors to minimize the necessary efforts for finding and identifying classes to contribute to. Moreover, automatic means of curating and delegating work-tasks to users can be derived by ontology-engineering tool developers, which can help to potentially increase participation as discussed in Kittur and Kraut [31].

User roles can be identified: Across all datasets we were able to identify that a limited number of users have contributed to the majority of all changes. These highly active users are very likely to be *target users* for all other users, meaning that they are very likely to change the same class after another user. Across all five datasets, the roles of these *target users* could be identified by us as moderators or administrators of the corresponding projects performing maintenance tasks, such as gardening (e.g., pruning outdated classes, fixing errors, etc.) or manual verification of newly added data.

Furthermore, we were able to show that moderators and administrators divide work among each other, as they are not very likely to change the same classes directly after another administrator or moderator, even though these users exhibit the highest absolute numbers of changes in the corresponding projects. Looking at the transition probabilities of Figure 3 it is possible to identify users or even groups of users who have a high tendency to work on the same classes, thus might be collaborators or reverting/correcting changes of each other.

Users edit the ontology top-down and breadth-first: The *Depth-Level Paths* analysis (see Section 4.2.1) demonstrated that users have a very high tendency of staying in the same depth level when contributing to the ontology. If editors change depth levels while editing the ontology they exhibit a minimal preference to do so in a *top-down* rather than a *bottom-up* manner. Furthermore, the results suggest that users move along the hierarchy as we were able to show that they follow a *top-down* editing strategy for classes that are closer to the root node while

Table 2: **A summary of all findings** applicable to all investigated biomedical ontologies. All listed findings are discussed in more detail in Section 5.

User-sequence paths (cf. Section 4.1)	**Users work in micro-workflows**	Information about which users successively change a class can be identified; i.e., information about who has edited classes in the past contains predictive information about who is going to change a class next.
	User-roles can be identified	Looking at historic data, we can identify different user roles, i.e., administrators and moderators, gardeners (a contributor focused on pruning ontology classes and fixing syntactical errors) and users that frequently interact with (collaborate/revert) each other.
Structural paths (cf. Section 4.2)	**Users' edit behavior is influenced by the class hierarchy**	Contributors, when adding content to the ontology, are influenced by the class hierarchy.
	Users edit the ontology top-down and breadth-first	By and large, users exhibit a minor tendency towards top-down editing behavior when changing hierarchy levels while contributing. However, when staying in the same hierarchy level, contributors rather follow a *breadth-first* edit behavior, moving from one sibling of a class to the next sibling.
	Users edit closely related classes	Contributors have a very high tendency to consecutively change closely related classes, as opposed to randomly and distantly related classes.
Property paths (cf. Section 4.3)	**Users perform property-based workflows**	Contributors, when adding content to the ontology, tend to concentrate their efforts on one single property, which is added and edited for multiple classes.

this changes to a *bottom-up* editing strategy for classes closer to the deepest depth levels and transitions are more likely to occur along the immediate higher or lower depth level.

To further investigate the distances between changed classes at the same depth levels we investigated the *Hierarchical Relationship Paths* (e.g., child, parent, sibling, cousin, etc.) between these changed classes. We found that users, when they edit classes on the same depth level, follow a *breadth-first* manner, focusing on editing all the siblings of a class before switching to a completely different area of the ontology to continue their work after a *BREAK*.

Users edit closely related classes: Additionally to the *breadth-first* manner that users follow when editing classes in the same depth level, we discovered that users have a very high tendency to work on closely related classes (e.g., the sibling or cousin of the currently changed class). The information collected in Section 4.2 allows to potentially predict (or narrow down) the class a user is going to contribute to next, which, if accurate, is a very valuable information that could be used for a variety of improvements and adaptions. For example, project-administrators could adjust the milestones of the development-strategy to better reflect the way users contribute to the ontology while user-interface designers could emphasize certain areas of the ontology to direct users towards specific classes – especially after they return from a *BREAK* – or implement pre-fetching algorithms to minimize load-times. For contributors in particular, the task of identifying and finding classes that they (i) want and (ii) have the necessary expert knowledge to contribute to is a time-consuming task, which potentially can be minimized by implementing class recommender based on the results of the *Structural Paths Analysis* and *User-Sequence Paths Analysis*.

Users perform property-based workflows: The investigation of sequential patterns for property-contributions showed that in ICD-11, users have a very high tendency of consecutively changing the same property across multiple classes. We could also identify specific patterns that emerge when users successively change properties in collaborative ontology-engineering projects.

The results collected in the Section 4.3 provide new insights for administrators and ontology-engineering tool developers, as they allow the generation of work-tasks (e.g., Please verify the property *title* of the class *XII Diseases of the skin*!). So far, users are always presented first with the section of the interface that allows for changing or adding the *title* and *definition*, which could be one explanation for the high probabilities of users changing these properties when returning from a *BREAK*.

Note, that for this analysis we have used the data from ICD-11 and ICTM, which both share a very similar ontology-engineering tool, thus the results might be biased towards the used ontology-editor.

5.2. Higher order Markov chains

Based on our proposed methodology of using first-order Markov chain models (see Section 3.3) resulting in the findings summarized in Section 5.1, we currently lay our focus on detecting patterns only derived from successive interactions within collaborative ontology-engineering projects. This means, that we identify how likely it is that one specific interaction follows another one (e.g., which user edits a class after another one). This is reasoned by the definition of a first-order Markov chain based on the Markovian property which postulates that the next interaction only depends on the current one.

Contrary, Markov chain models can also be defined on higher orders; this means that the next state of the model (or interaction in our case) depends on a series of preceding ones instead of only the current one. For example, a *second-order* Markov chain model postulates that the next state depends on the current state and also the previous one. Previous studies suggest that human navigation on the Web might be better modeled by using higher order models compared to first-order models (e.g., [32, 29]). Hence, we could assume that this might also be the case for our use-case. By also modeling our data with such higher order models, we would potentially be able to identify longer patterns (e.g., *User A* regularly edits a class after *User B and User C*). Also, possible recommender systems could benefit from the additional predictive power of such higher order chains.[12] While highly interesting, this analyses would be out-of-scope for this article which is why we leave this open for future work.

5.3. Differences between the investigated projects

Even though each project exhibits a different number of depth levels, which all receive a different amount of attention by the contributors, we can observe commonalities of edit strategies between them. For example, the levels 3 to 6 exhibit the highest number of changes in our observation period for ICD-11, while for OPL these levels are 6 and 7.

Regarding the hierarchical relationships we can see that consecutively changing the same class is very likely to happen in ICD-11, ICTM, BRO and OPL regardless of the source relationship (evident in the darker colored *Self* columns in Figures 6(a), 6(b), 6(d) and 6(e)). This *Self*-relationship is still very prominent, however the transition probabilities towards *Self* for NCIt are not as dominant as they are for the other datasets.

Another observation depicted in the transition maps is the clear focus on transitions from *Sibling* to *Sibling* across three out of five datasets, with the exception of ICTM and OPL. One explanation for ICTM could be the fact that some properties of the ontology are multi-lingual, thus require users to add multiple languages for the same property, which are all stored as a single change. For OPL, transitions, except towards *Self* are in general really scarce, indicating that users focused on editing and entering multiple property values (or one property value) of a single class before continuing to the next class.

When looking at the sequence of changed properties for each class (in contrast to: for each user) we can observe a concentration on consecutively changing the same property in ICTM, which is most likely a direct result of the multi-lingual nature of the properties used in this project. In ICD-11 on the other hand, transitions between changed properties of classes are much more diverse and less focused on transitions between the same properties. This observation indicates that either not all properties have received a substantial amount of values for all the possible properties and/or that users make use of this special export functionality of iCAT, thus successively changing the same property is less common as the content is only inserted once into the system.

In the *User-Interface Sections Paths* analysis we have mapped the changed properties to the corresponding sections of the user interface of the used ontology-engineering tools, which essentially represents a more abstract analysis of the *Property Paths* analysis. By investigating the sequences of user interface sections we could confirm that, for ICD-11, users have a very high tendency to consecutively change the same properties for multiple classes, evident in the scarce transitions between different sections and the high concentration on transitions between the same sections. For ICTM this behavior was not as distinctive as it was for ICD-11, which could be due to the missing export functionality and therefore the lack of the previously explained manual import sessions.

In general these observations indicate that the absence or presence of a given functionality of the ontology-engineering tool can produce (and influence) different editing behaviors when developing an ontology.

5.4. Limitations

We were not able to recreate the exact class hierarchy of the ontology for every single change across our observation periods for all datasets. This limitation is partly due to a lack of detail in the change-logs. Thus, we decided to focus our analysis, using all five ontologies *as is* at the latest point in time, which is also what would most likely be used in a *real-world* scenario.

For example, if a class was changed by a user while it was located on depth level 3 and at a later point in time moved to a different location where it now resides at depth level 5, we would assume that this class has always been on depth level 5. Please note that this bias is only present in the *Structural Paths* analyses (Section 4.2). To measure the extent of the potential bias, we counted all changes that were performed on a class before it was moved within in the ontology. Applying this rule to our change dataset, we collected a total of $116,204$ of $439,229$ changes for ICD-11 and $18,958$ of $67,522$ for ICTM. These numbers represent about $1/4$ and $1/3$ of all changes for ICD-11 and ICTM respectively. For BRO 276 of $2,507$ (ca. $1/10$) and for OPL 2 of $1,993$ of all changes were performed on classes, which have been moved afterwards.

Note that an additional requirement for the identification of sequential patterns in collaborative ontology-engineering projects using Markov chains is the availability of rather large change-logs. In general, the less common entities (e.g., properties) are present in the change-log the more (exponentially) observations have to be available in order to detect more fine-grained patterns. Without enough observations (changes), the identification of sequential patterns is either very hard, and can only be approximated, or not possible at all. As can be seen in Table 1, we have selected all of our datasets to satisfy this requirement, as all chosen datasets exhibit a substantial number of changes.

Furthermore, we have included *artificial session breaks* into our analysis as described by Walk et al. [30] to analyze where or what users start to edit in the ontology and where or what users edit before they take a break. For all user-based analyses we have introduced a *BREAK* if two consecutive changes of the same user were apart longer than 5 minutes.

All analyses in this paper are based on *isKindOf* relationships for determining distances and locations within the ontology. We plan on further expanding this analysis by investigating the impact of other kinds of relationships and other features that are available in ontologies on our pattern detection approach.

Even though all datasets presented in this paper are created with WebProtégé or one of its derivatives, there is only one requirement that prevents practitioners from performing this analysis on other ontologies: The availability of a change-log (in

[12] Note that it is necessary to apply model selection techniques as described in [29] in order to identify the most appropriate Markov chain order based on statistical significant improvements of higher orders compared to lower orders

the required granularity for the deemed analyses) that can be mapped onto the underlying ontology. Note that it would be possible to conduct this analysis for ontologies created by single individuals, meaning that "collaboration" is only a requirement when the nature of the analysis requires investigating transitions between multiple users.

Also, the kind of knowledge base (classification, taxonomy or ontology), the used representation language (e.g., OWL and OWL-DL expressivity, RDF, Turtle) or the development tool of a particular collaborative ontology-engineering project in question does not prohibit conducting a pattern analysis as presented in this paper, as long as the underlying knowledge base (and thus the change-log) exhibits the necessary granularity and the semantic properties of interest for the analysis.

However, this also means that the differences of the knowledge representation used languages (i.e., expressivity and types) are not considered by our analysis, with NCIt being a thesaurus and the rest of the investigated datasets being ontologies. Thus, whenever differences are observed between NCIt and the remaining datasets, further research is warranted to determine the origin of this observation.

Furthermore, the analysis presented relies on investigating usage logs of collaborative ontology-engineering projects by looking at changes, performed by users of the corresponding systems. As this only represents one possible way of interacting with the underlying ontology, albeit the most frequently used one, an extension of the conducted Markov chain investigation warrants future work to include, for example, discussions for consensus building, suggestions of terms by users or automatic imports.

6. Related work

For the analysis and evaluation conducted in this paper, we identified relevant information and publications in the domains of (i) Markov chain models, (ii) collaborative authoring systems and (iii) sequential pattern mining.

6.1. Markov chain models

In the past, Markov chain models have been heavily applied for modeling Web navigation – some sample applications of Markov chains can be found in [33, 34, 35, 36, 37, 38]. Also, the Random Surfer model in Google's PageRank [39] can be seen as a special case of a Markov chain.

Previously, researchers investigated whether human navigation is memoryless (i.e., of first order) in a series of studies (e.g., [40, 36]). However, these studies mostly showed that the memoryless model seems to be a quite plausible abstraction (see e.g., [41, 42, 37, 38]). Recently, a study picked up on these investigations and suggested that the Markovian assumption (i.e., property) might be wrong [32]. However, this study did not reveal any statistically significant improvements of higher order models. Singer et al. [29] solved this problem by developing a framework for determining the appropriate order of a Markov chain for a given set of input data. In Walk et al. [30] we applied and mapped the presented framework onto structured logs of changes and provided an in-depth description of the requirements and steps necessary to use the framework in this setting.

In this paper we present a detailed analysis of sequential patterns by applying and analyzing Markov chains across the change-logs of five collaborative ontology-engineering projects in the biomedical domain. A more detailed explanation of the necessary steps to be able to apply Markov chains onto the change-logs of collaborative ontology-engineering projects is presented in Walk et al. [30]. Note that we focus on applying first-order Markov chain models in this work while we see the application of also higher order models as highly interesting future work as discussed in Section 5.2.

6.2. Collaborative authoring systems

Research on collaborative authoring systems such as Wikipedia has in part focused on developing methods and studying factors that improve article quality or increase user participation. These problems represent important facets of collaborative authoring systems and solutions to tackle these problems are of interest for collaborative ontology-engineering projects.

For example, Cabrera and Cabrera [43] demonstrated the effect of minimizing the costs and efforts necessary for users to contribute on potentially achieving higher contribution rates. Another approach, also presented by Cabrera and Cabrera [43], focuses on providing an environment where interactions and communication between contributors are encouraged and performed frequently over a long period of time to establish a group identity and to promote personal responsibility.

More recent research on collaborative authoring systems, such as Wikipedia, focuses on describing and defining not only the act of collaboration amongst strangers and uncertain situations that contribute to a digital good [44] but also on antagonism and sabotage of said systems [45]. It has also been discovered only recently that Wikipedia editors are slowly but steadily declining [46]. Therefore Halfaker et al. [47] have analyzed what impact reverts have on new editors of Wikipedia. Kittur and Kraut [31] showed that an increase in participation can be achieved by directly delegating specific tasks to contributors. As simple as this approach may appear, the identification of work (and thus specific tasks) is still a tedious and time-consuming process, which can only partly be automated due to its assigned complexity.

With the analysis that we described here, we provide new results that we can use to tackle some of the problems for collaborative authoring systems. These problems are also present in collaborative ontology-engineering projects. For example, we can identify new tasks by combining the results of the *User-Sequence Paths* (Section 4.1) and *Property Paths* (Section 4.3) analyses to suggest classes and the corresponding properties to work on to users.

6.3. Sequential pattern mining

In 1995 Agrawal and Srikant [48] have first addressed the problem of sequential pattern mining. They stated that given a collection of chronologically ordered sequences, sequential pattern mining is about discovering all sequential patterns weighted according to the number of sequences that contain these patterns. The presented algorithm represents one of the first *a priori* sequential pattern mining algorithms. This means that a specific pattern cannot occur more frequently (above a threshold) if a sub-pattern of this pattern occurs less often (below that threshold). Other examples of a priori algorithms are [49, 50].

One of the biggest problems assigned to the a priori based sequential pattern mining algorithms was (in the worst case) the exponential number of candidate generation. To tackle this problem Han et al. [51] developed the FP-Growth algorithm.

Many researchers have adapted different algorithms and approaches for different domains to anticipate changing requirements, such as Wang and Han [52] and Hsu et al. [53] who analyzed algorithms for sequential pattern mining in the biomedical domain.

In Walk et al. [30] the authors have presented a novel application of Markov chains to mine and determine sequential patterns from the structured logs of changes of collaborative ontology-engineering projects. Making use of this framework we investigate differences and commonalities across five different collaborative ontology-engineering projects from the biomedical domain.

7. Conclusions & future work

In this work, we discovered intriguing social and sequential patterns that suggest that large collaborative ontology-engineering projects are governed by a few general principles that determine and drive development. Specifically, our results indicate that patterns can be found in all investigated projects, even though the National Cancer Institute Thesaurus (NCIt), the International Classification of Diseases (ICD-11), the International Classification of Traditional Medicine (ICTM), the Ontology for Parasite Lifecycle (OPL) and the Biomedical Resource Ontology (BRO) (i) represent different projects with different goals, (ii) use variations of the same ontology-editors and tools for the engineering process and (iii) differ in the way the projects are coordinated. Using the presented Markov chain analysis, multiple different user-roles could be identified in all investigated datasets. We were also able to see that users work in micro-workflows, meaning that given a specific user, we can identify the most likely users that are editing a specific class next, again independent from the investigated project. When contributing to a project that is created using WebProtégé, iCAT, iCAT-TM or Collaborative Protégé, users exhibit a tendency to do so in a *top-down* and *breadth-first* manner, editing primarily closely related classes while moving along the ontological hierarchy. In ICD-11 and ICTM we were able to identify property-based workflows, meaning that users concentrate their efforts on adding and editing values for one specific property for multiple classes.

The analysis presented not only provides new insights about the engineering and development processes of each single project, but also shows that the analysis of sequential patterns potentially provides actionable insights for different stakeholders in collaborative ontology-engineering projects.

Furthermore, the information of the next possible action (e.g., a user, a change-type, a property, set of classes) or the combination of multiple of these next actions could be used by ontology-engineering tool developers to potentially augment users in collaboratively creating an ontology. For example, by making use of the *Property Paths* analysis to highlight, prefetch, rearrange or adjust sections and content of the interface dynamically, according to the user's needs.

The next logical step to further deepen our understanding of collaborative ontology-engineering projects involves applying the gathered results to productive and live environments, for example as plug-in for (Web)Protégé. Simultaneously, this would allow us to collect valuable data to quantify the usefulness and actionability of the results, generated with our presented approach, in real world scenarios.

Additionally, expanding the Markov chain analysis to take other types of interactions (e.g., discussions, automatic imports and term suggestions by users) into account, represents a potential topic of future work. This also includes a detailed analysis of human factors studies in terms of user-studies (e.g., with a heuristic evaluation or A/B testing) or more sophisticated approaches, such as eye tracking, to assess the usefulness of the presented results for augmenting users when collaboratively engineering an ontology.

Furthermore, as change tracking and click tracking data will likely become available more broadly in the future, we believe that the analysis of this paper and the possible benefits of putting the results into practical use represent an import step towards the development of better (and simpler) ontology editors, which can dynamically anticipate the editing-style of the users. Project administrators could make use of the results of the analysis, for example by allowing for easier delegation of work to the "right" users. This is even more emphasized when considering that the Markov chain analysis is not computationally intensive, making it highly suitable for productive use.

As biomedical ontologies play an increasingly critical role in acquiring, representing, and processing information about human health, we can use quantitative analysis of editing behavior to generate potentially useful insights for building better tools and infrastructures to support these tasks.

Acknowledgement

This work was generously funded by a Marshall Plan Scholarship with support from Graz University of Technology.

[1] T. Gruber, A translation approach to portable ontology specifications, Knowledge Acquisition 5 (1993) 199–220.

[2] W. Borst, Construction of engineering ontologies for knowledge sharing and reuse (1997).

[3] R. Studer, V. R. Benjamins, D. Fensel, Knowledge engineering: Principles and methods, volume 25, 1998, pp. 161–197.

[4] N. F. Noy, T. Tudorache, Collaborative ontology development on the (semantic) web., in: AAAI Spring Symposium: Symbiotic Relationships between Semantic Web and Knowledge Engineering, AAAI, 2008, pp. 63–68.

[5] T. Groza, T. Tudorache, M. Dumontier, Commentary: State of the art and open challenges in community-driven knowledge curation, Journal of Biomedical Informatics 46 (2013) 1–4. URL: http://dx.doi.org/10.1016/j.jbi.2012.11.007. doi:10.1016/j.jbi.2012.11.007.

[6] M. Krötzsch, D. Vrandecic, M. Völkel, Semantic MediaWiki, in: Proceedings of the 5th International Semantic Web Conference 2006 (ISWC 2006), Springer, 2006, pp. 935–942.

[7] S. Auer, S. Dietzold, T. Riechert, OntoWiki–A Tool for Social, Semantic Collaboration, in: Proceedings of the 5th International Semantic Web Conference (ISWC 2006), volume LNCS 4273, Springer, Athens, GA, 2006.

[8] C. Ghidini, B. Kump, S. Lindstaedt, N. Mahbub, V. Pammer, M. Rospocher, L. Serafini, MoKi: The Enterprise Modelling Wiki, in: L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvönen, R. Mizoguchi, E. Oren, M. Sabou, E. P. B. Simperl (Eds.), Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications 2009, Springer, Berlin, Heidelberg, 2009, pp. 831–835.

[9] T. Schandl, A. Blumauer, Poolparty: SKOS thesaurus management utilizing linked data, The Semantic Web: Research and Applications 6089 (2010) 421–425.

[10] T. Tudorache, C. Nyulas, N. F. Noy, M. A. Musen, WebProtégé: A Distributed Ontology Editor and Knowledge Acquisition Tool for the Web, Semantic Web Journal 4 (2013) 89–99.

[11] T. Tudorache, S. M. Falconer, C. I. Nyulas, N. F. Noy, M. A. Musen, Will Semantic Web technologies work for the development of ICD-11?, in: Proceedings of the 9th International Semantic Web Conference (ISWC 2010), ISWC (In-Use), Springer, Shanghai, China, 2010.

[12] J. Pöschko, M. Strohmaier, T. Tudorache, N. F. Noy, M. A. Musen, Pragmatic analysis of crowd-based knowledge production systems with icat analytics: Visualizing changes to the icd-11 ontology, in: Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI) Spring Symposium: Wisdom of the Crowd, Stanford, CA, USA, 2012.

[13] S. Walk, J. Pöschko, M. Strohmaier, K. Andrews, T. Tudorache, C. Nyulas, M. A. Musen, N. F. Noy, PragmatiX: An Interactive Tool for Visualizing the Creation Process Behind Collaboratively Engineered Ontologies, International Journal on Semantic Web and Information Systems (2013).

[14] S. M. Falconer, T. Tudorache, N. F. Noy, An analysis of collaborative patterns in large-scale ontology development projects., in: M. A. Musen, . Corcho (Eds.), K-CAP, ACM, 2011, pp. 25–32.

[15] C. Pesquita, F. M. Couto, Predicting the extension of biomedical ontologies, PLoS Comput Biol 8 (2012) e1002630. URL: http://dx.doi.org/10.1371%2Fjournal.pcbi.1002630. doi:10.1371/journal.pcbi.1002630.

[16] R. S. Goncalves, B. Parsia, U. Sattler, Analysing the evolution of the nci thesaurus, in: Proceedings of the 2011 24th International Symposium on Computer-Based Medical Systems, CBMS '11, IEEE Computer Society, Washington, DC, USA, 2011, pp. 1–6. URL: http://dx.doi.org/10.1109/CBMS.2011.5999163. doi:10.1109/CBMS.2011.5999163.

[17] R. S. Gonçalves, B. Parsia, U. Sattler, Facilitating the analysis of ontology differences, in: Proceedings of the Joint Workshop on Knowledge Evolution and Ontology Dynamics (EvoDyn), 2011.

[18] R. S. Gonçalves, B. Parsia, U. Sattler, Categorising logical differences between owl ontologies, in: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, CIKM '11, ACM, New York, NY, USA, 2011, pp. 1541–1546. URL: http://doi.acm.org/10.1145/2063576.2063797. doi:10.1145/2063576.2063797.

[19] N. F. Noy, A. Chugh, W. Liu, M. A. Musen, A framework for ontology evolution in collaborative environments, in: The Semantic Web-ISWC 2006, Springer, 2006, pp. 544–558.

[20] B. C. Grau, I. Horrocks, Y. Kazakov, U. Sattler, Just the right amount: extracting modules from ontologies, in: Proceedings of the 16th international conference on World Wide Web, ACM, 2007, pp. 717–726.

[21] B. C. Grau, I. Horrocks, Y. Kazakov, U. Sattler, A logical framework for modularity of ontologies, in: Proceedings of the 20th International Joint Conference on Artifical Intelligence, IJCAI'07, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007, pp. 298–303. URL: http://dl.acm.org/citation.cfm?id=1625275.1625322.

[22] E. Mikroyannidi, L. Iannone, R. Stevens, A. Rector, Inspecting regularities in ontology design using clustering, in: Proceedings of the 10th International Conference on The Semantic Web - Volume Part I, ISWC'11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 438–453. URL: http://dl.acm.org/citation.cfm?id=2063016.2063045.

[23] M. Strohmaier, S. Walk, J. Pöschko, D. Lamprecht, T. Tudorache, C. Nyulas, M. A. Musen, N. F. Noy, How ontologies are made: Studying the hidden social dynamics behind collaborative ontology engineering projects, Web Semantics: Science, Services and Agents on the World Wide Web 20 (2013). URL: http://www.websemanticsjournal.org/index.php/ps/article/view/333

[24] H. Wang, T. Tudorache, D. Dou, N. F. Noy, M. A. Musen, Analysis of user editing patterns in ontology development projects, in: On the Move to Meaningful Internet Systems: OTM 2013 Conferences, Springer, 2013, pp. 470–487.

[25] S. Staab, R. Studer, Handbook on Ontologies, 2nd ed., Springer Publishing Company, Incorporated, 2009.

[26] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, P. F. Patel-Schneider (Eds.), The Description Logic Handbook: Theory, Implementation, and Applications, Cambridge University Press, New York, NY, USA, 2003.

[27] N. Sioutos, S. de Coronado, M. W. Haber, F. W. Hartel, W.-L. Shaiu, L. W. Wright, NCI Thesaurus: A semantic model integrating cancer-related clinical and molecular information, Journal of Biomedical Informatics 40 (2007) 30–43.

[28] J. D. Tenenbaum, P. L. Whetzel, K. Anderson, C. D. Borromeo, I. D. Dinov, D. Gabriel, B. A. Kirschner, B. Mirel, T. D. Morris, N. F. Noy, C. Nyulas, D. Rubenson, P. R. Saxman, H. Singh, N. Whelan, Z. Wright, B. D. Athey, M. J. Becich, G. S. Ginsburg, M. A. Musen, K. A. Smith, A. F. Tarantal, D. L. Rubin, P. Lyster, The Biomedical Resource Ontology (BRO) to enable resource discovery in clinical and translational research, Journal of Biomedical Informatics 44 (2011) 137–145.

[29] P. Singer, D. Helic, B. Taraghi, M. Strohmaier, Memory and structure in human navigation patterns, arXiv preprint arXiv:1402.0790 (2014).

[30] S. Walk, P. Singer, M. Strohmaier, D. Helic, N. F. Noy, M. A. Musen, Sequential usage patterns in collaborative ontology-engineering projects, arXiv preprint arXiv:1403.1070 (2014).

[31] A. Kittur, R. E. Kraut, Harnessing the wisdom of crowds in wikipedia: quality through coordination, in: Proceedings of the 2008 ACM conference on Computer supported cooperative work, CSCW '08, ACM, New York, NY, USA, 2008, pp. 37–46.

[32] F. Chierichetti, R. Kumar, P. Raghavan, T. Sarlos, Are web users really markovian?, in: Proceedings of the 21st international conference on World Wide Web, WWW '12, ACM, New York, NY, USA, 2012, pp. 609–618. URL: http://doi.acm.org/10.1145/2187836.2187919. doi:10.1145/2187836.2187919.

[33] J. Borges, M. Levene, Evaluating variable-length markov chain models for analysis of user web navigation sessions, IEEE Trans. on Knowl. and Data Eng. 19 (2007) 441–452. URL: http://dx.doi.org/10.1109/TKDE.2007.1012. doi:10.1109/TKDE.2007.1012.

[34] M. Deshpande, G. Karypis, Selective markov models for predicting web page accesses, ACM Trans. Internet Technol. 4 (2004) 163–184. URL: http://doi.acm.org/10.1145/990301.990304. doi:10.1145/990301.990304.

[35] R. Lempel, S. Moran, The stochastic approach for link-structure analysis (salsa) and the tkc effect, Comput. Netw. 33 (2000) 387–401. URL: http://dx.doi.org/10.1016/S1389-1286(00)00034-7. doi:10.1016/S1389-1286(00)00034-7.

[36] P. L. T. Pirolli, J. E. Pitkow, Distributions of surfers' paths through the world wide web: Empirical characterizations, World Wide Web 2 (1999) 29–45. URL: http://dx.doi.org/10.1023/A:1019288403823. doi:10.1023/A:1019288403823.

[37] R. Sen, M. Hansen, Predicting a web user's next access based on log data, Journal of Computational Graphics and Statistics 12 (2003) 143–155. URL: http://citeseer.ist.psu.edu/sen03predicting.html.

[38] I. Zukerman, D. W. Albrecht, A. E. Nicholson, Predicting users' requests on the www, Proceedings of the Seventh International Conference on User Modeling, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1999, pp. 275–284. URL: http://dl.acm.org/citation.cfm?id=317328.317370.

[39] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, in: Proceedings of the seventh international conference on World Wide Web 7, WWW7, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, 1998, pp. 107–117.

[40] J. Borges, M. Levene, Data mining of user navigation pat-

terns, in: Revised Papers from the International Workshop on Web Usage Analysis and User Profiling, WEBKDD '99, Springer-Verlag, London, UK, UK, 2000, pp. 92–111. URL: http://dl.acm.org/citation.cfm?id=648036.744399.

[41] I. Cadez, D. Heckerman, C. Meek, P. Smyth, S. White, Model-based clustering and visualization of navigation patterns on a web site, Data Min. Knowl. Discov. 7 (2003) 399–424. URL: http://dx.doi.org/10.1023/A:1024992613384. doi:10.1023/A:1024992613384.

[42] R. R. Sarukkai, Link prediction and path analysis using markov chains, Proceedings of the 9th international World Wide Web conference on Computer networks: the international journal of computer and telecommunications netowrking, North-Holland Publishing Co., Amsterdam, The Netherlands, The Netherlands, 2000, pp. 377–386. URL: http://dl.acm.org/citation.cfm?id=347319.346322.

[43] A. Cabrera, E. F. Cabrera, Knowledge-Sharing Dilemmas, Organization Studies 23 (2002) 687–710.

[44] B. Keegan, D. Gergle, N. S. Contractor, Hot off the wiki: dynamics, practices, and structures in Wikipedia's coverage of the Tohoku catastrophes., in: F. Ortega, A. Forte (Eds.), Int. Sym. Wikis, ACM, 2011, pp. 105–113.

[45] N. Shachaf, Beyond vandalism: Wikipedia trolls., Journal of Information Science; Jun2010, Vol. 36 Issue 3, p357-370, 14p, 2 Charts (2010).

[46] B. Suh, G. Convertino, E. H. Chi, P. Pirolli, The singularity is not near: slowing growth of wikipedia, in: WikiSym '09: Proceedings of the 5th International Symposium on Wikis and Open Collaboration, ACM, New York, NY, USA, 2009, pp. 1–10.

[47] A. Halfaker, A. Kittur, J. Riedl, Don't bite the newbies: how reverts affect the quantity and quality of Wikipedia work., in: F. Ortega, A. Forte (Eds.), Int. Sym. Wikis, ACM, 2011, pp. 163–172.

[48] R. Agrawal, R. Srikant, Mining sequential patterns, in: Proceedings of the Eleventh International Conference on Data Engineering, ICDE '95, IEEE Computer Society, Washington, DC, USA, 1995, pp. 3–14. URL: http://dl.acm.org/citation.cfm?id=645480.655281.

[49] R. T. Ng, L. V. S. Lakshmanan, J. Han, A. Pang, Exploratory mining and pruning optimizations of constrained associations rules, in: Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, SIGMOD '98, ACM, New York, NY, USA, 1998, pp. 13–24. URL: http://doi.acm.org/10.1145/276304.276307. doi:10.1145/276304.276307.

[50] S. Sarawagi, S. Thomas, R. Agrawal, Integrating association rule mining with relational database systems: Alternatives and implications, in: Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, SIGMOD '98, ACM, New York, NY, USA, 1998, pp. 343–354. URL: http://doi.acm.org/10.1145/276304.276335. doi:10.1145/276304.276335.

[51] J. Han, J. Pei, Y. Yin, Mining frequent patterns without candidate generation, in: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, SIGMOD '00, ACM, New York, NY, USA, 2000, pp. 1–12. URL: http://doi.acm.org/10.1145/342009.335372. doi:10.1145/342009.335372.

[52] J. Wang, J. Han, Bide: Efficient mining of frequent closed sequences, in: Proceedings of the 20th International Conference on Data Engineering, ICDE '04, IEEE Computer Society, Washington, DC, USA, 2004, pp. 79–. URL: http://dl.acm.org/citation.cfm?id=977401.978142.

[53] C.-M. Hsu, C.-Y. Chen, B.-J. Liu, C.-C. Huang, M.-H. Laio, C.-C. Lin, T.-L. Wu, Identification of hot regions in protein-protein interactions by sequential pattern mining, BMC bioinformatics 8 (2007) S8.

3.4. Sequential Action Patterns in Collaborative Ontology-Engineering Projects: A Case-Study in the Biomedical Domain

This section covers the final article tackling the first research question of this thesis. In detail, it studies the predictability of human trails on the Web by modeling the data with Markov chain models of varying order utilizing the developed framework of this thesis. Again, the trails at interest stem from edit actions in collaborative ontology engineering projects of the biomedical domain. First, this article studies whether regularities and sequential patterns exist in such trails. For doing so, colleagues and I have introduced and used additional methods: (i) an adaption of the Wald-Wolfowitz runs test for randomness detection which I make available online[6] and (ii) a pattern detection algorithm based on PrefixSpan. We have indeed found regularities as well as (longer) serial dependence between subsequent elements of given trails.

Ultimately, we have chosen the Markov chain modeling approach presented in Section 3.2 for predicting human trails. By keeping the findings about regularities and patterns in mind, we have been explicitly interested whether the incorporation of memory into the model can improve accuracy of prediction. As expected, we have found that higher order Markov chain models can improve accuracy for most of the trails studied. Such models can not only be useful for various recommendation approaches, but can also give researchers the tools for assessing the impact of potential changes in the underlying platform. In analogy to this thesis, this article demonstrates a further application of the Markov chain framework as well as provides a strong argument for the benefit of studying memory effects in question.

[6] https://github.com/psinger/RunsTest

Sequential Action Patterns in Collaborative Ontology-Engineering Projects: A Case-Study in the Biomedical Domain

Simon Walk*
Graz University of Technology
Graz, Austria
simon.walk@tugraz.at

Philipp Singer*
GESIS - Leibniz Institute for the Social Sciences
Cologne, Germany
philipp.singer@gesis.org

Markus Strohmaier
GESIS & Univ. of Koblenz
Cologne & Koblenz, Germany
markus.strohmaier@gesis.org

ABSTRACT

Within the last few years the importance of collaborative ontology-engineering projects, especially in the biomedical domain, has drastically increased. This recent trend is a direct consequence of the growing complexity of these structured data representations, which no single individual is able to handle anymore. For example, the World Health Organization is currently actively developing the next revision of the International Classification of Diseases (ICD), using an OWL-based core for data representation and Web 2.0 technologies to augment collaboration. This new revision of ICD consists of roughly $50,000$ diseases and causes of death and is used in many countries around the world to encode patient history, to compile health-related statistics and spendings. Hence, it is crucial for practitioners to better understand and steer the underlying processes of how users collaboratively edit an ontology. Particularly, generating predictive models is a pressing issue as these models may be leveraged for generating recommendations in collaborative ontology-engineering projects and to determine the implications of potential actions on the ontology and community. In this paper we approach this task by (i) *exploring* whether regularities and common patterns in user action sequences, derived from change-logs of five different collaborative ontology-engineering projects from the biomedical domain, exist. Based on this information we (ii) *model* the data using Markov chains of varying order, which are then used to (iii) *predict* user actions in the sequences at hand.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Medical information systems; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*Web-based interaction*

Keywords

Markov Chain; Sequential Pattern; State Prediction; Collaborative Ontology-Engineering

*Both authors contributed equally to this work.

CIKM'14, November 3–7, 2014, Shanghai, China.
Copyright 2014 ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2662049 .

1. INTRODUCTION

The complexity of structured knowledge representations, especially in the biomedical domain, has dramatically increased over the last decade. This recent trend is the direct result of the increasing requirements for these ontologies to satisfy, due to a growing field of application. For example, the International Classification of Diseases in its 10^{th} revision (ICD-10) is used to encode patient history data and to compile health-related spending and morbidity as well as mortality statistics for international comparison. To increase the utility of ICD, the World Health Organization (WHO) is currently developing the 11^{th} revision of this classification (ICD-11), using the Internet and Web 2.0 technologies as collaboration platform and an OWL-based core for knowledge representation. This change in knowledge representation will allow for additional information to be stored inside ICD-11. For example, diseases will have (among others) explicitly defined related/affected body parts and diagnostic criteria. Compared to ICD-10, the new revision now contains around $50,000$ diseases and causes of death, thus has roughly tripled in size and is to be developed until 2017.

Due to this increase in complexity, ontologies, such as ICD-11, can no longer be developed by single authorities. Instead, WHO decided to open-up the development process of ICD-11, allowing everyone with access to the Internet to contribute and discuss changes made to the ontology. However, this open and collaborative ontology-engineering process poses many, yet unidentified, problems to tackle and anticipate. For instance, tracking and monitoring user actions or the overall progress of the underlying ontology as well as helping users to identify work tasks, which they have the required expertise to contribute to, are two either computationally expensive or very time consuming tasks. In particular, administrators of collaborative ontology-engineering projects are in need of better tools to understand and augment users when contributing to these projects.

Objective. Our main objective is to predict user actions in collaborative ontology-engineering projects; e.g., the property a user is most likely to edit next. We want to achieve this task by first exploring whether regularities and sequential patterns exist, then building upon these observations for modeling the data and finally, evaluating the prediction accuracy of each model.

Approach. Specifically, we will approach this objective as follows in subsequent order:
(i) Exploring action sequences: First, we investigate whether action sequences based on several dimensions (e.g., sequential properties changed by users as illustrated in Figure 1) exhibit regularities or are emerging in random fashion before we mine and study common sequential patterns in our data.

(ii) Modeling action sequences: Next, we establish our model approach using Markov chains of varying order, allowing us to incorporate our insights from the first research approach. We also present model selection techniques that can be used for testing and evaluating the accuracy of these models.
(iii) Predicting user actions: Subsequently, we fit these models to our data and evaluate each model, giving insights into their predictive power. The models may be leveraged for generating recommendations in collaborative ontology-engineering projects and to determine the implications of potential actions on the ontology and community.

We perform our experiments on five datasets stemming from different biomedical projects (ICD-11, The International Classification of Traditional Medicine (ICTM), The National Cancer Institute Thesaurus (NCIt), The Biomedical Resource Ontology (BRO) and The Ontology of Parasite Lifecycle (OPL); for more details see Section 2).

Contributions. To the best of our knowledge, this paper presents the most detailed analysis of sequential user actions in collaborative ontology-engineering projects in the biomedical domain for predicting future actions. We find (significant) evidence that (i) regularities and (long) sequential patterns do exist and (ii) demonstrate their utility for predicting the action that is most likely to occur next in our datasets.

Our insights not only improve our understanding of how users engage in collaborative ontology-engineering projects but can also potentially improve the workflow of collaborators by, e.g., recommending properties to contributors to edit next. By doing so, we may be able to better leverage the expertise of contributors by steering them into the right direction. Apart from that, practitioners may also be able to enhance the quality of specific parts of the ontology by promoting them to the right users. Having predictive models for user actions will also allow collaborative ontology-engineering project administrators to assess potential actions regarding their implications on the underlying ontology and community.

Structure of this article. We introduce our experimental setup in Section 2 before we explore action sequences in Section 3. We introduce our model approach in Section 4 and apply and evaluate these models in Section 5. We discuss (Section 6) our findings and related work (Section 7) next and conclude our work in Section 8.

2. EXPERIMENTAL SETUP

In this section we first briefly introduce our five datasets, stemming from the biomedical domain, before we elaborate on our specific dataset preparation steps.

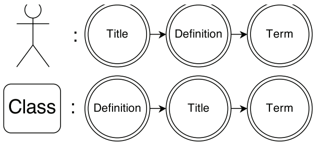


Figure 1: The top row of the figure depicts an exemplary user-based property sequence with properties *Title*, *Definition* and *Term* for a user. This means that the first property that was changed by the user is *Title*, then *Definition* and last *Term*. The bottom row of the figure shows the class-based sequential property path for a class and the same properties *Title*, *Definition* and *Term*. Analogously, the first property that was changed for the class was *Definition*, then *Title* and last *Term*.

2.1 Dataset Description

Table 1 lists the detailed features and observation periods for all datasets used in our analysis. The two largest datasets are ICD-11[1] and the National Cancer Institute Thesaurus (NCIt) [28] with 48,771 and 102,865 classes and 439,299 and 294,471 changes respectively. NCIt is a reference vocabulary for clinical care, translational, basic research and cancer biology. The International Classification of Traditional Medicine (ICTM), which was first intended to be a stand-alone biomedical ontology but was merged with ICD-11 after our observation period, represents a collaborative ontology-engineering project of medium size, with 1,506 classes and a total of 67,522 changes. ICTM is developed by WHO and tries to unify knowledge from traditional medicine practices from China, Japan and Korea. The Biomedical Resource Ontology (BRO) and the Ontology for Parasite Lifecycle (OPL) are two smaller sized collaborative ontology-engineering projects with only 528 and 393 classes and 2,507 and 1,993 changes respectively. BRO is a controlled terminology for describing the source type, areas of research, and activity of biomedical related resources. OPL models the life cycle of a parasite, which is responsible for a number of human diseases.

2.2 Dataset preparation

We extracted sequences from activity logs of the five collaborative ontology-engineering datasets to perform our experiments on. All extracted sequences are either *class- or user-based* (see Figure 1). A class-based sequence depicts a chronology of a specific feature of all changes that were performed *by any user on a single class*. A user-based sequence, analogously, captures the ordered list

Table 1: Characteristics of the investigated datasets. Note that all datasets differ in size (number of classes and users), activity (number of changes) and observation periods. ICD-11 and ICTM both exhibit changes that were performed automatically and are denoted as *# of bots (changes)* in the table. For our analysis we removed these changes.

		ICD-11	ICTM	NCIt	BRO	OPL
Ontology	# of classes	48,771	1,506	102,865	528	393
	# of changes	439,229	67,522	294,471	2,507	1,993
Users	# of users	109	27	17	5	3
	# of bots (changes)	1 (935)	1 (1)	0 (0)	0 (0)	0 (0)
Duration	first change	18.11.2009	02.02.2011	01.06.2010	12.02.2010	09.06.2011
	last change	29.08.2013	17.7.2013	19.08.2013	06.03.2010	23.09.2011
	observation period (ca.)	4 years	2.5 years	3 years	1 month	3 months

[1] http://www.who.int/classifications/icd/ICDRevision/

of specific features of changes that were performed *on any class by a single user* for each dataset. Note that we are interested in studying collaborative behavior in this paper and hence, provide an aggregated view on the data based on all users or all classes. Thus, we always work with a set of distinct sequences where each sequence corresponds to one single user (user-based) or one single class (class-based). In a preprocessing step, we pruned all sequences that exhibit less than two elements, for example, if a class was only ever changed by one user, we removed this specific entry from our training set. Note that we have removed all automatic changes performed in ICD-11 and ICTM for our analyses (see Table 1). In Sections 3 and 4, we will closely investigate the following aspects (and thus sequences) of the activity logs:

(i) Users for Classes. These, solely class-based, sequences consist of chronologically ordered lists, where each list captures one class, of users that changed a specific class.

(ii) Change-Types for Classes and Users. Such a sequence contains a chronology of change-types of the performed changes by a specific user on any class (user-based) or the change-types of the performed changes for a specific class by any user (class-based). We aggregated the performed change-types into abstract classes of changes, which was necessary due to the large variety of different change-types present in our investigated datasets. All changes that edit the value of a property of a class have been aggregated (i.e., added property, edited property, deleted property). Analogously, we have aggregated the changes performed on classes (i.e., added class, moved class, removed class, deleted class).

(iii) Properties for Classes and Users. These sequences consist of chronologically ordered lists of properties changed by a specific user of any class (user-based) or the properties changed for a specific class by any user (class-based).

Note that we were not able to conduct the *Change-Types for Classes and Users* and *Properties for Classes and Users* analyses for NCIt. The reason for this is the existence of a specific feature in the ontology-editor that is used to develop NCIt, which allows contributors to queue changes and commit batches of changes simultaneously to the ontology.

3. EXPLORING ACTION SEQUENCES

In this section we explore the nature of our action sequences at hand. We first investigate randomness and regularities in Section 3.1 and then continue to extract common sequential patterns in Section 3.2.

3.1 Randomness and Regularities

To begin with, we are interested in determining whether our data sequences are produced in random fashion or based on some regularities. One common way to investigate randomness in such sequences or time series is to use *autocorrelation* with varying lags [6]. This method builds on Pearson's product-moment correlation coefficient which determines linear relationships between lagged variables. Contrary, in our paper, we work with categorical data in our sequences (e.g., properties) which is why the autocorrelation method is not directly applicable to our problem at hand.

Another way of determining randomness in data sequences is the so-called *runs test* which is also more specifically entitled *Wald-Wolfowitz runs test* [35, 7]. It is a non-parametric test in which the null hypothesis (the sequence was produced randomly; the elements of the sequence are independent to each other) is tested against the alternative hypothesis stating that the sequence was not produced randomly. In particular, the null hypothesis gets rejected if the total number of runs – a run is a series of identical values (e.g., the sequence "AABA" has three runs "AA", "B" and "A") – is too small leading to a clustered arrangement or too large resulting in a systematic arrangement [21]. Predominantly, the test is only suited for sequences with binary or dichotomous observations. O'Brien and Dyck [21] adapted the initial method by proposing a test that is based on a linear combination of the weighted variances of run lengths. This approach can now be extended to also work with categorical observations which is required for our analyses.[2] We exemplarily applied this method on our individual ICD-11 sequences, and can clearly see that a significant proportion of sequences is produced in a non-random way. This is imminent as the null hypotheses regularly gets rejected (p-value below 0.05) – e.g., the null hypotheses gets rejected for more than 60% of all user property sequences. Our observations in this section warrant further investigations of patterns and structural properties in these sequences. Hence, we next focus on investigating how these present regularities in our sequential patterns look like; i.e., we focus on mining common sequential patterns.

3.2 Sequential Pattern Mining

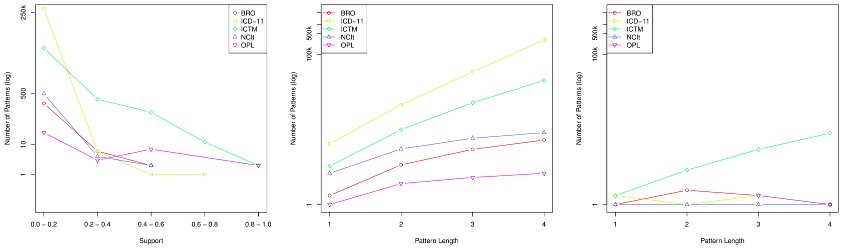
Given our observations made in Section 3.1, we are now interested in actual sequential patterns that account for the regularities in the activity logs. There do exist a variety of algorithms to extract the most frequently used sequential patterns from a set of sequences. We make use of PrefixSpan [22] to investigate commonly used sequential patterns in collaborative ontology-engineering project change-logs, as the algorithm concentrates on expanding (or growing) frequently used patterns and strictly matches only patterns to sequences that are completely identical (i.e., do not exhibit gaps or skipped elements). Support for sequential pattern mining algorithms, a measure to determine how frequent certain patterns are observed in the data, is usually defined as the percentage of all investigated paths that contain a given pattern. Note that all paths have to be chronologically sorted and patterns only consist of succeeding states. For example, the pattern "AB" is *not* present in the sequence "ACBA", as "B" never immediately succeeds "A".

PrefixSpan first scans all available sequences and denotes the number of occurrences for each element in all sequences. It then stores the occurrences and the remainder of the sequences (the suffix) and uses the most frequently used sequential patterns as prefix requirement for the next iteration. Analogously, the prefix is again expanded until a certain level (minimum) support is reached.

We have applied PrefixSpan on the five collaborative ontology-engineering project datasets to see if and to what extent such sequential patterns are present. As can be seen in Figure 2(a), PrefixSpan was able to extract between 5 to 500 patterns for the *Predicting Users for Classes* analysis across all five datasets with a support of 0.2 to 0.4. This means that the identified sequential patterns are present in 20 to 40 percent of all investigated sequences. Figures 2(b) and 2(c) show the number of identified patterns of lengths 1 to 4 for support levels of 0.0 to 0.2 and 0.2 to 0.4. Similar observations could be made for the other analyses.

Given the high number of sequential patterns of lengths 2 to 4 we argue that such patterns play a crucial role in the contributor logs of collaborative ontology-engineering projects at hand. Hence, we believe that there might be some dependence between subsequent

[2] We make an implementation of this method available online at `https://github.com/psinger/RunsTest`. Note though that the method has some limitations. For example, there have to be more than one distinct run length for an element, more than one success run and the number of successes minus the number of success runs of an element has to exceed one. For more details please refer to [21] and the source on github. Hence, we only recommend to perform the test on "somewhat" longer sequences with more runs which is the case for our data at hand.



(a) Patterns per Support

(b) Number of Patterns with length n for Support 0.0 - 0.2

(c) Number of Patterns with length n for Support 0.2 - 0.4

Figure 2: Results of the PrefixSpan analysis on the *Predicting Users for Classes Sequences*: Figure 2(a) shows the number of extracted patterns (y-axis; log-scale) by PrefixSpan for a given support range (x-axis). Support is defined as the percentage of paths that exhibit a certain pattern. For example, the roughly 500 sequential patterns extracted for ICTM with a support level of 0.2 - 0.4 are all present in 20 to 40 percent of all analyzed sequences. Furthermore, Figures 2(b) and 2(c) depict the length (x-axis) and number (y-axis; log-scale) of patterns found for each dataset for support levels 0.0 - 0.2 and 0.2 - 0.4.

elements in a sequence – i.e., memory effects might be in play (see also Rosvall et al. [25] for a discussion surrounding memory in networks). Consequently, we want to incorporate these potential memory effects into our model approach in the next section, in which we resort to Markov chain models of varying order. The goal is to find a model that can describe action sequences and predict user actions in a sound way.

4. MODELING ACTION SEQUENCES

As our main goal of this work is to predict user actions in collaborative ontology-engineering projects, we need to find an appropriate model that we can fit to the data and leverage for prediction. Our choice falls on Markov chain models which are suitable for modeling categorical sequences. Specific variations of model parameters allow us to incorporate our findings of Section 3; i.e., that regularities and specifically, serial dependence seems to play a role in the action sequences at hand. Consequently, we first give a brief introduction into Markov chain models in Section 4.1 also elaborating a way to incorporate our observations about regularities and patterns in the action sequences. Finally, we will explain two model selection techniques in Section 4.2, which is crucial for deciding between different models, which will help us to evaluate the performance of our models. We then apply the methods established in this section in Section 5.

4.1 Markov Chains

A Markov chain is a stochastic process that models transitions from one state to another based on a given state space S. It usually is referred to as *memoryless* which constitutes the so-called *Markov property* stating that the next state only depends on the current state and not on a series of preceding ones. We now briefly provide an introduction to Markov chains; we point the interested reader to a more thorough introduction in previous work [27, 37].

For such a *first-order* Markov chain[3] – a sequence of random variables $X_1, X_2, \ldots, X_n$ – the following holds:

$$\begin{aligned} P(X_{n+1} = x_{n+1} | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) &= \\ P(X_{n+1} = x_{n+1} | X_n = x_n) & \end{aligned} \tag{1}$$

Motivated by our observations in Section 3, where we could see that at least some sequences are arranged in a non-random way – i.e., dependence between elements in a sequence – as well as where we could identify longer sequential patterns to be present in our sequences, we are now also interested in extending this notion of memorylessness of Markov chains to also include memory effects. This means, that we not only want to model the next state as being dependent on the current state, but also on a sequence of preceding states (memory effect). Hence, we now also look at Markov chain models of order k where the future depends on the past k states. We can define a Markov chain model of order k as a process that satisfies:

$$\begin{aligned} P(X_{n+1} = x_{n+1} | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) &= \\ P(X_{n+1} = x_{n+1} | X_n = x_n, X_{n-1} = x_{n-1}, \ldots, & \\ X_{n-k+1} = x_{n-k+1}) & \end{aligned} \tag{2}$$

Such higher order chains can be modified to a first-order Markov chain by using a state space of compound states of size k^4; i.e., the state state includes all sequences of length k which finally leads to a set of size $|S|^k|S|$ (see [27] for details). Additionally, we also introduce a so-called *zero-order* Markov chain model where $k = 0$. In such a model the next state does not depend on any other one but we can see this as a *weighted random selection* that should serve as a baseline for our Markov chain models of varying order.

A Markov chain model is represented by a stochastic transition matrix P if the state space is finite (which it is in our case). This matrix contains the transition probabilities of a state x_i to another state x_j for all possible combinations; the probabilities of each row sum to one. The elements of this matrix represent the parameters

[3] For our chains we assume *time-homogeneity*, i.e., the probability of transitions is independent of n.

[4] We prepend k reset states and append one reset state to each sequence so that we "forget" the history of other sequences in the dataset [9].

θ that we have to determine. For doing so we resort to Bayesian inference (see [30, 27] for details). We use a Laplace prior for the inference process – i.e., we set each $\alpha_{ij} = 1$.

4.2 Markov Chain Model Selection

As we are interested in modeling memory in the process, we model the data with a set of models with varying orders k and consequently, have to evaluate the performance of each model leading to a determination of the most appropriate order out of this set. We need to note that lower order models are always nested within higher order ones by definition and hence, higher order models will always fit at least as good as lower order ones. Nonetheless, such higher order Markov chain models need exponentially more parameters and thus may result in severe overfitting.

First, we apply Bayesian model selection [30, 27] giving us a tool to decide between an array of models. The benefit of this method is that it naturally includes a *Occam's razor*, which means that higher order models receive a penalty due too much higher complexity, which can help us to avoid overfitting and give us insights into significance [17].

As a second method for evaluating varying order Markov chain models we use a stratified[5] k-fold cross-fold validation[6]. Following the concepts of Singer et al. [27] and Walk et al. [37] we train the Markov chain models on each training set and validate the predictive power on the test set. First, we rank the probabilities of each row in the transition matrix – which are the expectations of the Bayesian posterior – using *modified competition ranking* that includes a natural *Occam's razor* for higher orders. Next, we determine the rank of each transition of the test set – i.e., from each *start state* to each *target state* – and henceforth, average over all transitions in the test set. Finally, we average over all folds and visualize the results. Note that the best accuracy to be achieved would be one as this would mean that each transition in the test set would be the highest probability of the transition matrix learned from the training set. This method also directly gives us a prediction accuracy of each model that can provide us with insights into the general prediction performance of a model.

5. PREDICTING USER ACTIONS

In this section we present results for fitting and evaluating (via prediction) the Markov chain models of varying order for all conducted analyses (see Section 4.2). We were not able to conduct all analyses for NCIt, as the ontology editor used for developing NCIt exhibits some special functionality, which makes it impossible to extract chronologically ordered change-types and properties (cf. Section 2).

5.1 Predicting Users for Classes

The Bayesian model selections (see Table 2) mostly suggest first- or second-order Markov chain models to be appropriate fits for the underlying data. Only for NCIt a higher order – i.e., a fifth-order – is suggested. In order to study the predictive power of these varying order Markov chain models, we conducted a stratified 3-fold cross-fold validation task (see Figure 3(a) and Table 2) which mostly agrees with our Bayesian model selection results in terms of order appropriateness. This means, that a first- (ICD-11, ICTM and BRO) or second-order (NCIt and OPL) model are shown to have the best predictive power throughout all datasets (accounting for overfitting).

The results indicate that the next event in a sequence seems to be dependent on at least the previous one; partly, also on a sequence of previous states (memory effects). Such Markov chain models (of first or second order) can be used for predicting the next contributor for a class while simultaneously compensating for overfitting. An average position of mostly below two can be achieved with the corresponding best working model.

This tells us that we have a well-working tool for predicting the user that is most likely changing a class next. We may leverage this for recommending classes to users which are eligible for change. By doing so we may manage to severely improve the workflow of users as they may not need to tap into their own intuitions about which class to change next. Also, this process could improve the quality of some classes by automatically finding experts who should edit the class.

5.2 Predicting Change Types for Users

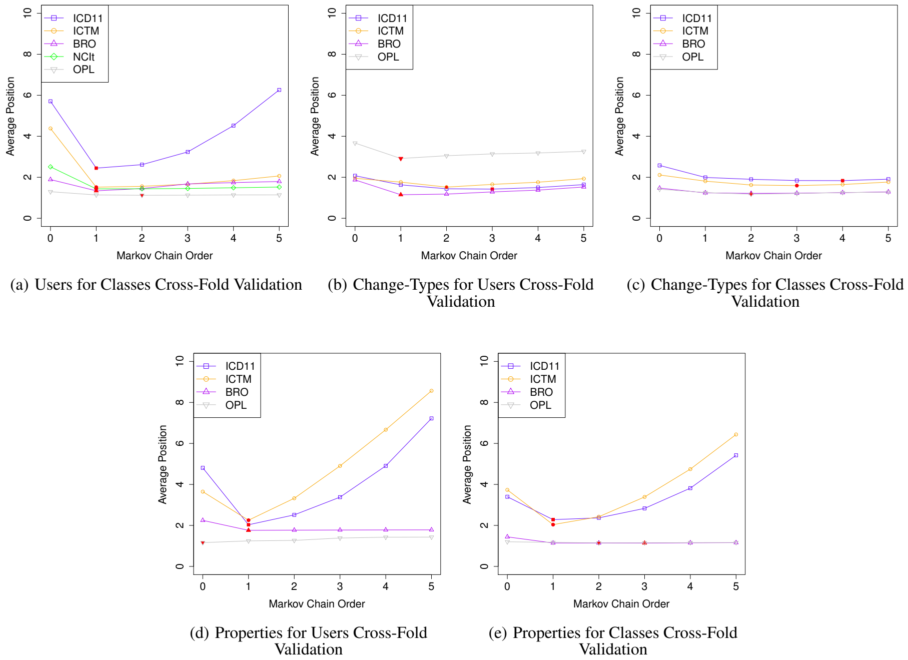
The Bayesian model selection (see Table 2) suggests a fourth-order Markov chain model for ICD-11 and ICTM, a second-order model for BRO and a first-order model for OPL. Subsequently, we conducted a 3-fold stratified cross-fold validation for ICD-11 and ICTM and a 2-fold stratified cross-fold validation for OPL and BRO, due to the smaller number of users available in the latter two datasets (see Figure 3(b) and Table 2). The results suggest that a third-order Markov chain model performed best for predicting the change-type a user is going to perform next for ICD-11. For ICTM and OPL a second-order yielded the best prediction results, while a first-order Markov chain model performed best for BRO. The

[5] Stratified refers to the fact that we try to keep the number of observations equal in each fold.

[6] Note that the number of folds is determined individually for each evaluation due to their stratified nature.

Table 2: The results for all datasets and all analyses conducted in Section 5. Rows marked with *CV* indicate the order of the best-performing Markov chain models of our stratified cross-fold validation task (Section 4.2). Rows marked with *Bayes* depict the order of the Markov chain models determined by the Bayesian model selection task (Section 4.2).

		ICD-11	ICTM	NCIt	BRO	OPL
Predicting Users for Classes (**Section 5.1**)	Bayes	2	1	5	2	2
	CV	1	1	2	1	2
Predicting Change Types for Users (**Section 5.2**)	Bayes	4	4	-	2	1
	CV	3	2	-	1	2
Predicting Change Types for Classes (**Section 5.3**)	Bayes	4	3	-	2	2
	CV	4	3	-	2	2
Predicting Properties for Users (**Section 5.4**)	Bayes	2	1	-	3	4
	CV	1	1	-	1	0
Predicting Properties for Classes (**Section 5.5**)	Bayes	2	1	-	3	5
	CV	1	1	-	3	5



(a) Users for Classes Cross-Fold Validation

(b) Change-Types for Users Cross-Fold Validation

(c) Change-Types for Classes Cross-Fold Validation

(d) Properties for Users Cross-Fold Validation

(e) Properties for Classes Cross-Fold Validation

Figure 3: Results for the *Stratified Cross-Fold Validation* analysis: The plots depict the results of the stratified cross-fold validation for all five datasets for the conducted analyses. The filled elements represent the Markov chain model for each dataset, which achieved the best (lowest) average accuracy (position) score in the prediction tasks. The position score is calculated by determining the position of the next most likely state to occur in a test path given k previous states, where k represents the investigated Markov chain order. Probabilities to select the next most likely state are created using the training set to calculate the transition maps for all datasets and Markov chain orders. The figures show that we can model activity sequences for all of our analyses as first- or higher-order Markov chain models perform best in our prediction task for all datasets, with the only exception of OPL for the *Predicting Properties for Users* analysis (see Figure 3(d)).

cross-fold prediction task also yielded an average accuracy (position) between roughly 1.8 and 3.5.

This indicates that higher-order Markov chains can be used for predicting the change-type a user is most likely to perform next. Practitioners may use this information for recommending change types users should edit next. By doing so we may help to improve the overall progress and quality of the ontology; e.g., if we know that several areas of the ontology or classes lack certain changes, we can steer contributors, which exhibit a preference to perform these kinds of changes, into a specific direction and enforce their contributions in certain branches of the underlying knowledge representation.

5.3 Predicting Change Types for Classes

As depicted in Table 2 the Bayesian model selection suggests a second-order Markov chain model for BRO and OPL, while a third-order model for ICTM and a fourth-order Markov chain model for ICD-11 work best. A stratified 3-fold cross-fold validation (see Figure 3(c) and Table 2) completely agrees with these results for all datasets. The best fitting Markov chain models allow for an average prediction accuracy (position) between 1.8 and 2.0.

The presented results indicate that we can predict the change-type that is most likely conducted on a class next, given at least the two most recent changes on said class as input for our trained Markov chain models. Similar to predicting change types for users, practitioners can use this information for recommending change types that may be useful to change next on a given class. For example, if a class is most likely to receive a certain change type next, we can combine this information with the change types for users and identify a suitable contributor to recommend this class for editing.

5.4 Predicting Properties for Users

The Bayesian model selection yields a second- and first-order Markov chain model for ICD-11 and ICTM and a third- and fourth-order model for BRO and OPL (see Table 2). The conducted 3-fold stratified cross-fold validation, to predict the property a specific user is most likely to change next, yielded a first-order Markov chain model for ICD-11 and ICTM (see Figure 3(d) and Table 2). Due to a limited number of users, a stratified 2-fold cross-fold validation was conducted for BRO and OPL, which showed that a first- and zero-order Markov chain model performs best for predicting the next property for a given user respectively. This means that there was no difference between the Markov chain models trained for OPL and randomly (weighted) choosing (zero-order) the property a user is most likely to change next.

This also means, that for ICD-11, ICTM and BRO we were able to show that subsequent properties users change are dependent on each other; at least for an order of one, which allows for an average prediction accuracy between 1.9 and 2.2. For OPL, the Bayesian model selection and the cross validation approaches do not directly agree with each other; i.e., the Bayesian method suggest an order of four while, interestingly, cross validation would prefer an order of zero (weighted random selection).

In general, by using at least first-order Markov chains it is possible to predict the property a user is most likely to change next for all datasets, except OPL. For steering users into the right direction, we may recommend appropriate properties to change next to contributors.

5.5 Predicting Properties for Classes

Our Bayesian model selection results (see Table 2) suggests for ICD-11 and ICTM a second- and first-order Markov chain model respectively. Furthermore, the results indicate that for BRO a third- and for OPL a fifth-order seem to be appropriate. A stratified 3-fold cross-fold validation (see Figure 3(e) and Table 2) yielded the same results, except for ICD-11, where a first-order model, instead of a second-order model, represents the best predictive accuracy for the underlying data. The conducted cross-fold validation prediction task yielded an accuracy (average position) between roughly 1.8 and 2.4.

Again, our results indicate that we can predict the property that is changed next for a given class reasonably well by using at least a first-order Markov chain. Similar to predicting properties for users, we may now enhance the overall quality of the ontology in an automatic way by aligning the gained information with the properties derived from our user analysis results and recommend users to change specific suitable properties of classes next.

6. SUMMARY AND DISCUSSION

In the previous sections we have studied action sequences of five collaborative ontology-engineering projects from the biomedical domain (see Section 2). To begin with, we provided an initial analysis regarding regularities and sequential patterns in Section 3 to give a basic insight into the processes underlying the user action sequences at hand. First, we started by looking at randomness and regularities by applying an adopted version of the so-called *runs test* exemplary to the ICD-11 dataset in Section 3.1. Our results clearly indicated that a significant array of sequences, based on different features, are produced in a non-random way; this means that at least a portion of sequences is produced in a clustered or systematic arrangement. These observations warranted further studies regarding detailed insights into how these potential regularities look like; hence, we focused on mining sequential patterns next (see Section 3.2). We applied *PrefixSpan* on our User sequences and could identify numerous sequential patterns of longer length – specifically lengths 2 to 4. This lead us to the conclusion that longer patterns seem to play a crucial role in contributor logs of collaborative ontology-engineering projects and that there might be a dependence between subsequent elements in the sequences at hand. Consequently, we hypothesized that it would be beneficial to consider memory effects when modeling our data, and thus user actions. This means, that we wanted to incorporate information of the past into deriving future information – for example, it might be useful to check the two past properties a user has changed for predicting the property she will most likely change next.

For doing so we resorted to Markov chain models of varying order (see Section 4.1) that we applied to our data. We used a Bayesian model selection method for finding the appropriate order for each set of sequences at interest. Supplementary, we were interested in investigating the predictive power of such models, which we evaluated using a cross validation task as described in Section 4.2. The results, as shown in Section 5, confirm our hypotheses: It is indeed useful to incorporate memory effects into the process of modeling user contribution in collaborative ontology-engineering projects. This is particularly imminent as several higher order models are to be preferred throughout all investigations, as can be seen in Table 2. For example, an order of three means that we can best model or predict the next event (e.g., property) by looking at the past three events in a sequence – hence, memory effects are in play. We need to note that all our applied methods compensate the goodness of fit with the corresponding complexity of a model, thus, we penalize higher orders (Occam's razor) which is a necessary step for accounting for potential overfitting.

We can see that both the Bayesian model selection as well as the cross validation prediction task mostly result in similar order suggestion even though they are based on distinct approaches. If the outcome of both methods differ, we can for the most part observe that the cross validation method ensues slightly lower orders than the Bayesian method. This can be explained by the different ways both methods work. The Bayesian method always learns the Markov chain model on the complete model and then performs a model selection strategy which is based on comparing the posterior probabilities of varying order models. Contrary, the cross validation technique learns the Markov chain on a different set (training) compared to where it is evaluated (testing). These differences also account for the drastic mismatch observed between the cross-fold validation prediction task and the Bayesian model selection for OPL in our *Predicting Properties for Users* analysis, where only a very limited number of sequences (three) with unevenly distributed properties across these sequences, is available. Also, the way we rank the probabilities in the cross validation evaluation influences the outcome. Currently, we use modified competition ranking which assigns the worst rank to ties and hence, we very strictly penalize higher orders. Hence, it comes to no surprise for us that if different, the cross validation mostly suggest lower orders than the Bayesian approach. One advantage of the Bayesian approach though is that we could further incorporate penalizations of higher orders when working with model selection; e.g., using an exponential prior [27].

In general, the application of Markov chains on the activity logs of five collaborative ontology-engineering projects has shown that regularities exist. These regularities can potentially be used and exploited by project and community managers to augment and assist users in contributing to the underlying structured knowledge representation. For example, knowing which property a user is most likely to change next and which user is most likely to change a specific concept next could be used to automatically adjust and modify the interface to allow for quicker and personalized workflows. This is especially important for projects the size of ICD-11 or NCIt with thousands of potential classes to contribute to.

We also need to note that the corresponding orders that get suggested might also be – at least to some extent – influenced by how the sequences are shaped; i.e., potential influence factors might be: the distribution of the length of sequences or the number of sequences in a dataset. However, we can argue that these are also properties emerging from how users behave in such systems. Yet, if we are specifically interested in comparing the models of different datasets we need to look deeper into these factors which we leave open for future work. Furthermore, we only work with limited data which also influences the choice of order. Precisely, the number of distinct states as well as the number of observations affect the appropriate order. Basically, the more states one works with, the more difficult it is to compensate the much higher complexity of higher order models with the goodness of fit. Also, we do not necessarily know what would happen if we would perform our investigations on an unlimited number of observations; most likely higher orders will then statistically significantly outperform lower ones (that we e.g., found in our studies) – notwithstanding, working with limited data is a common scenario for researchers and practitioners warranting our experiments and findings.

7. RELATED WORK

The work presented in this paper was inspired by work of the following research areas: Collaborative ontology-engineering, Markov chains and sequential pattern mining.

7.1 Collaborative Ontology Engineering

An ontology represents an explicit specification of a shared conceptualization [14, 5, 32]. In computer-science, this definition usually refers to a construct (formalization) that is automatically processable by a machine representing an abstraction of the real world (shared conceptualization). Ontologies allow computers to "understand" relationships between entities and objects that are modeled in an ontology.

On the other hand, collaborative ontology engineering represents a new field of research with many new problems, risks and challenges. Contributors of such projects, similar to Wikipedia, engage remotely (e.g., via the Internet or a client–server architecture) in the development process to create and maintain an ontology. As mentioned, an ontology represents a formalized and abstract representation of a specific domain; thus, disagreements between authors on certain subjects can occur and tools are needed that augment collaboration and help contributors in reaching consensus when modeling these (and other) topics. Indeed, the majority of the literature about collaborative ontology engineering sets its focus on surveying, finding and defining requirements for the tools used in these projects [20, 13]. Various tools have been developed, specifically aiming at supporting the collaborative development of ontologies. For example, Semantic MediaWikis [18] and its derivatives [2, 12, 26] add semantic, ontology modeling and collaborative features to traditional MediaWiki systems.

Protégé, WebProtégé [34] and its extensions and derivatives for collaborative development are prominent stand-alone tools that are used by a large community worldwide to develop ontologies in a variety of different projects. Both WebProtégé (and its derivatives) and Collaborative Protégé have shown to provide a robust and scalable environment for collaboration and are used in several large-scale projects, including the development of ICD-11 [33].

For analyzing and visualizing the collaborative processes that occur during these projects, Pöschko et al. [24] and Walk et al. [36] have developed *PragmatiX*, a tool that allows to visualize and analyze aspects of the history of collaboratively engineered ontologies. The tool also provides quantitative insights into the ongoing collaborative development processes. Falconer et al. [11] investigated the change-logs of collaborative ontology-engineering projects, showing that users exhibit regularities in their contribution behavior when editing to the ontology. Strohmaier et al. [31] analyzed the collaborative processes in a number of different collaborative ontology-engineering projects by investigating hidden social dynamics and provide new metrics to quantify various aspects of these engineering processes. Wang et al. [39] used association-rule mining to analyze user editing patterns in collaborative ontology-engineering projects.

7.2 Markov chain models

In previous Web studies, Markov chain models have been frequently applied for understanding and modeling Web navigation (e.g., [23, 10, 42]). Mostly, the used Markov chain models were memoryless following the Markovian assumption which is e.g., also modeled in the *random surfer model* in Google's PageRank[8]. Nonetheless, various researchers were also interested in studying the appropriateness of modeling memory effects into models of human navigation – i.e., using higher order chains (e.g., [4, 23]). Yet, the studies revealed that the benefit of higher orders can frequently not compensate the higher complexity and the first-order Markov chain model seems to be a plausible choice. Recently, Chierichetti et al. [9] turned towards again questioning the choice of a first-order chain for modeling human navigation and suggested that the Markovian assumption might be wrong. Consequently, Singer et al. [27] introduced a series of precise model selection techniques for choosing the appropriate Markov chain order. They applied the framework to a series of human navigational datasets and again showed that the memoryless model indeed seems to be a plausible abstraction for human navigation based on the lack of statistically significant improvements of higher order models mostly due to the much higher complexity as already pointed out several years ago. However, the authors also showed that human navigation on a topical level reveals memory effects. Walk et al. [37] adopted this framework to be applicable to structured logs of changes in collaborative ontology-engineering projects and investigated the structure of first-order Markov chains for the change-logs of five different collaborative ontology-engineering projects [38].

7.3 Sequential Pattern Mining

In 1995, Agrawal and Srikant [1] have first addressed the problem of sequential pattern mining. They stated that given a collection of chronologically ordered sequences, sequential pattern mining is about discovering all sequential (chronologically ordered) patterns weighted according to the number of sequences that contain these patterns. The algorithms presented in Agrawal and Srikant [1], in particular AprioriAll and AprioriScale, represent the first *a priori* sequential pattern mining algorithm. In 1996, Srikant and Agrawal [29] further included time-constraints and sliding windows to the definition of sequential patterns and introduced the generalized sequential pattern algorithm (GSP). This means that a specific pattern cannot occur more frequently (above a threshold) if a sub-pattern of this pattern occurs less often (below that threshold). Many other examples of a priori algorithms have been discussed in literature [19, 40, 3], with SPADE [41] being one of the most prominently used and referred to algorithms. One major problem assigned to the a priori based sequential pattern mining algorithms was (in the worst case) the exponential number of candidate generation. To tackle this problem so called pattern-growth approaches have been developed [15, 22].

Many researchers have adapted different algorithms and approaches for different domains to anticipate changing requirements, such as

[16] who analyzed algorithms for sequential pattern mining in the biomedical domain. In Walk et al. [37] the authors have presented a novel application of Markov chains to mine and determine sequential patterns from the structured logs of changes of collaborative ontology-engineering projects.

For the analysis presented in this paper we made use of *PrefixSpan* [22] to investigate if the change-logs of collaborative ontology-engineering projects exhibit commonly used, sequential patterns – we thoroughly introduced this algorithm in Section 3.2.

8. CONCLUSIONS & FUTURE WORK

In this paper our main objective was to predict user actions in collaborative ontology-engineering projects. To that end, we first *explored* if and to what extent regularities and sequential patterns can be extracted from the change-logs of our five datasets. We found that at least a set of sequences were produced in a non-random way and that frequent (longer) patterns can be extracted. We then *modeled* user actions by using Markov chain models which allowed us to incorporate our findings about regularities and patterns. We fitted the models to our sequence data and evaluated them with a specific focus on prediction accuracy. We found that incorporating memory effects (serial dependence) into our models can indeed be useful. The generated predictive models for user actions can not only be used for various recommendation purposes, but also provide project administrators and managers with the means to assess the impact of potential changes on the ontology and the community. For example, knowing which user is most likely to change a specific concept next combined with the information of what kind of change that user is most likely to perform next can potentially be exploited to create personalized task recommendations or to adapt the user-interface to allow for dynamically assisted and faster workflows.

In future work, we first want to extend our choice of models for predicting user action by exploring, for example, varying order Markov chain models, Hidden Markov chain models or Semi Markov chain models. When fitting these models to the data, we plan on providing further evaluation comparisons between these distinct models and consequently, also want to explore the potential of incorporating memory into these alternative models. Furthermore, we want to look at other data sources (e.g., Semantic MediaWikis) to be able to produce more general statements, independent from the datasource, and also closely investigate the influence of different data properties as discussed in Section 6.

We strongly believe that the analysis and predictive models presented in this paper represents an important step towards a better understanding of collaborative ontology-engineering projects in the biomedical domain.

Acknowledgements

We want to thank Natalya F. Noy, Tania Tudorache and Mark A. Musen from the Stanford Center for Biomedical Informatics Research (BMIR) for providing access to the data. This work was partially funded by the DFG German Science Fund grant STR 1191/3-2.

9. REFERENCES

[1] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, ICDE '95, pages 3–14, Washington, DC, USA, 1995. IEEE Computer Society.

[2] S. Auer, S. Dietzold, and T. Riechert. Ontowiki - a tool for social, semantic collaboration. In *Proceedings of the 5th International Conference on The Semantic Web*, ISWC'06, pages 736–749, Berlin, Heidelberg, 2006. Springer-Verlag.

[3] C. Bettini, X. S. Wang, and S. Jajodia. Testing complex temporal relationships involving multiple granularities and its application to data mining (extended abstract). In *Proceedings of the Fifteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, PODS '96, pages 68–78, New York, NY, USA, 1996. ACM.

[4] J. Borges and M. Levene. Data mining of user navigation patterns. In *Revised Papers from the International Workshop on Web Usage Analysis and User Profiling*, WEBKDD '99, pages 92–111, London, UK, UK, 2000. Springer-Verlag.

[5] W. Borst. Construction of engineering ontologies for knowledge sharing and reuse. 1997.

[6] G. E. Box, G. M. Jenkins, and G. C. Reinsel. *Time series analysis: forecasting and control*. John Wiley & Sons, 2013.

[7] J. V. Bradley. Distribution-free statistical tests. 1968.

[8] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the seventh international conference on World Wide Web 7*, WWW7, pages 107–117, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.

[9] F. Chierichetti, R. Kumar, P. Raghavan, and T. Sarlos. Are web users really markovian? In *Proceedings of the 21st international conference on World Wide Web*, WWW '12, pages 609–618, New York, NY, USA, 2012. ACM.

[10] M. Deshpande and G. Karypis. Selective markov models for predicting web page accesses. *ACM Trans. Internet Technol.*, 4(2):163–184, May 2004.

[11] S. Falconer, T. Tudorache, and N. F. Noy. An analysis of collaborative patterns in large-scale ontology development projects. In *Proceedings of the sixth international conference on Knowledge capture*, K-CAP '11, pages 25–32. ACM, 2011.

[12] C. Ghidini, B. Kump, S. Lindstaedt, N. Mahbub, V. Pammer, M. Rospocher, and L. Serafini. Moki: The enterprise modelling wiki. In *The Semantic Web: Research and Applications*, pages 831–835. Springer, 2009.

[13] T. Groza, T. Tudorache, and M. Dumontier. Commentary: State of the art and open challenges in community-driven knowledge curation. *Journal of Biomedical Informatics*, 46(1):1–4, Feb. 2013.

[14] T. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[15] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, SIGMOD '00, pages 1–12, New York, NY, USA, 2000. ACM.

[16] C.-M. Hsu, C.-Y. Chen, B.-J. Liu, C.-C. Huang, M.-H. Laio, C.-C. Lin, and T.-L. Wu. Identification of hot regions in protein-protein interactions by sequential pattern mining. *BMC bioinformatics*, 8(Suppl 5):S8, 2007.

[17] R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the american statistical association*, 90(430):773–795, 1995.

[18] M. Krötzsch, D. Vrandečić, and M. Völkel. Semantic mediawiki. In *The Semantic Web-ISWC 2006*, pages 935–942. Springer, 2006.

[19] H. Mannila, H. Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.

[20] N. F. Noy and T. Tudorache. Collaborative ontology development on the (semantic) web. In *AAAI Spring Symposium: Symbiotic Relationships between Semantic Web and Knowledge Engineering*, pages 63–68. AAAI, 2008.

[21] P. C. O'Brien and P. J. Dyck. A runs test based on run lengths. *Biometrics*, pages 237–244, 1985.

[22] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of the 17th International Conference on Data Engineering*, ICDE '01, pages 215–224, Washington, DC, USA, 2001. IEEE Computer Society.

[23] P. L. T. Pirolli and J. E. Pitkow. Distributions of surfers' paths through the world wide web: Empirical characterizations. *World Wide Web*, 2(1-2):29–45, Jan 1999.

[24] J. Pöschko, M. Strohmaier, T. Tudorache, N. F. Noy, and M. A. Musen. Pragmatic analysis of crowd-based knowledge production systems with iCAT Analytics: Visualizing changes to the ICD-11 ontology. In *Proceedings of the AAAI Spring Symposium 2012: Wisdom of the Crowd*, 2012.

[25] M. Rosvall, A. V. Esquivel, A. Lancichinetti, J. D. West, and R. Lambiotte. Memory in network flows and its effects on spreading dynamics and community detection. *Nature Communications*, 5, 2014.

[26] T. Schandl and A. Blumauer. Poolparty: SKOS thesaurus management utilizing linked data. *The Semantic Web: Research and Applications*, 6089:421–425, 2010.

[27] P. Singer, D. Helic, B. Taraghi, and M. Strohmaier. Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PloS one*, 9(7):e102070, 2014.

[28] N. Sioutos, S. de Coronado, M. W. Haber, F. W. Hartel, W.-L. Shaiu, and L. W. Wright. NCI Thesaurus: A semantic model integrating cancer-related clinical and molecular information. *Journal of Biomedical Informatics*, 40(1):30–43, February 2007.

[29] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '96, pages 3–17, London, UK, UK, 1996. Springer-Verlag.

[30] C. C. Strelioff, J. P. Crutchfield, and A. W. Hübler. Inferring markov chains: Bayesian estimation, model comparison, entropy rate, and out-of-class modeling. *Physical Review E*, 76:011106, Jul 2007.

[31] M. Strohmaier, S. Walk, J. Pöschko, D. Lamprecht, T. Tudorache, C. Nyulas, M. A. Musen, and N. F. Noy. How ontologies are made: Studying the hidden social dynamics behind collaborative ontology engineering projects. *Web Semantics: Science, Services and Agents on the World Wide Web*, 20(0), 2013.

[32] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. volume 25, pages 161–197, 1998.

[33] T. Tudorache, S. Falconer, C. Nyulas, N. F. Noy, and M. A. Musen. Will semantic web technologies work for the development of icd-11? In *The Semantic Web–ISWC 2010*, pages 257–272. Springer, 2010.

[34] T. Tudorache, C. Nyulas, N. F. Noy, and M. A. Musen. WebProtégé: A Distributed Ontology Editor and Knowledge Acquisition Tool for the Web. *Semantic Web Journal*, 4(1/2013):89–99, 2013.

[35] A. Wald and J. Wolfowitz. On a test whether two samples are from the same population. *The Annals of Mathematical Statistics*, 11(2):147–162, 1940.

[36] S. Walk, J. Pöschko, M. Strohmaier, K. Andrews, T. Tudorache, C. Nyulas, N. F. Noy, and M. A. Musen. PragmatiX: An Interactive Tool for Visualizing the Creation Process Behind Collaboratively Engineered Ontologies. *International Journal on Semantic Web and Information Systems*, 9(1):45–78, 2013.

[37] S. Walk, P. Singer, M. Strohmaier, D. Helic, N. F. Noy, and M. A. Musen. Sequential usage patterns in collaborative ontology-engineering projects. *arXiv preprint arXiv:1403.1070*, 2014.

[38] S. Walk, P. Singer, M. Strohmaier, T. Tudorache, M. A. Musen, and N. F. Noy. Discovering beaten paths in collaborative ontology-engineering projects. *Journal of Biomedical Informatics*, 2014.

[39] H. Wang, T. Tudorache, D. Dou, N. F. Noy, and M. A. Musen. Analysis of user editing patterns in ontology development projects. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, OTM '13, pages 470–487. Springer, 2013.

[40] J. T.-L. Wang, G.-W. Chirn, T. G. Marr, B. Shapiro, D. Shasha, and K. Zhang. Combinatorial pattern discovery for scientific data: Some preliminary results. In *Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data*, SIGMOD '94, pages 115–125, New York, NY, USA, 1994. ACM.

[41] M. J. Zaki. Spade: An efficient algorithm for mining frequent sequences. *Machine learning*, 42(1-2):31–60, 2001.

[42] I. Zukerman, D. W. Albrecht, and A. E. Nicholson. *Predicting users' requests on the WWW*. Springer, 1999.

3.5. Computing Semantic Relatedness from Human Navigational Paths: A Case Study on Wikipedia

This article provides answers to the second research question of this thesis. With regard to the first three articles tackling the first research question, this thesis has highlighted that human trails on the Web exhibit patterns, regularities and memory effects at least on some scale. Based on these findings and statements by related work, I have been interested whether we can also utilize human trails on the Web for tasks that are usually solved by using Web content only. As an example, the second research question focuses on studying whether we can leverage human navigational trails for the task of calculating semantic relatedness between concepts.

To that end, this article presents a series of experiments studying the usefulness of around 1.8 million navigational trails through concepts of Wikipedia for this task. Colleagues and I have used a method that builds upon the notion of co-occurrence information and that captures how humans navigate between concepts. The basic idea is that concepts are more semantically related to each other if these concepts frequently co-occur close by in navigational trails. I also make an implementation of this method available online[7] for facilitating future research. By applying the method to given data and evaluating the results on a set of gold-standards and baseline corpora, this article illustrates that we can indeed calculate semantic relatedness between concepts by simply looking at how humans navigate between concepts. However, not all trails are equally useful and intelligent selection of navigational trails based on several characteristics of trails can further enhance the quality of the calculated scores. Overall, this article further demonstrates that human trails on the Web exhibit patterns, regularities and strategies that seem to guide their consecutive behavior. In fact, we can also utilize these as shown in this article for the exemplary task of calculating semantic relatedness between concepts. This argues for an expansion of existing methods to also consider human trails on the Web.

[7] https://github.com/psinger/PathTools

Computing Semantic Relatedness from Human Navigational Paths: A Case Study on Wikipedia

Philipp Singer[a], Thomas Niebler[b], Markus Strohmaier[c,d], Andreas Hotho[b]

[a] *Knowledge Technologies Institute, Graz Technical University, Graz, Austria*
[b] *Data Mining and Information Retrieval Group, University of Würzburg, Würzburg, Germany*
[c] *Computer Science Institute, University of Koblenz-Landau, Koblenz, Germany*
[d] *Computational Social Science Group, GESIS, Cologne, Germany*

Abstract

In this article, we present a novel approach for computing semantic relatedness and conduct a large-scale study of it on Wikipedia. Unlike existing semantic analysis methods that utilize Wikipedia's content or link structure, we propose to use *human navigational paths* on Wikipedia for this task. We obtain 1.8 million human navigational paths from a semi-controlled navigation experiment – a Wikipedia-based navigation game, in which users are required to find short paths between two articles in a given Wikipedia article network. Our results are intriguing: They suggest that (i) semantic relatedness computed from human navigational paths may be more precise than semantic relatedness computed from Wikipedia's plain link structure alone and (ii) that not all navigational paths are equally useful. Intelligent selection based on path characteristics can improve accuracy. Our work makes an argument for expanding the existing arsenal of data sources for calculating semantic relatedness and to consider the utility of human navigational paths for this task.

Keywords:
semantic relatedness, navigation, Wikipedia

Email addresses: `philipp.singer@tugraz.at` (Philipp Singer), `thomas.niebler@uni-wuerzburg.de` (Thomas Niebler), `markus.strohmaier@gesis.org` (Markus Strohmaier), `hotho@informatik.uni-wuerzburg.de` (Andreas Hotho)

This article appears in International Journal of Semantic Web Information Systems, vol 9(4), edited/authored by Amit Sheth, Copyright 2013, IGI Global, http://www.igi-global.com/journal/international-journal-semantic-web-information/1092. Reprinted by permission of the publisher.

1. Introduction

Computing *semantic relatedness*[2] between concepts represents a fundamental challenge on our way to a semantically-enabled web. Especially, common sense knowledge in terms of semantic relatedness is of special interest in e.g., improving information retrieval or language processing. To obtain a judgement of semantic relatedness of two terms or concepts, the idea is to rely on the accumulated or common knowledge. Rubenstein & Goodenough (1965) have pointed out that there is a positive relationship between the degree of semantic relatedness of a pair of terms and the degree to which their contexts are similar. Hence, the idea is that a semantic relatedness score captures this common sense knowledge over a set of contexts and abstracts and generalizes it.

Psychological experiments (Tversky, 1977; Medin et al., 1993) have shown that semantic relatedness is both *context dependent* and *asymmetric*. Context dependency means that the determined relatedness is influenced by the context the words appear in and the semantic relatedness may be asymmetric as people may provide distinct ratings depending on the direction the words are presented. Nevertheless, Aguilar & Medin (1999) showed that this asymmetry just occurs at special occasions and Medin et al. (1993) also showed that the difference in ratings for a given word pair is less than five percent. Hence, we will focus on *symmetric semantic relatedness* in this work, as we believe that this is sufficient for the investigations we want to conduct and we can ignore these small differences.

Recent approaches to identify semantic associations between concepts exploit the rich fabric of emerging information networks such as Wikipedia. Existing semantic analysis methods such as those by Gabrilovich & Markovitch (2007), Ponzetto & Strube (2007a) or Yeh et al. (2009) have shown great potential by using textual or structural (link) information on Wikipedia. While these methods have produced promising results, they only capture semantics from a limited set of people (e.g., Wikipedia editors) and they mostly neglect pragmatics (i.e., how Wikipedia is used). At the same time, millions of web users navigate Wikipedia daily to find information, to educate themselves or for research issues. When navigating a set of articles on Wikipedia, users typically need to tap into their intuitions about real-world concepts and the perceived relationships between them in order to progress towards their set of targeted articles. Humans tend to find intuitive paths instead of necessarily short paths, while contrary an automatic al-

[2] Note that semantic *relatedness* does not necessarily mean the same as *similarity*. Amongst others it includes: similarity, meronymy, hypernymy or IS-A relationships.

gorithm would try to find a shortest path between two concepts that may not be as semantically rich and intuitive as a navigational path conducted by a human.

A great advantage of such navigational paths by humans is that they can be captured in a very simple way. The only prerequisite is that there is a group of users that navigate a system. Furthermore, many existing methods only work well if the system at hand provides high quality content that can be leveraged for calculating semantic relatedness. Contrary, our approach is independent of the content of a resource. It also gives opportunities to calculate semantic relatedness between different kind of resources. For example, suppose we want to calculate semantic relatedness between images and textual pages of a website. This would be a very difficult task for content based approaches, as both resources exhibit different features. The method proposed in this work though would work on any type of resource as long as it is navigated by users.

While such data about navigational paths could potentially represent a profoundly rich resource for calculating semantic relatedness between concepts, it has not received much attention by the research community yet.

1.1. Research Questions

Consequently, we would like to explore (i) whether human navigational paths represent a useful resource for calculating semantic relatedness between concepts on Wikipedia at all, and (ii) if so, in what ways, e.g., what kinds of navigational paths are particularly useful?

In this paper, we tackle these questions and present a series of principled experiments studying the usefulness of almost 1.8 million human navigational paths on Wikipedia for calculating semantic relatedness between concepts (cf. our previous work on this topic (Singer et al., 2013)). Navigational data was obtained from a semi-controlled, large-scale navigation experiment – a Wikipedia-based game called "The WikiGame"[3], in which users need to navigate from a given Wikipedia concept (the starting node) to another concept (the target node). These human navigational Wikigame paths present an abstraction of real user navigation in information networks and enable us to give detailed insights into the usefulness of such data[4].

[3] http://www.thewikigame.com

[4] When we speak about human navigational paths throughout or experiments we refer to the paths captured via the game.

1.2. Contributions of the paper

Our experiments demonstrate that human navigational paths – captured via a Wikipedia-based navigation game – can represent a viable source for calculating semantic relatedness between concepts in information networks. We show that semantic relatedness calculated on this kind of human navigational data can be more precise than semantic relatedness calculated on paths automatically extracted from Wikipedia's plain link structure. Finally, we find that not all navigational paths are equally useful. Intelligent selection of navigational paths based on path characteristics can improve accuracy.

The paper is structured as follows: In Section 2, we give an overview of related work. Section 3 describes our methodology for calculating semantic relatedness based on navigational paths together with a description of the datasets and evaluation methods that we have used in this work. This is followed by Section 4, where we conduct baseline experiments to explore whether human navigational paths can contribute to the task of computing semantic relatedness. In Section 5, we present results from path selection experiments where we investigate which characteristics of human navigation paths render them useful for semantic relatedness. Finally, we discuss and conclude our work in Section 6.

2. Related Work

Computing semantic relatedness between concepts has received much attention from our research community in the last few years, and a wide array of approaches exists. Semantic relatedness scores are widely needed and used in a variety of applications and studies, e.g., word sense disambiguation (Resnik, 1998), usage for word spelling errors (Budanitsky & Hirst, 2001), text segmentation using lexical cohesion (Kozima, 1993; Manabu & Takeo, 1994), image (Smeulders et al., 2000) or document (Srihari et al., 2000) retrieval, cognitive science (Talmi & Moscovitch, 2004) and many more. For a great overview over many different methods to calculate semantic relatedness, see the survey done by Zhang et al. (2012).

Li et al. (2003) point out that semantic relatedness measures and methods can basically be categorized into two groups: *edge-counting-based* and *information-theory-based* methods. When we suppose that a lexical taxonomy has a tree shape then Rada et al. (1989) proved that the distance in the minimum number of edges that separate two given words in such a tree is a metric for specifying the semantic distance between these two words – or to be more precise: the semantic relatedness. While these edge-counting methods make use of *IS-A* relations only, they are very useful for applications with highly constrained taxonomies (Li et al., 2003). According to Resnik (1998) the information-theory-based methods define semantic relatedness between two words using information content and the more information two concepts or words share the more related they are. Jiang & Conrath (1997) presented an approach for computing semantic relatedness between words and concepts combining both edge-based and information-theory-based methods. This method is often simply referred to as the *Jiang-Conrath distance*.

Above described methods can be applied to different information resources. One of the most often and successfully used resource for calculating semantic relatedness is the lexical database WordNet[5] (Miller, 1995). Yang & Powers (2005) proposed a new methodology for calculating semantic relatedness on WordNet using edge-counting techniques. In (Patwardhan, 2006) the authors introduced a WordNet based measure of semantic relatedness by combining both structure and content of WordNet and furthermore enhanced it with co-occurrence information derived from raw text. This enabled the authors to build *gloss* vectors and hence, they used cosine similarity in order to specify semantic relatedness scores between words. A similar approach has been conducted by Banerjee & Pedersen (2003) who used glosses to determine the number of shared words between the definitions of two words for specifying the semantic relatedness between them. Budanitsky & Hirst (2001) compared five different measures of semantic relatedness on WordNet and concluded that the Jiang-Conrath distance is the most accurate by evaluating the results against human judgements and an actual NLP task. Pedersen et al. (2004) introduced a PERL module that implemented nine different measures of semantic relatedness using WordNet and it is widely used by researchers. In subsequent work by Budanitsky & Hirst (2001) the authors again evaluated several semantic relatedness measures using the introduced PERL module using the task of detecting and correcting real-world spelling errors. The authors again show that the Jiang-Conrath distance is superior to other methods. Navigli & Ponzetto (2012a) took WordNet one step further by creating BabelNet, an automatically generated multilingual extension of WordNet. In their publication, they covered the generation of BabelNet by incorporating WordNet, Wikipedia and Machine Translation tools, its evaluation on both new and existing gold standard datasets and the viability to use BabelNet as a resource to perform both monolingual and cross-lingual word sense disambiguation (see Navigli & Ponzetto (2012b)).

More recently, with the rise of the Web 2.0, user-generated content provided

[5] http://wordnet.princeton.edu/

great opportunities for calculating semantic relatedness scores by directly leveraging data generated by humans. Especially tagging systems have attracted lots of interest as a source of data for this task in the past (e.g., (Strohmaier et al., 2012), (Helic et al., 2011), (Cattuto et al., 2008) or (Markines et al., 2009)). But also information networks like Wikipedia have received attention as a resource for calculating semantic relatedness. Because giving a complete review of the literature in this vast field of calculating semantic relatedness using user generated content is beyond the scope of this paper, we will primarily focus our discussion on a few algorithms and methods that are most salient and relevant to this work. Instead, we point the interested reader to a capacious survey about the uses of Wikipedia for many purposes done by Hovy et al. (2012).

Many of the methods we discuss here have been developed for or can easily be applied to Wikipedia. In the following, we differentiate between methods which focus on exploiting different aspects of information networks such as Wikipedia – especially *content* and *links*.

2.1. Content-based methods

A simple way of determining the relatedness between concepts is to represent the content of Wikipedia articles as bag-of-word vectors (Manning et al., 2008). Relatedness between two concepts can then be computed by calculating the similarity between vectors by e.g., using *cosine similarity*.

Gabrilovich & Markovitch (2007) applied *tf-idf* to Wikipedia and introduced a method called *Explicit Semantic Analysis (ESA)*. This method builds a weighted inverted index and extracts a weighted vector of Wikipedia concepts. The vectors of different concepts can be compared, which leads to a calculation of relatedness between terms based on their *tf-idf* weighted vectors. One of the advantages of ESA is that it allows to calculate the relatedness between arbitrary text – e.g., individual words or long documents.

Another method for calculating semantic relatedness is *Latent Semantic Analysis (LSA)* (Landauer et al., 1998; Deerwester et al., 1990). LSA can be used for determining semantic relatedness between Wikipedia concepts by producing word count matrices based on articles and reducing their dimensionality using *singular value decomposition*. Similarity again can be calculated using the angle between vectors.

In addition to analyzing content, link based methods have received increasing attention by our research community lately.

2.2. *Link-based methods*

Two main types of link based methods can be distinguished: (a) methods focusing on link information present for a specific page – i.e., links on a page can be seen as some type of topical markers – and (b) methods exploiting paths through Wikipedia's link network.

2.2.1. *Links as topical markers for Wikipedia concepts*

Ito et al. (2008) use co-occurrence information between links present on the same page for computing semantic relatedness between concepts using a co-occurrence window size of *k* and pruning the vectors with a *tf-idf* based approach. Milne (2008) has proposed a new method of calculating semantic relatedness on Wikipedia leveraging the link structure called "The Wikipedia Link Vector Model". This model judges the similarity between two articles by calculating the angle between the link vectors between two pages. The vectors are built by link counts weighted by the probability of each link occurring. Furthermore, the links get an additional weighting to reduce the impact of frequently occurring links to very common target concepts. Turdakov & Velikhov (2008) have established a similar approach to exploit Wikipedia's link structure in order to calculate similar Wikipedia pages. The technique uses Dice's measure and also ranks two pages similar, if the fraction of similar links is high. The authors as well use a different weighting scheme for the type of link that occurs on a page and they evaluate their approach based on a *word-sense disambiguation* task showing that they can achieve better results than a naive technique of just looking at the neighborhoods of the context and the term in Wikipedia. A more recent method is *Salient Semantic Analysis (SSA)* (Hassan & Mihalcea, 2011). SSA leverages salient features in the context of a term. For example, links on Wikipedia can be interpreted as salient features for terms inside some predefined distance.

2.2.2. *Topology based methods*

Ito et al. (2008) have introduced an adaption to tf-idf called *pfibf* utilizing links between two concepts inside Wikipedia's link network. The assumption is that (i) the number of paths from article *i* to *j* in the Wikipedia topology and (ii) the length of each path from article *i* to *j* determine the relatedness between two concepts.

In (Yeh et al., 2009) the authors present *WikiWalk*, a method that performs random walks based on Personalized PageRank. Based on the output vectors of individual random walks for given words, semantic relatedness is calculated by computing the similarity between both vectors. By pruning the initialization of the teleport vector with Explicit Semantic Analysis, the authors report that their method can even slightly outperform ESA.

Yazdani & Popescu-Belis (2013) created a network topology by parsing the contents of Wikipedia articles and linking articles which are semantically similar. They applied a weighted random walk technique on both the artificially created network as well as the basic Wikipedia topology and calculated the *visiting probability* from one set of nodes to another. They finally showed that a combination of both techniques performed better than both techniques alone.

Strube & Ponzetto (2006) show that straightforward path based measures work very well when focusing on Wikipedia's category taxonomy and that a combination with WordNet is very suitable in order to improve the corresponding accuracy. Furthermore, the authors have evaluated their results by performing a NLP based case study, showing that such knowledge bases collaboratively produced by a huge amount of users like Wikipedia actually can be used for such tasks with similar effects to hand-crafted taxonomies by experts like WordNet (see also (Ponzetto & Strube, 2007b)). In (Milne & Witten, 2008) the authors proposed a similar approach called the "Wikipedia Link-based Measure (WLM)" which as well only leverages Wikipedia's hyperlink structure while it ignores the content and category hierarchy. In (Ponzetto & Strube, 2007a) the authors extend their idea by automatically determining *isa* and *notisa* relations between Wikipedia categories. An automatic extraction of the type of semantic relations has also been successfully conducted by Nakayama et al. (2008).

The work most related to this paper is by West et al. (2009), who have analyzed a set of human navigational paths obtained from *Wikispeedia*[6], a game similar to *"TheWikiGame"*. The authors introduce a method for computing an asymmetric relatedness measure for concepts based on human navigational paths in the corpus. The authors focus on calculating semantic relatedness based on information between a concept in a path and the target page of this game. To the best of our knowledge, West et al. (2009) have been the first to study semantics in human navigational paths on Wikipedia. While their work demonstrates the great potential of this approach, it is limited in some ways: (i) semantic relatedness can only be calculated between a node in a path and a specific target node of a game if they directly co-occur in a path or (ii) the dataset was limited to a small subset of Wikipedia and to a comparatively small set of navigational paths – concretely 1,694 paths.

[6] http://www.cs.mcgill.ca/ rwest/wikispeedia/

2.3. Summary

Calculating semantic relatedness has proven to be an important facet needed for several applications. Many researchers focused on leveraging lexical taxonomies for calculating semantic relatedness scores. More recently, our research community also proposed methods for using user generated content like tagging data or information networks like Wikipedia. As many existing state-of-the-art works evaluated their methods on the same WordSimilarity-353 gold standard dataset, we report some previous accuracy results in Table 1. However, we believe that it is difficult to directly compare our accuracy results to those obtained by existing well-known methods as the exact evaluation mechanisms of existing methods are difficult to judge. We provide a short discussion about this topic in Section 6.

Recent research on link and path based measures (e.g., (Ito et al., 2008), (Yeh et al., 2009) or (Strube & Ponzetto, 2006)) has demonstrated the potential of exploiting topological link structure of Wikipedia for determining semantic relatedness. Our work significantly expands the state-of-the-art in this area by presenting a method for calculating semantic relatedness that utilizes data about human navigational paths through Wikipedia's topological link network. We build on the work and first signals detected by West et al. (2009), but use a novel approach for calculating semantic relatedness based on a corpus of navigational paths that overcomes several limitations the method of West et al. (2009) exhibits. Concretely, the method conducted in this paper can calculate semantic relatedness between any two nodes in a corpus of paths and not only between a node in a path and a specific target game node. We also overcome the necessity of a direct co-occurrence in at least one path between two nodes if one wants to determine the semantic distance between these two concepts. The only limitation of our methodology is that a concept is present at least once in any single path of the corpus in

Table 1: WordSimilarity 353 scores for existing methods

Method	Score	Reference
WikiRelate!	0.48	(Strube & Ponzetto, 2006)
LSA	0.56	(Finkelstein et al., 2002)
WikiWalk	0.63	(Yeh et al., 2009)
WordNet	0.66	(Agirre et al., 2009)
WLVM	0.72	(Milne, 2008)
ESA	0.75	(Gabrilovich & Markovitch, 2007)

order to calculate the semantic relatedness between this and any other concept. In particular, we i) expand the scope of current investigations dramatically (we use ∼*1.8 million paths* from games that are taking place on the *entire* English Wikipedia), ii) deploy state-of-the-art evaluation techniques (WordSimilarity-353 and other standard evaluation datasets) and iii) identify characteristics of navigational paths that are most useful for computing semantic relatedness.

3. Methods and Datasets

In the following, we establish some preliminaries for our work; then we discuss different relatedness measures and the way we apply them to our corpus of human navigational paths. Finally, we describe the datasets at hand and our evaluation method.

3.1. Preliminaries

We define a Wikipedia $\mathbb{W}$ graph G as a graph $G_{\mathbb{W}} = (V_{\mathbb{W}}, E_{\mathbb{W}})$ with vertices – i.e., pages or concepts – $V_{\mathbb{W}}$ and directed edges – i.e., links – $E_{\mathbb{W}} = \{(v, w)|v, w \in V_{\mathbb{W}}\}$. A page $v = (id, title, content) \in V_{\mathbb{W}}$ is a triple of a positive integer *id*, denoting an unique number for easy identification, a string *title*, denoting the title of the page (name) as well as another string *content*, which contains a definition as well as a description of the concept given by the title. The content also contains all the links which define the edges originating from this page. In fact, an edge (v, w) can only be contained in $E_{\mathbb{W}}$, iff the *content* of page v contains a hyperlink to page w.

We can now define $inlinks(v)$ and $outlinks(v)$ for a given page v. The set of outlinks contains all links originating from v and is easily deduced as $outlinks(v) = \{(v, w) \in E_{\mathbb{W}}|w \in V_{\mathbb{W}}\}$. The set of inlinks contains all links pointing from different pages to page v and is defined analogously as $inlinks(v) = \{(w, v) \in E_{\mathbb{W}}|w \in V_{\mathbb{W}}\}$, but is not as directly tractable as the set of outlinks.

Given a graph $G = (V, E)$ (e.g., a Wikipedia graph $G_{\mathbb{W}}$) with vertices V and directed edges $E = \{(v, w)|v, w \in V\}$, we now define a *path* p as a n-tuple $(v_1, \ldots, v_n)$ with $v_i \in V, 1 \leqslant i \leqslant n$ and $(v_i, v_{i+1}) \in E, 1 \leqslant i \leqslant n - 1$. We define $\mathbb{P}$ as the set of all paths and the length of a path $len(p)$ as the length of the corresponding tuple $(v_1, \ldots, v_n)$. Additionally, we want to define $\mathbf{p} = \{v_k|k = 1 \ldots n\}$ as the set of nodes in a path p. Note that $|\mathbf{p}| \leqslant n$.

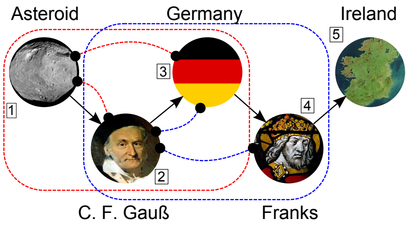
3.2. Measures for semantic relatedness

Schuetze & Pedersen (1997) introduced the method for calculating semantic similarity using lexical co-occurrence information between words – or in our case Wikipedia concepts. The basic idea is to represent each concept as a vector capturing the co-occurrence count to all other concepts in a multi-dimensional space.

A simple procedure for determining semantic relatedness between concepts based on such co-occurrence information is to use *direct co-occurrence*. *First-order co-occurrence* (Schuetze & Pedersen, 1997) implies that concepts can only be similar if they co-occur directly (e.g., in the same documents or in our case paths). However, in our experiments we have observed that this way of calculating semantic relatedness is not suitable for navigational data because many highly related concepts never directly appear in the same path. Furthermore, many word pairs of the WordSimilarity-353 evaluation dataset never co-occur directly in our available data. Also, first-order co-occurrence focuses on semantic relatedness with a tendency to more general concepts.

To avoid this problem, we calculate relatedness between concepts based on the similarity between their corresponding co-occurrence vectors. This is referred to as *second-order co-occurrence* (Schuetze & Pedersen, 1997), which assumes that words are semantically related if they share similar neighbors. Second-order co-occurrence emphasizes if two concepts i and j are similar in a synonymous way. This method also removes the necessity of two concepts directly co-occurring in a path for specifying the semantic relatedness between them and is one of the main advantages of our method over the one introduced in (West et al., 2009). We will use this method for the purpose of our paper.

In order to be able to calculate second-order co-occurrence similarity between two Wikipedia concepts i and j, the corresponding vectors $v_i = [co_{i1}, co_{i2}, ..., co_{in}]$ and $v_j = [co_{j1}, co_{j2}, ..., co_{jn}]$ for both concepts are required. In both vectors, co_{ik} or co_{jk} is the corresponding first-order co-occurrence count between concepts i and k or j and k. We can calculate the relatedness between vectors v_i and v_j by using a similarity (distance) measure between vectors. As an example, let us suppose we want to calculate the semantic relatedness between concept $i = Germany$ and $j = Ireland$ given our example illustrated in Figure 1a. We use the corresponding vectors v_i and v_j present in the symmetric co-occurrence matrix v depicted on the right side in Figure 1b and calculate the cosine similarity measure given both vectors, which results in 0.35 for this simple example (the sliding windows mechanism will be described in Section 3.3). Throughout this work we use *cosine similarity* (Cattuto et al., 2008; Salton, 1989) which has linear complexity and has shown good performance in comparable cases. The choice of similarity measures is secondary to our method[7].



(a) Sliding windows

	Asteroid	C. F. Gauß	Germany	Franks	Ireland
Asteroid	0	1	1	0	0
C. F. Gauß	1	0	1	1	0
Germany	1	1	0	1	1
Franks	0	1	1	0	1
Ireland	0	0	1	1	0

(b) Co-occurrence matrix

Figure 1: Figure 1a illustrates the sliding window mechanism for a window size of $k = 3$ on a path from *Asteroid* to *Ireland*[8]. Circles represent Wikipedia articles, rounded rectangles represent a window. The solid arrows represent the path taken, the dashed lines with dotted ends each represent a (symmetric) co-occurrence between two concepts. We only highlight the first two windows. The resulting co-occurrence matrix after all steps is shown on the right in Figure 1b.

3.3. Semantic relatedness for paths

To compute semantic relatedness using co-occurrence information inside a corpus of navigational paths, we define a *co-occurence graph* between concepts as a *weighted undirected graph* $G_{coocc} = (V_{\mathbb{W}}, E_{coocc})$ where the set of vertices $V_{\mathbb{W}}$ are the corresponding Wikipedia concepts available for all paths in the corpus. The set of edges E_{coocc} is defined as follows: An edge $e = \{u, v\}$ lies in E_{coocc}, iff u and v appear on the same path p, i.e., if $u, v \in p$. The weight of the edge $w(e)$ is determined by the number of co-occurrences of u and v on any path $p \in \mathbb{P}$. We use undirected co-occurrence edges as we do not want to explicitly capture the order of the appearance of two nodes in a path but rather specify their symmetric co-occurrence as we are also calculating symmetric semantic relatedness between

[7] We used other vector similarity measures like *Mutual Information* or *Dice Coefficient* with similar results.

[8] The asteroid picture is courtesy of NASA/JPL-Caltech. All other pictures are published under the Creative Commons licence.

two concepts. To capture relatedness of two concepts in a corpus of human navigation paths, we use *sliding windows* of a variable size k following the natural assumption that the distance between two concepts is crucial for calculating precise semantic relatedness scores (cf. (Schuetze & Pedersen, 1997)). In this paper we investigate paths instead of documents. Hence, we follow and investigate the hypothesis that the navigational distance between two concepts in a path (i.e., they are just a specified hop range away in a path) is important in order to calculate precise semantic relatedness scores. Given a navigational path with a large length of 20 visited nodes, it may make more sense to consider the co-occurrence between the first and third node in the path instead of the first and final target node in this long path.

Formally, this sliding window process can be expressed in the following way: An undirected co-occurence edge $e = \{u, v\}$ between two concepts $u, v \in V_{\mathbb{W}}$ only exists, iff u and v appear on the same path p and for the directed subgraph $q = (u, \ldots, v)$ of p the inequality $len(q) \leqslant k$ holds. Figure 1a illustrates how we calculate the co-occurrence between concepts available in a path with a sample window size of $k = 3$. The red box represents the first window of the path (left-most window) in which the concept *Asteroid* co-occurs with the next $k - 1 = 2$ concepts in the path (*C.F. Gauss* and *Germany*). Since we use a symmetric co-occurrence measure, the next two concepts co-occur with *Asteroid* as well. The window then slides one step to the right, (blue box, right most window). We repeat this step until position n is reached. The resulting co-occurrence matrix is shown in Figure 1b – higher co-occurence counts are possible for larger data. Using this matrix, the relatedness between concepts can be determined by calculating a similarity (i.e., cosine) between two concept-vectors (see Section 3.2).

3.4. Datasets

We now introduce the datasets for our experiments and the ways in which they have been obtained.

3.4.1. Wikigame

This dataset is based on the online game *"TheWikiGame"*[9]. The platform offers users a multiplayer game, where the goal is to navigate from one Wikipedia page (the start page) to another Wikipedia page (the target page) which is linked to the start page through Wikipedia's underlying topological link network. The users

[9] http://thewikigame.com/

can leverage Wikipedia's directed link structure to reach their target node, but in some cases users also establish new links in their paths between articles that might not yet exist in Wikipedia's topological link network. This can happen when, for example, users use the back button in their browser to navigate to a previous article, and the current article does not have a link back to the previous one. One explanation for such behavior could be that users originally end up at a concept they are not happy with and decide that going another route may be a better idea. This is a rich feature of this dataset as it enables us to establish relations between concepts that we normally would not see using Wikipedia's link network. The logic of the game can be transported to any information network consisting of links between resources. If the user is presented with all links leading from one page to another the game can be applied and played in similar fashion.

A path in this dataset is the attempt of a single player to solve a game. We only consider paths where a user navigates through at least two pages and only if those pages are available in our Wikipedia dump consisting of concepts from the *main namespace* (see Section 3.4.2). Furthermore, we know which paths are *successful* – i.e., the user has reached the target concept – and which are *unsuccessful* – i.e., the user has failed to find a route to the target in the given timeframe. Table 2 shows some main characteristics of our Wikigame dataset. The adjusted dataset at hand consists of 1,799,015 navigation paths captured between 2009-02-17 and 2011-09-12. The distribution of path lengths is discussed and depicted later in Figure 3. We can see differences in the length distribution for successful, unsuccessful paths and all paths, but each distribution exhibits a peak at a length of around six.

3.4.2. Wikipedia

Wikipedia offers complete dumps of the English Wikipedia, and for our experiments, we chose a Wikipedia dump dated on 2011-11-07. The reason for this choice was that this was the dump closest to the timestamps in the Wikigame dataset (see Section 3.4.1) that was publicly available[10]. We obtained the present page-to-page network provided by this dump and limited it to links between pages from the main namespace and also to links between the distinct pages available in our Wikigame dataset. The reason for this is that we want to compare the paths through the network with the corresponding topological network; if we would leave the original network untouched, it would be impossible to assert whether

[10] Wikipedia only makes a specific amount of recent dumps available for download

the difference in our results are based on the type of paths or on the number of distinct pages in the Wikigame dataset.

3.5. Evaluation

To evaluate semantic relatedness, we compare our results to a gold standard dataset, specifically the *WordSimilarity-353* dataset (Finkelstein et al., 2002). The WordSimilarity-353 dataset consists of 353 pairs of English words and names and includes all 30 nouns of the *Miller and Charles dataset* (Miller & Charles, 1991) and most of the 65 pairs of the *Rubenstein and Goodenough dataset* (Rubenstein & Goodenough, 1965). Each pair was assigned a relatedness value between 0.0 (no relation) and 10.0 (identical), denoting the assumed common sense semantic relatedness between two words. For each pair of words, ratings of 16 different people were collected. Finally, the total rating per pair was calculated as the mean value of each of the 16 user's ratings. This way, WordSimilarity-353 provides a valuable evaluation base for comparing our concept relatedness scores computed on Wikipedia to an established human generated and validated collection of word pairs. In (Miller & Charles, 1991) it was also shown that the correlation coefficient between the two sets of ratings – i.e., the *Miller and Charles dataset* and the *Rubenstein and Goodenough dataset* – is 0.97. Hence, we can conclude that human knowledge about semantic similarity between words is very stable over a large time span and we can use them for evaluating our semantic relatedness

Table 2: Characteristics of TheWikiGame dataset

#Pages	360,417
#Games	361,115
#Users	260,095
#Paths	1,799,015
#Visited nodes	10,758,242
Average path length	5.98
Average #paths per user	6.92
#Successful paths	653,081
#Visited nodes of successful paths	4,116,879
Average successful path length	6.30
#Unsuccessful paths	1,145,934
#Visited nodes of unsuccessful paths	6,641,363
Average unsuccessful path length	5.80

calculations (Li et al., 2003).

Because WordSimilarity-353 consists of English words and names, we map them to an according Wikipedia concept. We use an adapted version of WordSimilarity-353 called *WikipediaSimilarity-353*, which contains a manual mapping and disambiguation step of words contained in WordSimilarity-353 to Wikipedia concepts (Milne & Witten, 2008). As a further step, we manually checked the mappings for correctness and modified some of the mappings accordingly[11]. For some word pairs it is not possible to map it to appropriate Wikipedia concepts[12]. By removing such pairs where we can not map one word, we end up with 314 concept pairs where we can cover a total of 308 pairs with the concepts available in our Wikigame and Wikipedia dataset (see Sections 3.4.1 and 3.4.2). The main reason for our choice of using manual mappings instead of for example, using sense pairs with maximal similarity, is that the main focus of our work is to show the viability of human navigational paths for calculating semantic relatedness and not the necessarily best working method to date. Milne (2008) shows in his work that the accuracy drops by a large margin if one does not use a manual mapping and relies on automatic disambiguation. This automatic disambiguation step itself is not trivial and can probably introduce a large negative bias to our results as this would make inference of the results difficult as we would not know if the possibly bad results are based on the simple disambiguation step or on the bad results of our method. In the remaining chapters, we will refer to *WordSimilarity-353* even if technically, we mean *WikipediaSimilarity-353*. Our final mapping can be found online on our website[13].

Finally, we compare two rankings. We extract the first ranking of the original scores available through WordSimilarity-353. We also create a similarity ranking for the corresponding word pairs on different paths corpora with our semantic relatedness method, using the cosine similarity. In the last step, we compare both rankings with the Spearman rank correlation coefficient as stated in Formula 1. Using the Spearman rank correlation as evaluation metric enables us to specify how closely our semantic relatedness scores are in terms of a ranked list on all WordSimilarity-353 concept pairs. If the rank correlation is close to 1 we nearly produce the same ranking as human judges.[14]

[11] For example, we had to correct some Wikipedia ids of concepts.

[12] For example there are no appropriate Wikipedia pages available for both the terms in the word pairs “Hotel reservation” or “Boxing round”.

[13] http://www.philippsinger.info/wikisempaths.html

[14] One needs to note that the smaller the gold standard is one compares to, the more difficult it

$$\rho = \frac{Cov(rg_{WS}, rg_{WP})}{\sigma_{rg_{WS}} \sigma_{rg_{WP}}} \in [-1; 1] \tag{1}$$

In this formula, rg_{WS} refers to the ranks in WordSimilarity-353, and rg_{WP} to our results. The σ_{rg_X} values is the standard deviation of both ranks. Bear in mind that ranks can also contain tied values, i.e., where two word pairs share the same similarity value. We made sure that our implementation can also handle such ties. We also calculated significance using a two-sided p-value which roughly indicates the probability that a uncorrelated system produces a ranking that has at least the same Spearman rank correlation as the one computed from the original ranking produced by our method. We will not explicitly specify the p-values for each calculation, as all p-values are below the significance level of 0.01. Hence, when we talk about the Spearman rank correlation, we actually refer to the calculated ρ.

4. Semantics of navigational paths

To study feasibility, we first investigate whether a corpus of human navigational paths through an information network – i.e., navigational paths taken from the Wikigame conducted on Wikipedia's link network – can contribute to computing semantic relatedness of concepts using the introduced concept co-occurrence (cf. Section 3.3) in Section 4.1. In Section 4.2 we compare the results to those obtained from several baseline corpora to show the additional benefit of human navigational paths.

4.1. Contribution of navigational paths to semantic relatedness

To show the usefulness of human navigational paths for calculating semantic relatedness we conduct our experimental steps as described in Section 3.3 where we not only use sliding windows of varying size *k* but also the principle that all concepts in a path co-occur with all other present concepts in the path on the corpus of all available paths taken from "TheWikiGame" – which we denote as a "none" window size. One can think of the "none" window size as a size that is always exactly as long as the path.

Table 3 presents the evaluation results for varying window sizes. We report the number of pairs (shown in column *#pairs*), for which we can calculate a semantic relatedness score (stated in column *ws353*). The reason why one can not

may get to judge the actual results.

always evaluate against each single pair of concepts is that there might not be co-occurrence information available for concepts of pairs using a specific window size – i.e., generally the larger the window size, the more pair scores we can calculate. A first observation is that the method of letting all concepts in a path co-occur with all other concepts in the path denoted as "none" performs worse than some specific sliding window sizes denoted in the table. This strengthens our assumption that the distance between two concepts in a path is crucial for calculating precise semantic relatedness scores as pointed out in Section 3.3. Furthermore, we can see that the best accuracy can be achieved using a window size of $k = 3$ or $k = 4$. Hence, letting the surrounding two or three concepts $(k-1)$ given a concept in a path co-occur with the concept seems to be the most precise co-occurrence representation for determining the semantic relatedness between concepts in our corpus of human navigational paths. Interestingly, this observation correlates with the distance often applied in graph based methods for word sense disambiguation, as reported in Navigli & Lapata (2010).

To investigate the usefulness of our approach of reporting results obtained from evaluating the scores of all possible WordSimilarity-353 pairs for a specific window size or corpus, we also repeat the experiments by using all 353 word pairs and setting the relatedness scores to zero if we can not cover a pair as this is frequently done in related work (see the last column in Table 4). However, this method introduces high negative bias to the results as we observe that not surprisingly, those window sizes or corpora perform better that can simply cover

Table 3: Semantic relatedness calculated on human navigational paths. Our corpus consists of all available Wikigame paths where different window sizes ($2 \leqslant k \leqslant 5$) as well as the principle that all concepts in a path co-occur with all other concepts in the path denoted by "none" were evaluated against the WordSimilarity-353 golden standard by calculating the Spearman's rank correlation coefficient between the produced rankings of each method and the ones of the WordSimilarity-353 gold standard.

window size	#pairs	ws353
none	299	0.649
2	236	0.638
3	275	0.709
4	286	0.718
5	293	0.690

more WordSimilarity-353 pairs. We also calculate statistical significance tests between the dependent Spearman's rank correlation coefficients produced by different window sizes for this evaluation method using a one-tailed hypothesis test for assessing the difference between two paired correlations (Steiger, 1980). While the results indicate no statistic significant differences between window sizes 3 to 5 it is clearly visible that we would e.g., prefer an window size of 5 over "none" (the p-value $5.2 * 10^{-5}$ of the test is below the significance level of 0.05). Summarized, this evaluation represents a pessimistic evaluation compared to our optimistic one which only evaluates against possible word pairs, as it is hard to judge whether better accuracy is based on more precise calculations of semantic relatedness or simply more well defined term pairs. To further strengthen our evaluation approach we limit the evaluation in Table 3 to those pairs available throughout all window sizes (236 pairs) – see fifth column in Table 4 – and we can observe the exact same trend as our optimistic evaluation approach showed. Finally, we also sample 100 random pairs 100 times and average the results again showing in the fourth column of Table 4 that the best accuracy can be achieved using a window size of $k = 3$ or $k = 4$ and making a strong point for our evaluation approach. This agrees with similar observations by Ito et al. (2008) when evaluating against different subsets of WordSimilarity-353 pairs that the trend of accuracy always stays the same. Also, Milne & Witten (2008) pick up on this point as they directly

Table 4: Semantic relatedness accuracy calculated in similar fashion as for Table 3. This time, we report a variety of different evaluation approaches: (a) "possible pairs" reports the same results as in Table 3 and represent our optimistic evaluation, (b) "100 pairs" reports accuracy by sampling 100 word pairs 100 times and averaging the results, (c) corresponds to the accuracy by using only those word pairs that can successfully be determined for all windows sizes and (d) "all pairs" fills in zero semantic relatedness scores for word pairs for which no score can be calculated and represents the pessimistic evaluation. The observations illustrate the usefulness of our proposed "possible pairs" method.

window size	#pairs	possible pairs	100 pairs	236 pairs	all pairs
none	299	0.649	0.630	0.632	0.548
2	236	0.638	0.633	0.638	0.560
3	275	0.709	0.692	0.694	0.588
4	286	0.718	0.697	0.695	0.587
5	293	0.690	0.690	0.692	0.589

show that as they only include well-defined term pairs to their evaluation, they can achieve the appropriate results.

As the goal of this work is not to achieve the best possible semantic relatedness scores in comparison to related work techniques, but rather to identify whether and if so, human navigational paths can contribute to this task and to find the most appropriate window size and path corpus, we only report results obtained from applying our optimistic evaluation procedure which evaluates the scores of all possible WordSimilarity-353 pairs for a specific corpus. Note that we will also only cover a very small amount of pairs later on for our sampling strategies which makes the other evaluation methods not applicable – i.e., only using the same intersection of pairs for all methods would limit the gold standard tremendously (max. 30 pairs) and using all pairs by filling in zeros for missing word pairs would have high negative influences on methods that can only cover a small amount of pairs due to lack of data. This choice is based on abovementioned investigations and observations and gives us a logic way to evaluate our work. Due to tractability, we focus on window size $k = 3$ for the rest of this paper[15].

Table 3 demonstrates that human navigational paths contain information relevant for calculating semantic relatedness between concepts by exhibiting high quality relatedness evaluated against WordSimilarity-353. We investigate the additional benefit of the paths at hand to several baseline corpora next.

4.2. Additional benefit of navigational paths

As our human navigational paths of "TheWikiGame" are basically subsets of the underlying topological link network we need to investigate whether the observed effects are based on human intuitions and patterns while navigating or if automatic extractions of paths from the link network can produce similar or even better results. By doing so we can also investigate which role the rich topological link network plays for calculating semantic relatedness on paths.

To get first insights, we highlight basic properties of the Wikipedia link structure that we have studied, and the corresponding navigational paths that we have obtained in Figure 2. The figure contrasts the degrees of nodes in a subset of Wikipedia with the number of clicks on these nodes in a baseline random walk and in human navigational paths. As we see, the number of clicks on nodes from

[15] Note that a window size of $k = 4$ is just by a small margin more precise than a window size of $k = 3$ and the reason for only reporting results for $k = 3$ is based on faster runtime and better possibilities for interprating the results or looking into fingerprints. Nevertheless, we have also conducted further experiments by using a windows size of $k = 4$ which exhibit similar patterns.

human navigational paths differs significantly from (i) the network topology and (ii) the clicks generated by a random walk. On the one hand, we can see that human navigation tends to focus on a few nodes more heavily than a random walk on the network topology would lead us to expect, while on the other hand, they seem to place less focus on a wider range of nodes[16]. As both random walk and human navigational paths are basically subsets of weighted links, we can see that the weights emerging from user's choices during the game differ from the weights produced by a random walk. Hence, we want to explore whether these differences resulting from actual human navigation in information networks provide additional value for calculating semantic relatedness in comparison to navigation done by an automatic agent. To do so, we compare the corpus of paths from "TheWikiGame" with several baseline corpora which we introduce in the following sections. Finally, we present the results in Section 4.2.5.

4.2.1. Topological neighbor paths

A rather simple baseline for comparison consists of artificial sub-paths taken from Wikipedia's link network limited to concepts available in our Wikigame dataset (see Section 3.4.2). Given Wikipedia's topological (limited) link graph $\mathbb{W}_{wg} = (V_{wg}, E_{wg})$ with vertices V_{wg} and directed edges $E_{wg} = \{(v, w)|v, w \in V_{wg}\}$, we generate all possible paths of length three, where every node still lies in V_{wg}. This gives us the following set of paths $\mathbb{P}_{tb} = \{(u, v, w)|(u, v), (v, w) \in E_{wg} \cap V_{wg} \times V_{wg}\} \subset \mathbb{P}$.

The reason for choosing paths with the length three for this topological baseline corpus is that we focus on a window size of $k = 3$ – i.e., a concept co-occurs with the neighboring $k - 1 = 2$ concepts in a path – throughout this work (see Section 4). Hence, with this corpus of artificial paths we can calculate all possible co-occurrences between concepts in a window of size $k = 3$. For this baseline, we will not only report results based on co-occurrence vectors with their respective co-occurrence counts, but also based on *binary vectors* – i.e., two concepts get a co-occurrence count of one if they appear in at least one single path of length three together – ignoring the number of co-occurrences and thus controlling the vast amount of artificial paths. This enables us to investigate the influence of the degree of concepts on the results – note that again the extracted corpus of paths is a weighted subset of the plain Wikipedia link structure where the weight is in-

[16] The *Kullback-Leibler divergence* (0.738) and the *Spearman rank correlation* (0.130) between the click distributions of a random walk vs. human navigation indicate that there is indeed a significant difference between the distributions.

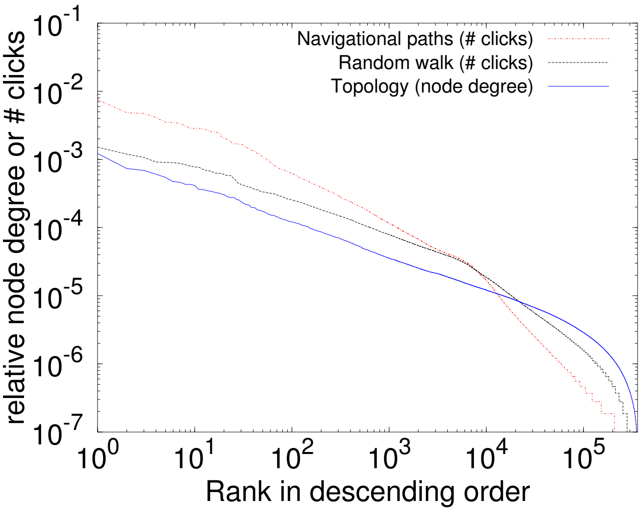


Figure 2: Properties of the Wikipedia link structure that we have studied and corresponding navigational paths that we have obtained. The figure compares the distribution of node degrees of the underlying Wikipedia topology (blue solid line) with the relative click frequency on the same set of nodes obtained from a random walk (black dashed line) and from navigational paths obtained from the Wikigame (red dotted line). The ranks on the x-axis are based on the corresponding node degree or #clicks for the corresponding node in descending order – e.g., the node with rank 1 has the highest degree.

fluenced by the degree of each node (e.g., a node with an out-degree of 4 is more likely to get higher co-occurrence counts than a node with an out-degree of 1).

4.2.2. Biased random walk paths

We aim to compare the usefulness of human navigational paths to artificial paths (e.g., produced by an algorithm) as another kind of baseline. Therefore, we perform a biased random walk through Wikipedia's underlying plain topological link structure preserving some of the structural information taken from our Wikigame paths. For each path, we select the start node and initialize a random walk on Wikipedia's link network limited to concepts available in the Wikigame. The random walk then walks freely through this network by choosing a random outlink available for a concept. The walk stops when the similar path length as the corresponding Wikigame path is reached. By doing so we end up with a corpus of paths that approximately has the same number of visited pages as “TheWikiGame” corpus, but exhibits dissimilar link weights. If the walker reaches a concept with no out-link, it goes back one position and tries another path. The relative concept click frequency of the resulting paths can be seen in Figure 2. We call the resulting set of random paths $\mathbb{P}_{random}$.

4.2.3. Permuted Wikigame paths

To understand how important the underlying link structure is for the task of calculating semantic relatedness on navigational paths and also to explore how much impact the sequence of concepts in a human navigational path has, we create so-called *permuted paths*. In these paths, we are still leaving the position of a concept in a path intact, but swap it with a node on the same position of another path and by doing so we detach the node with preceding and succeeding nodes of the path. For a given path $p = (v_1, \ldots, v_n) \in \mathbb{P}$, we randomly choose another path $q = (w_1, \ldots, w_m)$ and randomly swap a node in p with the corresponding node at the same position in q. We receive two new paths $p' = (v_1, \ldots, w_i, \ldots, v_n)$ and $q' = (w_1, \ldots, v_i, \ldots, w_m)$ where we lose the semantic information around the newly inserted node. Again, we preserve as much structural information as possible of our game paths while randomizing the semantic related information. We call the resulting path set $\mathbb{P}_{permuted}$. It is important to note in this scenario, nodes might not be linked from their predecessor or to their successor on the underlying Wikipedia topology. These newly created paths are called $\mathbb{P}_{permuted}$ and contain exactly as many paths as $\mathbb{P}$.

4.2.4. Swapped Wikigame paths

The purpose behind this method is to keep the link structure of Wikipedia intact but to swap out parts of a supposedly meaningful path with parts of another path. Our method works as described in the following: For a given path $p = (v_1, \ldots, v_{i-1}, v_{mid}, v_{i+1}, \ldots, v_n)$, we select another path $q = (w_1, \ldots, w_{j-1}, v_{mid}, w_{j+1} \ldots, w_m)$ with maybe a different length, but with the property that the node v_{mid} is in the middle of both paths. We cut both paths in half and exchange the back part of p with the one of q in such a way that we receive the new paths $p\prime = (v_1, \ldots, v_{i-1}, v_{mid}, w_{j+1}, \ldots, w_m)$ and $q\prime = (w_1, \ldots, w_{i-1}, v_{mid}, v_{j+1}, \ldots, v_m)$. The newly generated paths are called $\mathbb{P}_{swap}$ and contain exactly as many paths as $\mathbb{P}$.

4.2.5. Results

Table 5 presents the results using a window size of $k = 3$ with all available Wikigame paths and our baseline corpora as described above. In column *#paths* one can see the number of paths available for each corpus and in column *length* the total accumulated length of all paths in the corpus. Finally, in column *#pairs* we can see the number of pairs of the WordSimilarity-353 dataset where we can successfully calculate the semantic relatedness measures and the final *Spearman rank correlation* to WordSimilarity-353 is shown in column *ws353*. Further insights from these investigations are discussed next.

Wikipedia topology alone is useful: We know from other semantic analysis methods, that the Wikipedia topology alone provides useful information. For confirmation, we evaluated the scores obtained from our *Permuted Wikigame paths corpus*. The corresponding results confirm that we lose semantic preciseness when ignoring the original link and navigation structure. Keeping the original structure intact, but swapping parts of the paths – see *Swapped Wikigame paths* and the corresponding description above – we can see that the original navigation by a user has a high impact on the achieved accuracy, but that we can still achieve reasonable results by leaving the underlying link structure and partly navigational patterns intact. We can also see that *Biased random walk paths* perform similar to our *Topological neighbor paths corpus*. This is not surprising, as the random walks freely navigate the topological link network, even though they are biased towards a specific path length and are initialized by a given start node.

Human navigation paths improve results: A first observation is that the Wikigame path results outperform the baselines by a relevant margin – for example, it outperforms the best baseline method *Swapped Wikigame paths* by 0.041

Table 5: Comparison of semantic relatedness calculations using a window size of $k = 3$ evaluated against WordSimilarity-353 on all Wikigame paths with several baseline corpora.

Corpus	#paths	#pairs	ws353
All Wikigame paths $\mathbb{P}$	1,799,015	275	0.709
Topological neighbor paths $\mathbb{P}_{tb}$	6,042,578,644	308	0.659
Topological neighbor paths $\mathbb{P}_{tb}$ binary	6,042,578,644	308	0.485
Permuted Wikigame paths $\mathbb{P}_{permuted}$	1,799,015	292	0.381
Swapped Wikigame paths $\mathbb{P}_{swap}$	1,799,015	273	0.668
Biased random walk paths $\mathbb{P}_{random}$	1,797,326	274	0.660

(0.709 vs. 0.668). When looking at the *Topological neighbor paths corpus*, we can also see that Wikipedia's inherent link structure already can be used as a powerful resource for calculating semantic relatedness using our methodology. In order to see how the number of co-occurrences between concepts influences the semantic relatedness, we have also performed an analysis on the same corpus of topological neighbor paths, but this time we do not count how often two concepts co-occur, but only represent the co-occurrence state with a binary value – we can refer to this as the *plain structure*. We can now see that the accuracy evaluated against WordSimilarity-353 drops by a significant amount (from 0.659 to 0.485) indicating that the number of co-occurrences between concepts effects our method. We can observe from this that the weighting of links in a path corpus has high impact on the accuracy we can achieve.

With this initial exploration, we can conclude that human dynamic navigational paths on Wikipedia can contribute to computing semantic relatedness, but they are based on an already powerful network topology. The weighting provided by users' choice during navigation exhibits the most precise information for determing semantic relatedness between concepts. Next, we want to identify what kind of navigational paths are most useful for that task.

5. Path selection experiments

Human navigational paths can be characterized along many dimensions. For example, there exist *successful paths* where users were able to successfully reach the specified target nodes, while on *unsuccessful paths* users could not reach their goal. Other path characteristics may mostly move along high degree (vs. low degree) nodes. Figure 3 shows the distribution of path lengths in all paths (black line), only in successful paths (red line) and only in unsuccessful paths (blue line). Only looking at such path length distributions, we can already see that such distinct path types exhibit different features. We want to explore these differences and investigate their usefulness for the task of calculating semantic relatedness, e.g., investigate whether a subset of only successful paths is more useful than a subset of only unsuccessful paths.

This gives rise to a number of interesting questions related to different navigational paths, such as (a) *Are all navigational paths equally useful for computing semantic relatedness?* and (b) *If some navigational paths are more useful, what are the characteristics of these paths and how can they be exploited?* To analyze these and other questions, we begin our investigations by taking the corpus of all *successful paths* (which is the smaller set) and extract a random subset of *unsuccessful paths* of equal size, containing the same number of visited pages.

Similar to Section 4, we use a window size of $k = 3$ for our co-occurrence calculation and evaluate the relatedness scores against WordSimilarity-353; the results can be seen in Table 6. From that table, we see that a smaller subset of our corpus of all Wikigame paths $\mathbb{P}$ can perform remarkably well – compare with Table 5. Somewhat surprisingly, we see that a corpus of *unsuccessful paths* performs better than a corpus of *successful paths* with the same total number of visited concepts. A possible explanation for this behavior is that unsuccessful paths contain the behavior of mostly inexperienced users who try to follow nodes whose meanings are very close and hence, remain on a narrow semantic field which may also lose them the game. On the other hand, successful players might navigate through more distant concepts or very central concepts like "United States" which are common strategies for winning a game. Further investigations are necessary in order to explain this behavior in detail, which is not in the scope of this work.

Regardless the exact explanation of this behavior, the results suggest that subsets of paths with specific characteristics yield different results. This leads to the idea of investigating whether smaller sets of paths according to specific path characteristics can perform similarly or even more precise in regard to our relatedness calculations on the whole set of paths. In the following section, we will explore this by conducting different path selection experiments.

5.1. Characteristics of Paths

We introduce several measures $m : \mathbb{P} \to \mathbb{R}_0^+$ to characterize any path p in our corpus of paths $\mathbb{P}$. Each distinct measure makes use of a path characteristic, depending on the visited nodes, which actually characterize the path. The resulting measures will be subsequently used in section 5.2 to create path selections.

In the following, we will elaborate each of the different measures in greater detail. Let $p \in \mathbb{P}$ be an arbitrary path represented by the sequence of nodes

Table 6: Comparison of semantic relatedness calculations using a window size of $k = 3$ evaluated against WordSimilarity-353 on all Wikigame paths with several baseline corpora.

Corpus	#paths	length	#pairs	ws353
Successful Wikigame paths	653081	4116879	230	0.636
Unsuccessful Wikigame paths	710374	4116879	257	0.683

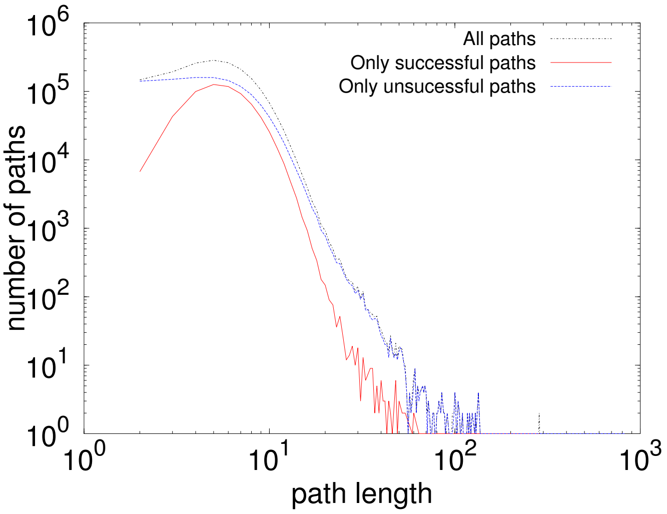


Figure 3: Illustration of the distribution of path lengths in all human navigation paths (black dotted line), only in successful paths (red solid line) and in unsuccessful paths (blue dashed line).

$(v_1, \ldots, v_n)$.

In- and outdegree. For a path p, we determine the in- and outdegree for each concept v_i in p derived from Wikipedia's complete topological link network. The idea behind this characteristic is to differentiate hubs and strongly connected concepts from dead ends and rather weakly connected concepts. The measure is calculated as ($m_{outdegree}(p)$ is defined analogously):

$$m_{indegree}(p) = \frac{1}{len(p)} \sum_{k=1}^{n} indegree(v_k).$$

Ratio. This measure represents a ratio of in- and outdegree for each node in a corpus of paths smoothed by the square root of the indegree (see (Trattner et al., 2012)). This characteristic is motivated by the notion that a page with e.g., 200 inlinks and 100 outlinks should be more important than a page with two inlinks and one outlink. If the outdegree for a node is zero, we set the ratio to zero as well. $ratio(v)$ is calculated in the following way for a node v:

$$ratio(v) = \frac{indegree(v)}{outdegree(v)} \cdot \sqrt{indegree(v)}.$$

Thus, the value of a path p is determined by

$$m_{ratio}(p) = \frac{1}{len(p)}\sum_{k=1}^{n} ratio(v_k).$$

TF-IDF. Interpreting a path as a document and the concepts present in a path as terms, we use the well known *tf-idf* scores (cf. (Salton & Buckley, 1988)) of each node in a path as a further characteristic. The idea behind this characteristic is that we can identify paths that include many concepts that are very important for the individual path compared to all other paths in the corresponding corpus. Hence, for each path p, we again take the mean of all tf-idf values in the path:

$$m_{tfidf}(p) = \frac{1}{len(p)}\sum_{k=1}^{n} tfidf(v_k).$$

Length. Finally, we use the length of a path p – i.e., the number of concepts visited in a path – as a last characteristic:

$$m_{length}(p) = len(p).$$

Our motivation for taking the length of a path as a characteristic is the notion that longer paths potentially contain more information because of more co-occurrences between concepts of the paths. Furthermore, we could observe in Figure 3 different path length characteristics for different types of Wikigame paths, which is interesting to investigate in greater detail.

5.2. Path selection strategies

Based on the characteristics described in Section 5.1 we now select smaller sets of paths according to abovementioned path characteristics. We investigate whether the relative performance of reduced corpora of paths $\mathbb{P}_m$, based on the accuracy of our relatedness scores, increases or decreases, compared to the performance of our complete set of paths $\mathbb{P}$, in analogy to Koerner et al. (2010).

For each characteristic, we calculate ten subsets of increasing size where the tenth subset corresponds to the set of all available Wikigame paths. The sizes of our subsets are calculated by the number of visited nodes inside the subset. If we consider the sum of all nodes $node\_sum = \sum_{p \in \mathbb{P}} len(p)$, a path selection of e.g., 10% does not necessarily contain $0.1 \cdot |\mathbb{P}|$, but rather $0.1 \cdot node\_sum$. More formally, we can express it as follows: Consider an ordered list $l_m = (p_1, \ldots, p_n)$ of paths, generated by a measure m. A selected subset $\mathbb{P}_x^m$ of size x for measure m can be expressed as:

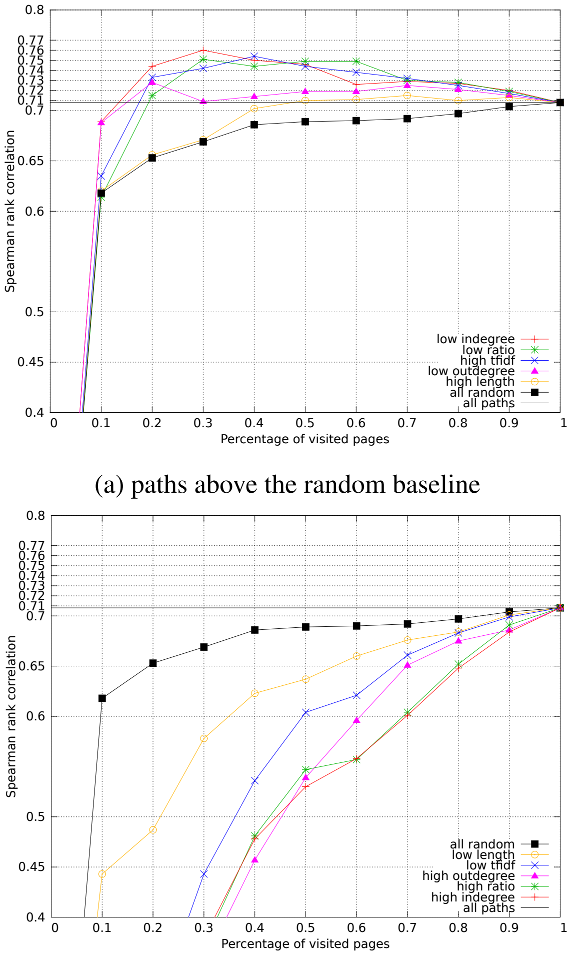
$$\mathbb{P}_x^m = \left\{ p_k | k = \max \left\{ s | \sum_{j=1}^{s} len(p_j) \leqslant \frac{x}{100} \cdot node\_sum \right\} \right\}. \tag{2}$$

Thus, a potential path selection with very long paths consists of fewer actual paths than a selection with mostly short paths, but both sets contain roughly the *same amount of visited nodes*. This renders the selection process more fair than just pure path counting as it enables us to fairly compare two corpora of the same size selected based on different path characteristics. Each selection process generated subsets consist of $x = \{10, 20, \ldots, 90\}\%$ of all visited pages. By proceeding with this selection process, the first subset – i.e., the 10% subset – consists of paths with the lowest measures for a corresponding characteristic – e.g., paths with the lowest mean indegree. Furthermore, we also revert the ordered list l_m in order to get a ranking $l_m^{rev} = (v_n, \ldots, v_1)$ where the small subsets contain paths with higher measures for a specific characteristic – e.g., paths with the highest mean indegree. After the generation of the path ordering lists and the path selection process described above, we run our semantic evaluation for each of these subsets.

Furthermore, we create a baseline for each individual split to learn whether the distinct accuracy results are genuinely dependent on the corresponding path selection process based on several characteristics. We shuffle the corpus of paths independently and randomly ten times in order to remove the original ordering in the complete set of paths. For each of these ten independent shuffles, we extract subsets according to the selection process described above. We end up with ten selections for each subset containing $x = \{10, 20, \ldots, 90\}\%$ of the visited pages. Finally, we perform our semantic analysis and evaluate the results accordingly for each selection and subset. We average the results for each subset based on the sum of selections for the corresponding subset and report the results in the following section; we will refer to this baseline as *random baseline*.

5.3. Results

In Figure 4 we present the results obtained from our individual sub-corpora of navigational Wikigame paths using our selection strategies pointed out in Section 5.2 based on characteristics of paths – or to be precise: characteristics of concepts inside paths averaged for each path – described in Section 5.1. Figure 4a illustrates selections where we can achieve better accuracy – i.e., Spearman rank correlation evaluated against WordSimilarity-353 – than using random selections



(a) paths above the random baseline

(b) paths below the random baseline

Figure 4: Semantic relatedness calculated on different path selections. Larger values on the y-axis correspond to higher Spearman rank correlation with the WordSimilarity-353 dataset. The black horizontal line depicts the result for the entire set of paths. Figures 4a and 4b show the results of different selection strategies. Figure 4a shows selection results with better-than-random performance while Figure 4b shows results with worse-than-random performance. In Figure 4a, we can see that only a small subset of 30% low indegree paths produces more precise semantics than the whole path corpus $\mathbb{P}$ would (scoring a rank correlation of 0.760 to the WordSimilarity-353 dataset). Paths characterized by low in- and outdegree always perform better than a random baseline, while their counterparts, starting from high degrees, perform significantly worse. Similar patterns can be observed when selecting paths according to their tf-idf values.

of all Wikigame paths – i.e., *random baseline* (black line, ■) – while Figure 4b shows selections performing worse. The horizontal black line with a Spearman rank correlation of 0.709 shows the results achieved when taking a corpus of all Wikigame paths (see also Table 3). For all selections we use a window size of $k = 3$ for our co-occurrence and subsequent semantic relatedness calculations. Our key findings are discussed next.

Intelligent path selection improves semantic relatedness. A first observation when looking at Figure 4a is that smaller random path selections do not lead to a similar or better accuracy (black line, ■), but that we indeed can find smaller corpora of navigational paths – selected on several characteristics – that perform equally or better than the complete corpus of Wikigame paths (that reaches an accuracy of 0.709). By incrementally adding paths with the lowest average indegree of their concepts, we can achieve the highest Spearman rank correlation with a sub-corpus of only 30% of all Wikigame paths (red line, +). The respective accuracy of 0.760 outperforms the accuracy of the whole Wikigame corpus by about 5% while covering less than a third of all visited pages in the complete corpus. Contrary, we can see in Figure 4b that a reverse accumulation of paths, beginning with those having a high average indegree (red line, +), leads to much worse accuracy compared with the random baseline and as well as with the accuracy of the complete corpus. A possible explanation for this is that low indegree nodes represent concepts that do not seem to be hubs nor exceptionally abstract concepts in comparison to high indegree nodes. Also, high indegree concepts may have much more co-occurrence counts with several other concepts while low indegree concepts may only have co-occurrence connections to a few very specific concepts (even when looking at a window size of $k = 3$). Hence, the co-occurrence vectors may be sparser, but more precise and this may enable us to calculate more accurate semantic relatedness scores. If we look deeper into the paths included in our selection corpora we can see that paths with the highest average indegree all include the concept *United_States* which is on the one hand, the most central concept in Wikipedia's topological link network, and on the other hand, also by far the most often navigated concept in our Wikigame paths. Hence, this concept co-occurs with many others and is no suitable descriptor for determining the semantic relatedness between concepts while paths with the lowest average indegree contain more variety and also more descriptive co-occurrences. To summarize: **Small selections of low indegree paths exhibit more fine-grained and precise semantics than the set of all paths.**

To give an example we illustrate in Figure 5 the concept co-occurrence vectors for the concepts *Vodka*, *Brandy* and *Bread* on the one hand, using our best

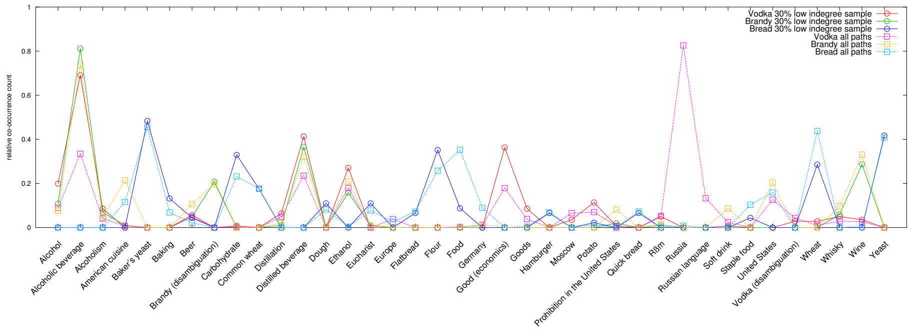


Figure 5: Semantic "fingerprints" for the concepts *Vodka*, co-occurrence count of fifteen to the corresponding concept. All counts are normalized by the L2 norm of the vector and fingerprints for a 30% low indegree selection (solid lines) and the full set of paths (dashed lines) are shown. The 30 % low indegree selection exhibits more fine-grained and precise semantics than the set of all paths.

overall performing corpus of 30% low indegree paths (solid lines, ○) and on the other hand, deriving the information from the all path corpus (dashed lines, ■). For visualization purposes the vectors are reduced in dimensionality by only representing co-occurrences to concepts where at least one vector exhibits a count of larger than 15. Furthermore, all counts are normalized by the L2 norm of the complete vector. In Figure 5 we can see that the concepts *Alcoholic beverage*, *Distilled beverage* and *Ethanol* exhibit similar peaks for the concepts *Vodka* (red solid line, ○) and *Brandy* (green solid line, ○) for the corpus of 30% low indegree paths, while having only few diverse peaks. We can observe that these common peaks contribute a lot to the high cosine similarity of 0.8043 that we can compute with this subset for the corresponding concept pair. This score agrees extremely well with the human score of 8.13 present in the WordSimilarity-353 dataset. In contrast, we can see that there are only a few similar normalized co-occurrences for the concepts *Vodka* (pink dashed line, ■) and *Brandy* (orange dashed line, ■) using the corpus of all paths and that the concept *Russia* exhibits a large diversity regarding the co-occurrence patterns for both concepts negatively influencing the relatedness score resulting in only 0.4205. The co-occurrence vectors for the concept *Bread* show for both corpora – i.e., 30% low indegree paths (blue solid line, ○) and all paths (turquoise dashed line, ■) – no common peaks to both other concepts resulting in extremely low relatedness scores. We can see from this, that our selection of low indegree paths exhibits much more fine-grained patterns for the concept pair *Vodka* and *Brandy* reaching also a higher relatedness score than our corpus of all paths by still keeping low scores for concept pairs, that are not semantically related.

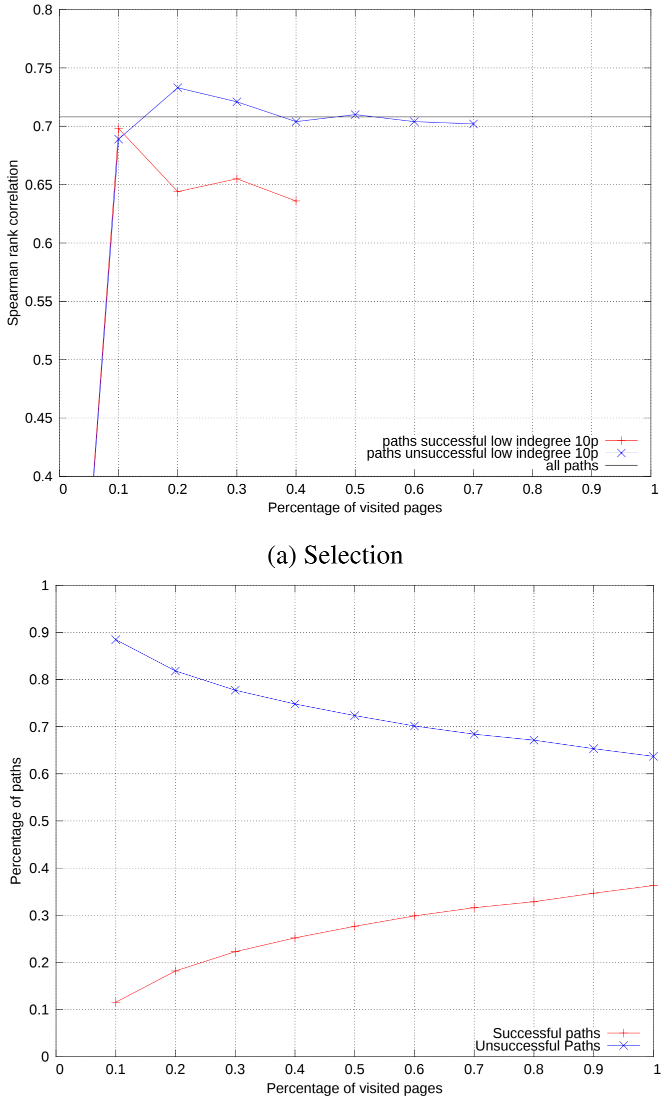
Other degree based selection strategies and corpus based characteristics (e.g., tf-idf) can also improve accuracy. Similar observations as above can be seen by selecting according to the average outdegree of paths starting with the lowest value depicted in Figure 4a (pink line, ▲). Smaller selections can outperform the corpus of all paths, but we can not achieve as good results as with our 30% selection of low indegree paths. Again, the opposite occurs for the reverse selection of paths starting with those having a high outdegree shown in Figure 4b (pink line, ▲) – i.e., all selections perform worse than the baseline and the complete corpus. Selections based on the average *ratio* of paths (green line, ✳) not surprisingly show similar patterns as the selection according to in- and out-degree, but indicate that a selection according to the average indegree of paths can achieve higher accuracy than using a combined measure. Selection strategies based on the *tf-idf* values of nodes inside paths indicate that we can strongly outperform the baseline and the target accuracy of a corpus of all paths for several sub-corpora incrementally adding paths with a high average tf-idf value shown in Figure 4a (blue line, ╳). Contrary, selecting paths with low tf-idf scores never reaches the accuracy of the random baseline as we can see in Figure 4b (blue line, ╳). Low average tf-idf valued paths exhibit similar patterns than those with a low average indegree. The difference though is that this measure is only corpus dependent and ignores characteristics of the underlying topological link network and this may exhibit advantages for specific scenarios. Finally, we can see from both illustrations in Figure 4 that a selection according to the length of paths (orange line, ◯) produces just three sub-corpora of paths – i.e., 70% to 90% selections of longest paths – that can slightly outperform the corpus of all paths.

A combination of successful and unsuccessful paths produces more precise semantics than using unsuccessful paths only. Our initial experiments showed that a corpus of unsuccessful paths outperforms a corpus of successful paths in regard to the accuracy of our semantic relatedness scores (see Table 6). Now that we know that a path corpus with lower indegree paths works better one possible reason for the better performance of unsuccessful paths might be that the average indegree of unsuccessful paths is lower as the average indegree of successul paths as we have investigated. However, with the observation that there are more intelligent ways of selecting a corpus of paths accordingly (e.g., by selecting low indegree paths), the question arises if we can furthermore improve the preciseness of semantic relatedness calculation by performing a similar selection just on the corpus of unsuccessful paths. To this end, we use our best performing characteristic measure – namely the *indegree* – and select in the typical way sub-corpora of unsuccessful paths starting with those having the lowest mean indegree. We do the same selection for successful paths to be able to compare both subsets. Again, we accumulate the number of paths in a selection towards the total number of visited nodes of the corpus of all paths; we end up with more selections for unsuccessful paths than for successful paths as we have a larger fraction of unsuccessful paths.

In Figure 6a we identify that we can outperform the horizontal black solid line indicating the accuracy obtained from a corpus of all Wikigame paths. The best results can be achieved by using a 20% split of only unsuccessful paths (blue solid line). While this accuracy of 0.733 outperforms the whole set of all paths, we still get a better result by selecting the whole corpus in a similar fashion as depicted in Figure 4a, where we could reach an accuracy of 0.760. When we now look deeper into the subsets of low indegree based selections calculated for the complete dataset, we see that around 25% of the paths inside the best performing 30% low indegree sub-corpus (selected on all paths) are successful paths (see Figure 6b). While unsuccessful paths tend to exhibit characteristics that make them more useful for computing semantic relatedness, we find that overall a combination of successful and unsuccessful paths produces the best results. The results also suggest that other characteristics such as the indegree and not success are better suited for selecting good subsets when performed on the whole set of paths.

Evaluating against other gold standard datasets confirms our observations. Throughout this section we have only used the *WordSimilarity-353* dataset as a gold standard for our evaluations. The reason for this choice was that it is a widely used gold standard for evaluating semantic relatedness scores against human judgements. Nevertheless, there also exist other prominent datasets similar to WordSimilarity-353: (a) the *Miller Charles gold standard* (Miller & Charles, 1991) (30 overall word pairs) and (b) the *Rubenstein Goodenough gold standard* (Rubenstein & Goodenough, 1965) (65 overall pairs). In order to triangulate our observations, we conducted the same experiments on both datasets by mapping words to concepts manually and calculating Spearman rank correlation. Again, we make our mappings available online[17]. The results for both gold standards are illustrated in Figure 7. Again, we can clearly see that we can outperform the accuracy of the complete set of paths by sampling smaller sets affirming the patterns

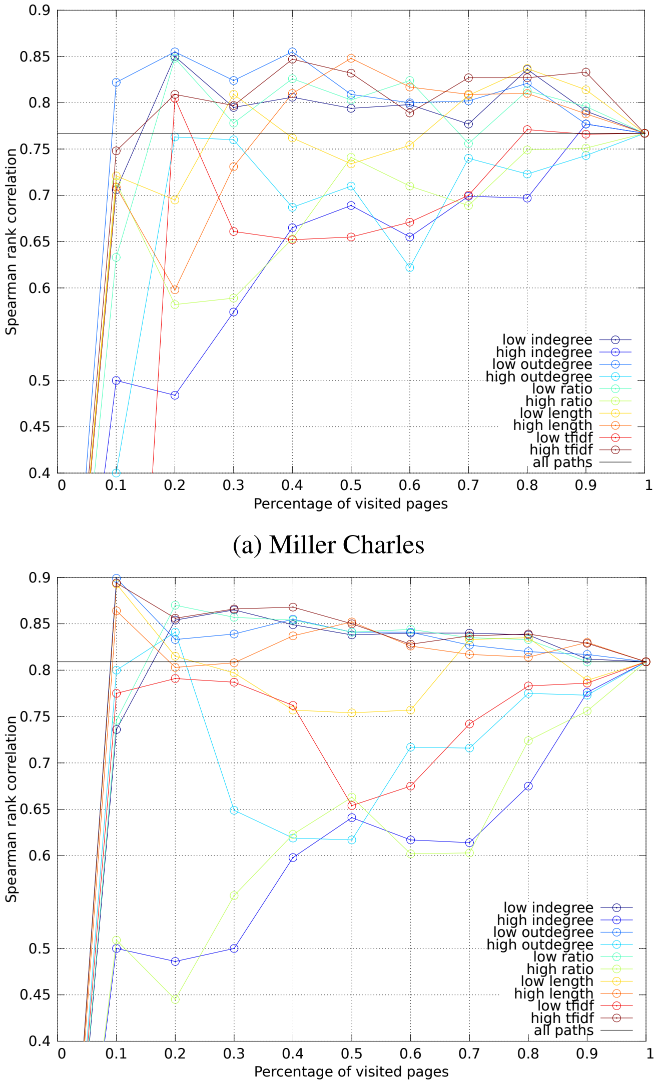
[17] http://www.philippsinger.info/wikisempaths.html



(a) Selection

(b) Proportion

Figure 6: Effect of successful / unsuccessful paths: (a) shows successful (red solid line) and unsuccessful paths (blue solid line) selected according to their average indegree starting with low indegree paths and their respective Spearman rank correlation evaluated against WordSimilarity-353. (b) shows the percentage of successful (red solid line) and unsuccessful paths (blue solid line) for our best performing selection of 30% low indegree paths (see Figure 4a). While path selection based on unsuccessful paths performs better than a selection of successful paths, we can see that



(a) Miller Charles

(b) Rubenstein Goodenough

Figure 7: Semantic relatedness evaluation for our sampling strategies evaluated against both the Miller Charles and Rubenstein Goodenough gold standards calculated and illustrated in similar fashion as in Figure 4. Again, we can see that specific samples can outperform the corpus of all Wikigame paths by excelling their corresponding Spearman rank correlations of 0.767 for the Miller Charles dataset and 0.809 for the Finkelstein Goodenough dataset.

observed in these experiments. This indicates that such sampling strategies can help us to remove paths with some kind of *semantic noise* by e.g., ignoring paths with a high average indegree of their visited concepts. By doing so, we can produce more precise semantics out of navigational path data. However, we need to take the results for these two additional evaluations with caution, as both gold standards are very limited in their number of word pairs they cover. We also can only capture at maximum 21 pairs for the Miller Charles and 40 pairs for the Rubenstein Goodenough dataset, while some samples can only cover a very low amount of pairs. Hence, this may also give rise to the slight unstable results in Figure 7 as sometimes the samples might simply capture very well-defined concept pairs while leaving others out. Contrary, this is not the case for the WordSimilarity-353 dataset where much more word pairs are available and where we can also cover much more pairs for all sub-samples.

6. Discussion and conclusions

To the best of our knowledge, this work represents the largest and most comprehensive effort to study semantics in human navigational paths to date. We (i) systematically evaluated information on $\sim$*1.8 million human navigation paths* captured via a semi-controlled navigation experiment against baselines that use the Wikipedia topology only or alternate the human navigational paths at hand and we (ii) evaluated the results against common reference datasets of relatedness. The main contributions of our work are the following. (1) Our experiments further indicate that such human navigational paths can represent a viable source for calculating semantic relatedness between concepts in information networks. (2) We show that semantic relatedness calculated based on human navigational data may be more precise than semantic relatedness computed from Wikipedia's link structure alone and (3) we find that not all navigational paths are equally useful. Intelligent selection of navigational paths based on path characteristics can improve accuracy.

If we compare our results to those obtained by previous works evaluated on the same full gold standard (see results from some well-known methods in Table 1) we can observe that we can match the accuracy of existing methods (our best score ends up at 0.76). Yet, there are obstacles in comparing the results to other methods directly. The main evaluation process of most of the related work remains a black box. Only slight adoptions to the Wikipedia dump used – e.g., by removing low degree concepts as ESA does – can already change the outcome tremendously. As the goal of this work is not to achieve the best performing method but rather detect signals in the data and show the usefulness of our approach we will not directly try to compare us with other works due to abovementioned reasons.

The method of leveraging human navigational paths using co-occurrence information presented throughout this work could also provide opportunities for improving existing content based methods in the sense of complementary information. For example, we could easily enrich existing co-occurrence based methods by interpolating the information extracted from human navigational paths. This would be a great way to incorporate pragmatic patterns to the content itself. In future, we want to concretely investigate the usefulness of such an approach by using navigational information by humans as additional signals for semantic relatedness for existing approaches.

A main limitation of this work is that we focus on human navigational paths derived from a game – namely "TheWikiGame". The game design itself may affect the structure of the paths and the resulting semantic relatedness scores. Some possible constraints of the game are: (a) a random choice of start and target nodes – hence, users also do target based navigation instead of pure exploration navigation, (b) users have a time constraint while navigating or (c) users tend to evolve strategies in order to win a game that may be counterproductive in terms of specifying semantic relatedness. Contrary, one could argue that real navigation more focuses on the goal of getting as much information as possible. One could also argue that such real human navigational data can even be more useful as humans may take more time for checking the current page and the next link would be chosen more accurately. They may also navigate on a more semantically narrow path. Nevertheless, the human navigational Wikigame paths present an abstraction of real user navigation in information networks and provide a further signal that such data can indeed be very useful for calculating semantic relatedness. In future we want to investigate human navigational paths in a less controlled navigational setting and investigate whether such paths can also contribute as much – or as hypothesized even better – as the data at hand indicates.

As mentioned throughout the work, our Wikigame paths are basically a subset of weighted links. Even though our results suggest that these paths can be more precise than artificial paths derived from Wikipedia's topological link network – note that these paths are again path sub-corpora of weighted links, where the weight is determined by an algorithm – we do not know if there might be a configuration of weights that leads to better results. Nevertheless, it is a complicated and not trivial task to automatically determine such a configuration of weights. As we can see from our experiments, human navigational paths seem to produce weighted link paths that can be very precise when calculating semantic relatedness. So, we may be able learn weighting configurations with the help of human navigational paths in order to automatically derive paths based on such weightings that may be even better than the human navigational paths themselves.

Our results are not limited by our evaluation approach as a) WordSimilarity-353 is an established gold standard that is frequently used to evaluate methods for computing semantic relatedness and b) our experiments with alternative gold standards for semantic relatedness have produced results exhibiting similar trends (cf. Section 5.3). However, we want to extend our evaluation approach in future by showing the usefulness of our method of computing semantic relatedness by using the output for several NLP tasks like word sense disambiguation, recommendation or text segmentation. Furthermore, we want to establish automatic disambiguation processes for our pipeline.

The findings of this work have interesting implications for future research: i) While our results focus on semantic relatedness, it appears plausible that other semantic tasks, such as hypo/hypernym detection can benefit from data about human navigational paths as well. For example, West & Leskovec (2012) have found that navigation in semi-controlled settings tends to consist of two phases where in an initial exploration phase more abstract concepts are sought out, while in a subsequent exploitation phase more specific semantic concepts are selected. This could be used in future methods to compute different levels of abstractedness for concepts based on their position in navigational paths. ii) While we have studied the usefulness of human paths in a semi-controlled navigation scenario, a natural next step would be to study less controlled navigational scenarios - such as actual human navigation paths - and their usefulness for computing semantic relatedness. None of our measures for modeling navigational paths is constrained to semi-controlled navigation scenarios, and they can all be applied to less controlled scenarios as well. iii) Our work makes a compelling argument for expanding the existing arsenal of data sources for calculating semantic relatedness. It suggests that in addition to data from textual or structural (link) sources, *usage* data - such as human navigational paths - could play a pivotal role in the future. Hence, we can envision that future methods for computing semantic relatedness might not produce objective scores for semantic relatedness, but *subjective* scores that take into account how concepts are used and perceived by large user populations via analyzing their aggregate navigation behavior.

6.1. Acknowledgements

We want to thank Alex Clemesha (TheWikiGame) for access to the TheWikiGame dataset and Denis Helic (TU Graz) for his cooperation. This work is in part funded by the FWF Austrian Science Fund Grant I677 as well as by the DFG through the PoSTS project.

References

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., & Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* NAACL '09 (pp. 19–27). Stroudsburg, PA, USA: Association for Computational Linguistics.

Aguilar, C. M., & Medin, D. L. (1999). Asymmetries of comparison. *Psychon. Bull. Rev.*, *6*, 328–337.

Banerjee, S., & Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th international joint conference on Artificial intelligence* IJCAI'03 (pp. 805–810). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Budanitsky, A., & Hirst, G. (2001). Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. *Workshop on WordNet and Other Lexical Resources, Second meeting of the North American Chapter of the Association for Computational Linguistics, Pittsburgh, USA*, .

Cattuto, C., Benz, D., Hotho, A., & Stumme, G. (2008). Semantic Grounding of Tag Relatedness in Social Bookmarking Systems. In A. P. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. W. Finin, & K. Thirunarayan (Eds.), *Proceedings of the 7th International Conference on The Semantic Web* (pp. 615–631). Berlin, Heidelberg: Springer-Verlag volume 5318 of *ISWC '08*.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, *41*, 391–407.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2002). Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, *20*, 116–131.

Gabrilovich, E., & Markovitch, S. (2007). Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of The Twentieth International Joint Conference for Artificial Intelligence* IJCAI '07 (pp. 1606–1611). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Hassan, S., & Mihalcea, R. (2011). Semantic Relatedness Using Salient Semantic Analysis. In W. Burgard, & D. Roth (Eds.), *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence* AAAI '11. AAAI Press.

Helic, D., Strohmaier, M., Trattner, C., Muhr, M., & Lerman, K. (2011). Pragmatic Evaluation of Folksonomies. In *Proceedings of the 20th International Conference on World wide web* WWW '11 (pp. 417–426). New York, NY, USA: ACM.

Hovy, E., Navigli, R., & Ponzetto, S. P. (2012). Collaboratively built semi-structured content and artificial intelligence: The story so far. *Artificial Intelligence*, .

Ito, M., Nakayama, K., Hara, T., & Nishio, S. (2008). Association Thesaurus Construction Methods based on Link Co-occurrence Analysis for Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management* CIKM '08 (pp. 817–826). New York, NY, USA: ACM.

Jiang, J., & Conrath, D. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. of the Int'l. Conf. on Research in Computational Linguistics* (pp. 19–33).

Koerner, C., Benz, D., Strohmaier, M., Hotho, A., & Stumme, G. (2010). Stop Thinking, Start Tagging - Tag Semantics emerge from Collaborative Verbosity. In *Proceedings of the 19th International World Wide Web Conference* WWW '10. Raleigh, NC, USA: ACM.

Kozima, H. (1993). *Computing Lexical Cohesion as a Tool for Text Analysis*. Technical Report.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An Introduction to Latent Semantic Analysis. *Discourse Processes*, *25*, 259–284.

Li, Y., Bandar, Z. A., & McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Trans. on Knowl. and Data Eng.*, *15*, 871–882.

Manabu, O., & Takeo, H. (1994). Word sense disambiguation and text segmentation based on lexical cohesion. In *Proceedings of the 15th conference on Computational linguistics - Volume 2* COLING '94 (pp. 755–761). Stroudsburg, PA, USA: Association for Computational Linguistics.

Manning, C. D., Raghavan, P., & Schuetze, H. (2008). *Introduction to Information Retrieval*. (1st ed.). New York, NY, USA: Cambridge University Press.

Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., & Stumme, G. (2009). Evaluating Similarity Measures for Emergent Semantics of Social Tagging. In *Proceedings of the 18th International Conference on World Wide Web* WWW '09 (pp. 641–650). New York, NY, USA: ACM.

Medin, D. L., Goldstone, R. L., & Gentner, D. (1993). Respects for similarity. *PSYCHOLOGICAL REVIEW*, *100*, 254–278.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Commun. ACM*, *38*, 39–41.

Miller, G. A., & Charles, W. G. (1991). Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes*, *6*, 1–28.

Milne, D. (2008). Computing Semantic Relatedness using Wikipedia Link Structure. In *Proceedings of the New Zealand Computer Science Research Student Conference* NZCSRSC '07.

Milne, D., & Witten, I. H. (2008). An Effective, Low-cost Measure of Semantic Relatedness Obtained from Wikipedia Links. In *In Proceedings of the Conference on Artificial Intelligence* AAAI '08.

Nakayama, K., Hara, T., & Nishio, S. (2008). Wikipedia Link Structure and Text Mining for Semantic Relation Extraction Towards a Huge Scale Global Web Ontology.

Navigli, R., & Lapata, M. (2010). An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE Trans. Pattern Anal. Mach. Intell.*, *32*, 678–692.

Navigli, R., & Ponzetto, S. P. (2012a). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, *193*, 217 – 250.

Navigli, R., & Ponzetto, S. P. (2012b). Babelrelate! a joint multilingual approach to computing semantic relatedness. In *AAAI Conference on Artificial Intelligence*.

Patwardhan, S. (2006). Using wordnet-based context vectors to estimate the semantic relatedness of concepts. In *In: Proceedings of the EACL* (pp. 1–8).

Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). Wordnet::similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004* HLT-NAACL–Demonstrations '04 (pp. 38–41). Stroudsburg, PA, USA: Association for Computational Linguistics.

Ponzetto, S. P., & Strube, M. (2007a). Deriving a Large Scale Taxonomy from Wikipedia. In *Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 2* (pp. 1440–1445). AAAI Press volume 2 of *AAAI '07*.

Ponzetto, S. P., & Strube, M. (2007b). Knowledge derived from wikipedia for computing semantic relatedness. *J. Artif. Int. Res.*, *30*, 181–212.

Rada, R., Mili, H., Bicknell, E., & Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, *19*, 17–30.

Resnik, P. (1998). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, *11*, 95–130.

Rubenstein, H., & Goodenough, J. B. (1965). Contextual Correlates of Synonymy. *Communications of the ACM*, *8*, 627–633.

Salton, G. (1989). *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

Salton, G., & Buckley, C. (1988). Term-weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, *24*, 513–523.

Schuetze, H., & Pedersen, J. O. (1997). A Cooccurrence-based Thesaurus and two Applications to Information Retrieval. *Information Processing and Management*, *33*, 307–318.

Singer, P., Niebler, T., Strohmaier, M., & Hotho, A. (2013). Computing semantic relatedness from human navigational paths on wikipedia. In *Proceedings of the 22nd international conference on World Wide Web companion* WWW '13 Companion (pp. 171–172). Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, *22*, 1349–1380.

Srihari, R. K., Zhang, Z., & Rao, A. (2000). Intelligent indexing and semantic retrieval of multimodal documents. *Inf. Retr.*, *2*, 245–275.

Steiger, J. H. (1980). Tests for comparing elements of a correlation matrix. *Psychological Bulletin*, *87*, 245.

Strohmaier, M., Helic, D., Benz, D., Koerner, C., & Kern, R. (2012). Evaluation of Folksonomy Induction Algorithms. *ACM Transactions on Intelligent Systems and Technology*, *3*, 74:1–74:22.

Strube, M., & Ponzetto, S. P. (2006). WikiRelate! Computing Semantic Relatedness using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence* (pp. 1419–1424). AAAI Press volume 2 of *AAAI '06*.

Talmi, D., & Moscovitch, M. (2004). Can semantic relatedness explain the enhancement of memory for emotional words? *Mem Cognit*, *32*, 742–51.

Trattner, C., Singer, P., Helic, D., & Strohmaier, M. (2012). Exploring the Differences and Similarities between Hierarchical Decentralized Search and Human Navigation in Information Networks. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies* i-KNOW '12 (pp. 14:1–14:8). New York, NY, USA: ACM.

Turdakov, D., & Velikhov, P. (2008). Semantic Relatedness Metric for Wikipedia Concepts based on Link Analysis and its Application to Word Sense Disambiguation. In S. Kuznetsov, P. Pleshachkov, B. Novikov, & D. Shaporenkov

(Eds.), *Proceedings of the SYRCODIS 2008 Colloquium on Databases and Information Systems Saint-Petersburg, Russia, May 29-30, 2008*. CEUR-WS.org volume 355 of *CEUR Workshop Proceedings*.

Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*, 327–352.

West, R., & Leskovec, J. (2012). Human Wayfinding in Information Networks. In *Proceedings of the 21st International Conference on World Wide Web* WWW '12 (pp. 619–628). New York, NY, USA: ACM.

West, R., Pineau, J., & Precup, D. (2009). Wikispeedia: An Online Game for Inferring Semantic Distances between Concepts. In *Proceedings of the 21st International Joint Conference on Artifical Intelligence* IJCAI '09 (pp. 1598–1603). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Yang, D., & Powers, D. M. W. (2005). Measuring semantic similarity in the taxonomy of wordnet. In *Proceedings of the Twenty-eighth Australasian conference on Computer Science - Volume 38* ACSC '05 (pp. 315–322). Darlinghurst, Australia, Australia: Australian Computer Society, Inc.

Yazdani, M., & Popescu-Belis, A. (2013). Computing text semantic relatedness using the contents and links of a hypertext encyclopedia. *Artif. Intell.*, *194*, 176–202.

Yeh, E., Ramage, D., Manning, C. D., Agirre, E., & Soroa, A. (2009). WikiWalk: Random Walks on Wikipedia for Semantic Relatedness. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing* TextGraphs-4 (pp. 41–49). Stroudsburg, PA, USA: Association for Computational Linguistics.

Zhang, Z., Gentile, A., & Ciravegna, F. (2012). Recent advances in methods of lexical semantic relatedness-a survey. *Natural Language Engineering*, *1*, 1–69.

3.6. Hyptrails: A Bayesian Approach for Comparing Hypotheses about Human Trails

In the final article presented in this cumulative thesis, I aim at answering the third research question which is motivated by the findings of the first four articles tackling the first two research question. In detail, this thesis has identified several patterns and strategies that seem to guide human trails on the Web. This is accompanied by various findings of a series of related works regarding behavioral aspects of human trails. However, it is difficult to gauge these hypotheses' relative plausibility within a coherent research approach which is the task of the third research question tackled by this article.

Following this, colleagues and I have presented an approach called HypTrails in this article. HypTrails allows researchers to express and compare hypotheses about human trails on the Web. We understand hypotheses as beliefs in transitions between states. For example, based on the observations in the previous sections, we might have a strong belief (hypothesis) that humans prefer to navigate the Web by choosing semantically related concepts on Wikipedia while navigating. With this approach, it is possible to compare this and similar hypotheses with each other given empirical human trail data.

Technically, HypTrails models human trails with a first-order Markov chain model utilizing Bayesian inference. The main idea is to incorporate hypotheses about human trails on the Web as informative Dirichlet priors into the inference process. Marginal likelihoods and Bayes factors are then leveraging for comparing the plausibility of hypotheses with each other. By doing so, HypTrails makes use of the sensitivity of the Bayes factor on the prior as thoroughly discussed in Section 2.1.2. By presenting a novel adaption of the so-called (trial) roulette method, the approach allows researchers to intuitively express their hypotheses about human trails on the Web as matrices with elements capturing beliefs in transitions between states. These matrices are then used by our method for eliciting proper Dirichlet priors.

We have demonstrate the general mechanics and applicability of HypTrails by performing experiments with both synthetic as well as empirical human trail data as introduced in Section 1.2. The synthetic trails have been produced according to known mechanics which we control. The empirical trails stem from three different domains: (i) human navigational trails on Wikipedia, (ii) business reviews on Yelp and (iii) songs listened to on Last.fm. Overall, this article answers the final research question, facilitates future studies about the production of human trails, and expands the repertoire of methods for studying human trails on the Web.

HypTrails: A Bayesian Approach for Comparing Hypotheses about Human Trails

Philipp Singer
GESIS & Graz University of Technology
philipp.singer@gesis.org

Denis Helic
Graz University of Technology
dhelic@tugraz.at

Andreas Hotho
University of Würzburg
hotho@informatik.uni-wuerzburg.de

Markus Strohmaier
GESIS & University of Koblenz-Landau
markus.strohmaier@gesis.org

ABSTRACT

When users interact with the Web today, they leave sequential digital trails on a massive scale. Examples of such human trails include Web navigation, sequences of online restaurant reviews, or online music play lists. Understanding the factors that drive the production of these trails can be useful for e.g., improving underlying network structures, predicting user clicks or enhancing recommendations. In this work, we present a general approach called HypTrails for comparing a set of hypotheses about human trails on the Web, where hypotheses represent beliefs about transitions between states. Our approach utilizes Markov chain models with Bayesian inference. The main idea is to incorporate hypotheses as informative Dirichlet priors and to leverage the sensitivity of Bayes factors on the prior for comparing hypotheses with each other. For eliciting Dirichlet priors from hypotheses, we present an adaption of the so-called (trial) roulette method. We demonstrate the general mechanics and applicability of HypTrails by performing experiments with (i) synthetic trails for which we control the mechanisms that have produced them and (ii) empirical trails stemming from different domains including website navigation, business reviews and online music played. Our work expands the repertoire of methods available for studying human trails on the Web.

Categories and Subject Descriptors: H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*Web-based interaction*

Keywords: Human Trails, Markov Chain, Hypotheses

1. INTRODUCTION

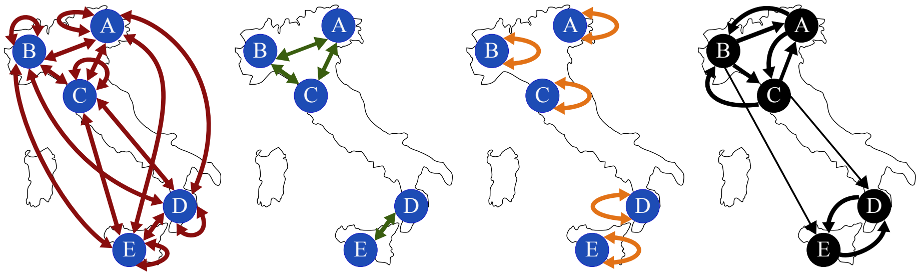
The idea of human trails in information systems can be traced back to early work by Vannevar Bush ("As We May Think" [10]), in which he described a hypothetical system called *Memex*. Bush hypothesized that human memory operates by association, with thoughts defined by internal connections between concepts. The Memex itself was intended as users' extension of their memory, where common associative trails between documents can be stored, accessed and shared. Eventually, Bush's ideas led to the concept of Hypertext [28] and the development of the World Wide Web [4].

Today, the Web facilitates the production of human trails on a massive scale; examples include successive clicks on hyperlinks when users navigate the Web, successive songs played in online music services or sequences of restaurant reviews when sharing experiences on the Web. Understanding such human trails and how they are produced has been an open and complex challenge for our community for years. A large body of previous work has tackled this challenge from various perspectives, including (i) modeling [7, 8, 13, 32, 36, 37], (ii) regularities and patterns [20, 44, 45] and (iii) cognitive strategies, finding that, for example, humans prefer to consecutively choose semantically related nodes [38, 47], humans participate in partisan sharing [2] or users benefit from following search trails [49]. In this paper, we are interested in tackling an important sub-problem of this larger challenge.

Problem. In particular, we take a look at the problem of expressing and comparing different *hypotheses* about *human trails* given empirical observations. We define trails as *a sequence of at least two successive states*, and hypotheses as *beliefs about transitions between states*. An intuitive way of expressing such hypotheses is in the form of Markov transitions and our beliefs in them. For example, we might have various hypotheses about how humans consecutively review restaurants on Yelp. Figure 1 (a-c) shows three exemplary hypotheses for transitions between five restaurants (A-E) in Italy, and actual empirical transitions (d). The *uniform hypothesis* in Fig 1(a) expresses a belief that all transitions are equally likely (a complete digraph). In Fig 1(b), the *geo hypothesis* expresses a belief that humans prefer to consecutively review geographically close restaurants, while the *self-loop hypothesis* in Fig 1(c) expresses the belief that humans repeatedly review the same restaurant without ever reviewing another one. Other hypotheses are easily conceivable. What is difficult today is *expressing and comparing such hypotheses within a coherent research approach*. Such an approach would allow to make relative statements about the plausibility of different hypotheses given empirical data about human trails.

Objectives. We thus tackle the problem of *comparing a set of hypotheses about human trails given data.* We present a Bayesian approach – which we call *HypTrails*[1] – that provides a general solution to this problem.

[1] Portmanteau for Hyp(ertext/otheses) Trails



(a) *Uniform hypothesis:* all transitions are equally likely

(b) *Geo hypothesis:* regional node transitions are most likely

(c) *Self-loop hypothesis:* self-transitions are most likely

(d) *Empirical transitions:* obtained from real world data

Figure 1: *Example.* This figure illustrates three exemplary hypotheses (a), (b), and (c) about human trails as well as empirical observations obtained from real-world data (d). We look at trails of online restaurant reviews in Italy; nodes A-E represent restaurants. Hypotheses (a) – (c) are expressed via edges, with edge weights indicating strength of belief. For empirical data (d), edge weights correspond to actually observed transitions (how many times a restaurant has been reviewed before another restaurant). Our proposed approach compares evidences for different hypotheses given observed data (d). In this example, the geographic hypothesis (b) would be the most plausible one as we can mostly observe regional transitions between restaurants in the data (d).

Approach & Methods. The HypTrails approach utilizes a *Markov chain model* for modeling human trails and *Bayesian inference* for comparing hypotheses. The main idea is to (i) let researchers express hypotheses about human trails as adjacency matrices which are then used for (ii) eliciting informative Dirichlet priors using an adapted version of the (trial) roulette method. Finally, the approach (iii) leverages the sensitivity of Bayes factors on the priors for comparing hypotheses with each other. We experimentally illustrate our approach by studying synthetic datasets with known mechanisms which we then express as hypotheses. We demonstrate the general applicability of HypTrails by comparing hypotheses for empirical datasets from three distinct domains (Wikigame, Yelp, Last.fm).

Contributions. Our main contribution is the presentation of HypTrails, a general approach for expressing and comparing hypotheses about human trails. While the basic building blocks of HypTrails are well established (Markov chains, Bayesian inference), we combine, adapt and extend them in an innovative way that facilitates intuitive expression and elegant comparison of hypotheses. In particular, our adaption of the (trial) roulette method represents a simple way of eliciting priors for Markov chain modeling. We demonstrate the applicability of our framework in a series of experiments with synthetic and real-world data. Finally, to facilitate reproducibility and future experimentation, we make an implementation of HypTrails openly available to the community[2].

Structure. We present our approach in Section 2. Section 3 describes the synthetic and empirical data analyzed; corresponding experiments are presented in Section 4. We discuss our work in Section 5, present related work in Section 6 and conclude in Section 7.

2. THE HYPTRAILS APPROACH

We start with defining the problem setting and giving a short overview of the proposed approach in Section 2.1. We proceed with explaining the fundaments of our approach based on Bayesian Markov chain modeling in Section 2.2 where we also emphasize our main idea of incorporating hypotheses as Dirichlet priors and leveraging the sensitivity of Bayes factors for comparing hypotheses with each other. In Section 2.3 we thoroughly discuss the process of eliciting Dirichlet priors from scientific hypotheses.

2.1 Problem Definition & Approach

We aim to produce a partial ordering O over a set of hypotheses $\mathbf{H} = \{H_1, H_2, ..., H_n\}$. We base the partial order on the plausibility of hypotheses given data D. Each hypothesis H describes beliefs about common transitions between nodes while data D captures empirically observed human trails. A hypothesis H can be expressed by an adjacency matrix Q where transitions $q_{i,j}$ with strong belief receive larger values than those with lower belief.

For generating the partial ordering O, our HypTrails approach resorts to Bayesian inference utilizing a Markov chain model. We incorporate a hypothesis H as informative Dirichlet priors into the inference process. For eliciting Dirichlet priors $Dir(\alpha)$ from a given hypothesis H expressed as matrix Q – i.e., for setting corresponding hyperparameters $\alpha_{i,j}$ – HypTrails uses an adaption of the so-called (trial) roulette method. The partial ordering O is achieved by calculating marginal likelihoods $P(D|H)$ (weighted averages of likelihood, where the weights come from the parameters' prior probabilities) for competing hypotheses H which we then can compare with each other by determining Bayes factors B.

2.2 Bayesian Markov Chain Modeling

HypTrails is based on Bayesian Markov chain modeling. In the following, we only cover those fundamentals that are directly related to our approach, and point the reader to previous work [37, 40] for a more detailed treatise of the topic.

Markov chain definition. A Markov chain model represents a stochastic system that models transitions between states from a given state space $S = \{s_1, s_2, ..., s_m\}$ with $m = |S|$ (e.g., the distinct restaurants of our example in Figure 1). It amounts to a sequence of random variables $X_1, X_2, ..., X_t$. This random process is usually memoryless (the so-called Markov property, first-order) meaning that the next state only depends on the current state and not on a sequence of preceding states. Note though that Markov chain models can also be extended to incorporate higher orders; see Section 5 for a discussion. We can define the Markov property as:

$$\begin{aligned} P(X_{t+1} = s_j | X_1 = s_{i_1}, ..., X_{t-1} = s_{i_{t-1}}, X_t = s_{i_t}) = \\ P(X_{t+1} = s_j | X_t = s_{i_t}) = p_{i,j}. \end{aligned} \tag{1}$$

Markov chain models have been established as a robust method

[2] https://github.com/psinger/HypTrails

for modeling human trails on the Web in the past (e.g., [14, 37, 45]), specifically focusing on human navigational trails (e.g., [6, 32]) with Google's PageRank being the most prominent example [8]. Hence, the Markov chain model is a natural and intuitive choice for our approach as it lets us explicitly model human trails with a dependence of the next state on the current state. We also consider hypotheses as beliefs about transitions without memory.

A Markov model is usually represented by a stochastic transition matrix P with elements $p_{i,j} = P(s_j|s_i)$ which describe the probability of transitioning from state s_i to state s_j; the probabilities of each row sum to 1. The elements of this matrix are the parameters θ that we want to determine. For doing so we resort to Bayesian inference.

Bayesian inference. Bayesian inference refers to the Bayesian process of inferring the unknown parameters θ from data; it treats data and model parameters as random variables. For a more detailed discussion of Bayesian inference please refer to [37, 40]. Following Bayes' rule, the posterior distribution of parameters θ given data D and hypothesis H is then defined as:

$$\overbrace{P(\theta|D,H)}^{\text{posterior}} = \frac{\overbrace{P(D|\theta,H)}^{\text{likelihood}}\overbrace{P(\theta|H)}^{\text{prior}}}{\underbrace{P(D|H)}_{\text{marginal likelihood}}} \tag{2}$$

The *likelihood function* describes the likelihood that we observe data D with given parameters θ and hypothesis (model) H. The *prior* reflects our belief about the parameters before we see the data or – more technically – the prior *encodes* our hypothesis H. Thus, *we use the prior as a representation for different hypotheses about human trails*. More precisely, we model the data with different – mostly informative – priors. We use the conjugate prior of the categorical distribution as the prior of each row of the transition matrix P; i.e., the Dirichlet distribution $Dir(\boldsymbol{\alpha})$. The hyperparameters $\boldsymbol{\alpha}$ represent our prior belief of the parameters and can be seen as a vector of pseudo counts $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_m]$. Given such a prior, the posterior distribution represents a combination of our prior belief and the actual data that we observe. For each row i of P we now have a posterior in the form of $Dir(n_{i,1} + \alpha_{i,1}, ..., n_{i,m} + \alpha_{i,m})$ where $n_{i,j}$ are the actual transition counts of the data between states s_i and s_j and $\alpha_{i,j}$ are the prior pseudo counts assigned to this transition. We provide a thorough description of how we elicit the needed Dirichlet priors from expressed hypotheses by researchers in Section 2.3.

Comparing hypotheses. Finally, the *marginal likelihood* (which we can also call evidence) expresses the probability of the data given a hypothesis H and plays a crucial role for comparing hypotheses;[3] it is defined as follows (for derivation please consult [37, 40]):

$$P(D|H) = \prod_i \frac{\Gamma(\sum_j \alpha_{i,j})}{\prod_j \Gamma(\alpha_{i,j})} \frac{\prod_j \Gamma(n_{i,j} + \alpha_{i,j})}{\Gamma(\sum_j (n_{i,j} + \alpha_{i,j}))} \tag{3}$$

Note that the hyperparameters $\alpha_{i,j}$ differ for various hypotheses H as we express them via different Dirichlet priors; the actual transition counts $n_{i,j}$ are the same for each hypothesis. For comparing the plausibility of two hypotheses, we resort to *Bayes factors* [21, 46]. Bayes factors are representing a Bayesian method for model comparison that include a natural *Occam's razor* guarding against overfitting. In our case, a model represents a hypothesis at interest with each having different priors with different hyperparameters that express corresponding beliefs. For illustrative purposes, we are now interested in comparing hypotheses H_1 and H_2 where $H_1, H_2 \in \mathbf{H}$, given observed data D. We can define the Bayes factor – note that we apply unbiased comparison assuming that all hypotheses are equally likely a priori – as follows:

$$B_{1,2} = \frac{P(D|H_1)}{P(D|H_2)} \tag{4}$$

$P(D|H)$ is the marginal likelihood (evidence) defined in Equation 3 and the Bayes factor can be seen as a summary of the evidence provided by the data in favor of one scientific hypothesis over the other. HypTrails is not only suited for comparing two hypotheses with each other, but rather a set of hypotheses $\mathbf{H} = \{H_1, H_2, ..., H_n\}$. For determining the partial order O over $\mathbf{H}$, we order the evidences that data D provides in favor of hypotheses H; i.e., by ordering $P(D|H)$ using a less-than-equal binary relation. However, ordering the evidences is not enough as we need to check the significance of their ratios which we tackle by calculating Bayes factors. In case that the significance is not present, we consider two hypotheses as being equal. For determining the strength of the Bayes factor we resort to Kass and Raftery's [21] interpretation table .

Leveraging the sensitivity of Bayes factors. Throughout this section we have described our main idea of incorporating hypotheses in the form of informative Dirichlet priors into the inference process. We leverage marginal likelihoods and Bayes factors for making informed decisions about the relative plausibility of given hypotheses. Usually, one common critique of Bayes factors is that they are highly sensitive with regard to the choice of the prior [21]. In contrast, posterior measures ignore the influence of the prior the more data one observes and incorporates in the model which is why they are more ignorant to the choice of prior and even encourage vaguely specified priors [41]. In our approach, we exploit this property of Bayes factors as *an elegant solution to the problem of comparing hypotheses*. As we express our different hypotheses in the form of priors, we are explicitly interested in using a measure that is sensitive to the choice of priors and hence, can give us insights into the relative plausibility of each hypothesis. Thus, marginal likelihoods and Bayes factors are an appropriate measure for comparing scientific hypotheses as pointed out by Wolf Vanpaemel [41].

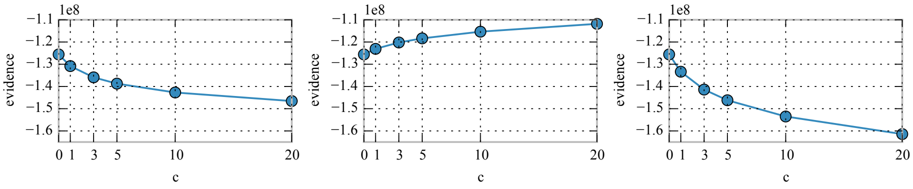
2.3 Eliciting Dirichlet Priors

This section explains in greater detail how we can express the hypotheses about human trails and how HypTrails elicits proper informative Dirichlet priors from these. First, we illustrate how the prior influences the evidence by studying several toy examples. Next, we present an intuitive way of eliciting the Dirichlet priors by introducing an adaption of the so-called *(trial) roulette method*.

Understanding influence of priors. In Section 2.2, we discussed that we use the sensitivity of Bayes factors with regard to the choice of prior (i.e., determined via marginal likelihoods) as a feature (or solution) rather than a limitation. It allows us to model hypotheses in the form of prior distributions which then can be compared by corresponding Bayes factors. But how exactly does the prior influence evidence (marginal likelihood)? Note that the posterior probability (see Equation 2) is a combination of our prior belief (pseudo counts) and the data we observe (transition counts) which is why both influence the evidence (see Equation 3).

To illustrate the influence, we apply several toy priors to human trail data – the choice of data is secondary and we observe the same behavior regardless of the underlying data; in this case we exemplary use navigation data (Wikigame dataset introduced in Section 3). First, we apply a *uniform prior*; i.e., $\boldsymbol{\alpha}$ has the same value for each row i and element j: $\alpha_{i,j} = 1 + c, \forall i, j$. By ranging the constant c over $0, 1, 3, 5, 10, 20$, we observe decreasing evidence (Figure 2(a))

[3] Note that we calculate log-evidence utilizing logarithms of the gamma function for avoiding underflow.



(a) Uniform prior

(b) Empirically aligned prior

(c) Empirically opposing prior

Figure 2: *Understanding influence of priors.* This figure shows how the choice of prior pseudo counts influences evidence; we apply several toy priors to navigation data (Wikigame, see Section 3). In (a) we use a uniform Dirichlet prior which means that for each row i, α has the same value for each element: $\alpha_{i,j} = 1 + c, \forall i, j$. By increasing the constant c (x-axis), we can observe that the evidence (y-axis) is decreasing; the largest evidence is at $c = 0$. By using an empirically aligned prior (b) as $(n_{i,j} > 0 \rightarrow \alpha_{i,j} = 1 + c) \wedge (n_{i,j} = 0 \rightarrow \alpha_{i,j} = 1), \forall i, j$, we end up with a larger evidence the larger c is. Finally, in (c) we intentionally set "bad" prior counts for the α values via $(n_{i,j} = 0 \rightarrow \alpha_{i,j} = 1 + c) \wedge (n_{i,j} > 0 \rightarrow \alpha_{i,j} = 1), \forall i, j$ showing that the evidence becomes smaller as we increase c. The results indicate that the more a hypothesis is aligned with empirical data, the larger the evidence is - and vice versa.

which is not surprising as the uniform pseudo counts do not mirror the observed transition counts well. Technically, with increased c the Dirichlet prior concentrates more and more of its probability mass on a uniform distribution of parameters, and thus the weights of alternative parameter configurations become smaller. However, the likelihood is larger for the alternative parameter configurations (coming from the data) and this results in a smaller weighted average of the likelihood, i.e., in a smaller evidence.

On the other hand, if we provide some form of an *empirically aligned prior* as $(n_{i,j} > 0 \rightarrow \alpha_{i,j} = 1 + c) \wedge (n_{i,j} = 0 \rightarrow \alpha_{i,j} = 1), \forall i, j$ we end up with a larger evidence the larger c is as we can see in Figure 2(b). This is because we actually increase the pseudo counts of transitions that we also observe in our data while we keep the pseudo counts for non-observed transitions at 1. In this case, we concentrate the prior probability mass on the parameter configuration that is very well aligned with the observations. As a consequence we give more weight for parameter configurations where the likelihood is anyhow large and this increases the evidence.

Finally, we illustrate the behavior of a toy prior that expressed an *empirically opposing prior* in the form of $(n_{i,j} = 0 \rightarrow \alpha_{i,j} = 1 + c) \wedge (n_{i,j} > 0 \rightarrow \alpha_{i,j} = 1), \forall i, j$. In this example we intentionally set the prior pseudo counts to the opposite of what the actual data tells us. We assign low prior pseudo counts (1) to elements with large observed transition counts while we incrementally increase the pseudo counts of transitions that we do not observe in our data. As expected, Figure 2(c) shows that the evidence decays as we increase c. Technically, we assign the greatest weights for parameter configurations with the smallest likelihoods resulting in a steep decay of evidence with increasing values of c.

Note that $c = 0$ results in the same evidence for all three toy priors as in all cases $\alpha_{i,j} = 1, \forall i, j$ (uniform prior). These toy examples demonstrate that if the prior is well aligned with data, then the evidence is rising with the strength of the prior. The marginal likelihood is the largest if the prior and the likelihood are concentrated over the same parameter regions and the evidence is lowest if they concentrate on different regions [50]. Hence, we want to choose an informative prior that captures the same regions as the likelihood. This leads to the observation that if our prior choice represents a valid hypothesis about behavior producing human trails on the Web, the evidence should be larger than a uniform prior, or an unlikely hypothesis prior with an equal amount of pseudo counts. We always want to compare hypotheses with each other that exhibit the same amount of pseudo counts assigned.

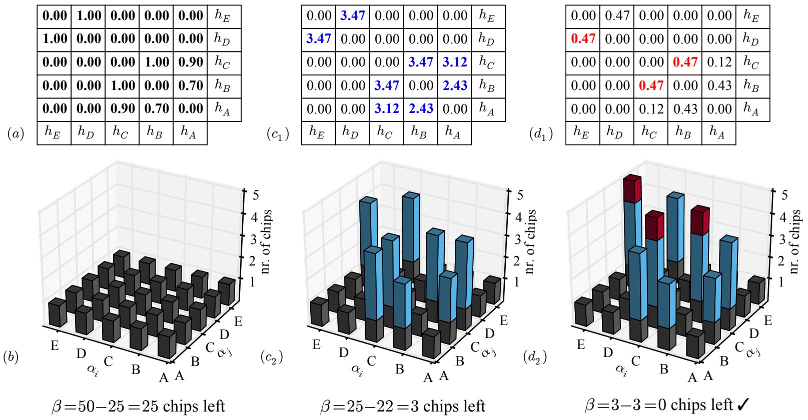
(Trial) roulette method. Our approach requires to define the parameters of prior Dirichlet distributions by setting the pseudo counts (hyperparameters) $\alpha_{i,j}$ given the hypothesis at interest. However, the process of eliciting prior knowledge is no trivial problem and requires careful steps (see [18, 29] for a discussion). As a solution, we present an adaption of the so-called *(trial) roulette method* which was originally proposed in [19] and further discussed in [16, 29]. It is a graphical method that allows experts to express their subjective belief by distributing a fixed set of chips (think about casino chips you set on a roulette table) to a given grid (e.g., bins representing result intervals). The number of chips assigned to an element of the grid then reflect the experts' belief in the specific bin.

In our work we adapt the (trial) roulette method. The grid can be understood as a matrix Q where each element $q_{i,j}$ of the grid represents the belief of a given hypothesis about the transition from state s_i to state s_j. Values $q_{i,j}$ are set by researchers for expressing a hypothesis. They need to be positive values and larger values indicate stronger belief in a given transition. The prior of each row of the transition matrix P of the Markov chain model is defined as a Dirichlet distribution (cf. Section 2.2) with parameters (pseudo counts) $[\alpha_{i,1}, \alpha_{i,2}, ..., \alpha_{i,j}]$ which we want to set given the hypothesis. Concretely, we want to automatically distribute a number of chips to the given pseudo counts according to the values provided in matrix Q expressing a hypothesis H. We define the overall number of chips to distribute for a given hypothesis as:

$$\beta = \overbrace{m^2}^{\text{uniform prior}} + \underbrace{k \cdot m^2}_{\text{additional informative prior}} \tag{5}$$

$m = |S|$ and m^2 amounts to the uniform prior – i.e., we assign the same number of pseudo counts (1) to each transition – which is why the number of uniformly assigned chips is equal to the overall number of parameters of the Markov chain model. Additionally, we distribute $k * m^2$ informative pseudo clicks for the given hypothesis, where k describes a weighting factor for the informative part. The larger we set k, the more we concentrate the Dirichlet distributions according to a hypothesis at interest – see Section 5 for a discussion.

By and large, the goal of our adaption of the (trial) roulette method is to not only give researchers an intuitive way of expressing their hypotheses as matrices Q, but also to elicit informative Dirichlet distributions according to the values $q_{i,j}$ of Q. Next, we want to illustrate the process of expressing a hypothesis and assigning prior pseudo counts via the example shown in Figure 3 using the (trial) roulette method. Let us again focus on human trails over reviewed



(a)

0.00	**1.00**	**0.00**	**0.00**	**0.00**	h_E
1.00	**0.00**	**0.00**	**0.00**	**0.00**	h_D
0.00	**0.00**	**0.00**	**1.00**	**0.90**	h_C
0.00	**0.00**	**1.00**	**0.00**	**0.70**	h_B
0.00	**0.00**	**0.90**	**0.70**	**0.00**	h_A
h_E	h_D	h_C	h_B	h_A	

(c_1)

0.00	**3.47**	0.00	0.00	0.00	h_E
3.47	0.00	0.00	0.00	0.00	h_D
0.00	0.00	0.00	**3.47**	**3.12**	h_C
0.00	0.00	**3.47**	0.00	**2.43**	h_B
0.00	0.00	**3.12**	**2.43**	0.00	h_A
h_E	h_D	h_C	h_B	h_A	

(d_1)

0.00	0.47	0.00	0.00	0.00	h_E
0.47	0.00	0.00	0.00	0.00	h_D
0.00	0.00	0.00	**0.47**	0.12	h_C
0.00	0.00	**0.47**	0.00	0.43	h_B
0.00	0.00	0.12	0.43	0.00	h_A
h_E	h_D	h_C	h_B	h_A	

β=50−25=25 chips left

β=25−22=3 chips left

β=3−3=0 chips left ✓

Figure 3: *Illustration of the (trial) roulette method.* In this figure the most important steps of our trial roulette method are visualized. We begin with a matrix expressing a researcher's hypothesis about human trails in (a) – in this case the exemplary geographic hypothesis for trails over businesses reviewed (cf. Figure 1). The (trial) roulette method proceeds with distributing a given number of chips (pseudo counts, in this case $\beta = 50$) to the Dirichlet priors. It starts by assigning one chip to each element (uniform) as can be seen in (b) before it proceeds by assigning the remaining chips according to their values of our given matrix in (a) as can be seen from (c) to (d). In each column, values of the matrices that receive at least one chip are marked bold and in the same color as the bars indicating chip assignments for the Dirichlet priors. In case of ties the ranking is produced in random fashion. For details please see Section 2.3.

restaurants in Italy (see Figure 1) and illustrate the method using the geographic hypothesis given in Figure 1(b). For this visualization, we assume that we set $k = 1$ leading to $\beta = m^2 + m^2$ chips we want to distribute. The following steps are necessary:

(a) Expressing the hypothesis. Researchers start with expressing the hypothesis matrix Q with elements $q_{i,j}$ that capture the belief about transitions of underlying human trails; the matrix can be seen in Figure 3(a). In this example, we have five states (restaurants) – i.e., $S = A, B, C, D, E$ and $m = 5$ leading to $\beta = 50$ chips to distribute – and we express our geographic hypothesis about common transitions with values between 0 and 1. The precondition is that only positive values are used and larger values ($q_{i,j}$) always express stronger beliefs compared to smaller values. In this case, the closer a value is to 1 the closer two restaurants are geographically and the more we believe in corresponding transitions. As can be seen in Figure 1(b), both restaurant pairs B-C and E-D are the closest in geographical terms which is why we lay the strongest belief in these symmetric transitions – concretely, we set $q_{B,C} = 1.0$, $q_{C,B} = 1.0$, $q_{D,E} = 1.0$ and $q_{E,D} = 1.0$. The next closest restaurant pair is A-C which is why we also have strong beliefs in humans consecutively reviewing restaurant A after C and vice versa – we set $q_{A,C} = 0.9$ and $q_{C,A} = 0.9$. Finally, we set $q_{A,B} = 0.7$ and $q_{B,A} = 0.7$ as we also have some (lower) belief that humans consecutively review given restaurants. We set all other transitions to zero as we believe that given restaurants are too far away. One matrix Q represents one general hypothesis H about human trails – hence, in a more realistic scenario one would want to express several of such matrices (such as for the other hypotheses given in Figure 1). Also note that we hand-pick values for this example and in a more rigorous investigation one would potentially use an automatic method for determining them (as we do in Section 4). The next steps are automatically performed by HypTrails for eliciting Dirichlet priors from such matrices, more specifically by the adapted (trial) roulette method.

(b) Initial uniform distribution. The (trial) roulette method starts with assigning uniform chips to each transition which can be seen as obtaining Laplace's prior ($\alpha_{i,j} = 1$ for each i and j) and accounts to the uniform prior part of Equation 5. The updated prior (i.e., hyperparameters for the Dirichlet distributions) can be seen in Figure 3(b) with black bars; all elements where one chip is assigned to are marked bold and black in Figure 3(a). By subtracting the distributed number of chips from β we have $\beta = 50 - 25 = 25$ chips left for the informative part described next.

(c) Informative distribution. Matrix Q gets normalized and then multiplied by the number of chips left: $Q = \frac{Q}{||Q||_1} * \beta$ where $||Q||_1$ is the ℓ_1-norm and $\beta = 25$. The resulting matrix can be seen in Figure 3(c$_1$). The method assigns as many chips to elements of the prior as the integer floored values of Q specify. So e.g., $q_{A,B} = 2.43$ and $\lfloor q_{A,B} \rfloor = 2$ leading to $\alpha_{A,B} += 2$ whereas $\lfloor q_{B,D} \rfloor = 0$ which is why the pseudo count for this transition is not increased. Overall, the method distributes 22 more chips marked bold and blue in Figure 3(c$_1$) leading to $\beta = 25 - 22 = 3$ chips left; the updated prior distributions (new chips marked blue) can be seen in Figure 3(c$_2$).

(d) Remaining informative distribution. Finally, the method subtracts the integer floored values from Q leading to the matrix illustrated in Figure 3(d$_1$) calculated by $Q = Q - \lfloor Q \rfloor$. It now needs to distribute the chips left (three in this case) according to the remaining values in Q. The method accomplishes that by ranking the values in descending order and assigning one chip to each element until none is left, starting from the largest and ending at the smallest. In case of a tie the ranking for the ties is produced in random fashion – hence, in this case $\alpha_{D,E}$ does not receive one more chip. We mark the elements that receive one further chip bold and red in Figure 3(d$_1$) and update our prior pseudo counts as can be seen in

Figure 3(d_2) also in red color. Now, the (trial) roulette method has no chips left and is finished.

The final chip assignment as can be seen in Figure 3(d_2) now represent the prior (hypothesis). In detail, each row corresponds to a Dirichlet distribution with corresponding pseudo counts (hyperparameters) $\alpha_{i,j}$ – e.g., $\alpha_{C,B} = 5$. By proceeding, our HypTrails approach now uses these Dirichlet priors for Bayesian Markov chain modeling inference as described in Section 2.2. Concretely, in combination with the transitions $n_{i,j}$ observed from data they influence the marginal likelihood calculated as defined in Equation 3. See Figure 2 for a visualization of how the prior influences the evidence. By repeating the trial roulette method and evidence calculation

3. DESCRIPTION OF DATASETS

In this section, we introduce both synthetic as well as empirical datasets which we consider for our experiments. We produce *synthetic data* consisting of simulated human trails – in this case navigational trails – with known mechanisms from a generated (i.e., artificial) network. The introduced *empirical data* stems from three real-world datasets from different domains. The state space S investigated is always defined by the distinct elements the trails traverse over – e.g., if we observe trails over five distinct restaurants being reviewed (see Figure 1) we consider these five for the state space.

3.1 Synthetic Datasets

We start by generating a directed random network using a generalized version of *Price's preferential attachment scale-free network model* [3, 33]. The network generation algorithm starts with a clique containing 11 nodes and proceeds to add nodes with an out-degree of 10 leading to an overall network size of $10,000$ nodes. These parameters are arbitrary and could be set differently. Next, we simulate three different kinds of navigational trails, each consisting of exemplary $1,000$ trails of length 5, as follows:

Structural random walk. For each trail we start at a random node of the network and perform a random walk through the network. The walker chooses the next node by randomly selecting one out-going link of the current node.

Popularity random walk. Again, the walker starts at a random node of the network, but now selects the next node by choosing the out-link according to the target's in-degree. The walker lays a softmax-like smoothing over the in-degrees of all target nodes ($e^{\deg^-(s)10}$) and then chooses the next node according to given probability leading to a small stochastic effect. This is aimed at averting too long loops that would happen with simple greedy selection.

Random teleportation. Again, we start with a random node in the network for each trail. However, we now completely ignore the underlying topological link network and simply randomly choose any other node of the network – i.e., teleporting through the network.

3.2 Empirical Datasets

For our experiments we also consider three different empirical datasets which are described next.

Wikigame dataset. First, we study navigational trails over Wikipedia pages that are consecutively visited by humans. The dataset is based on the online game called Wikigame (`thewikigame.com`) where players aim to navigate to a given Wikipedia target page starting from a given Wikipedia start page using Wikipedia's link structure only. All start-target pairs are guaranteed to be connected in Wikipedia's topological link network and users are only allowed to click hyperlinks and use the browser's buttons such as refresh, but not use other features such as the search field. In this article we study trails collected from users playing the game between 2009-02-17 and 2011-09-12. Overall, the dataset consists of $1,799,015$ trails – where each trail represents the consecutive websites visited by one user for one game played – through Wikipedia's main namespace including $360,417$ distinct pages with an average trail length of around 6. We use corresponding textual and structural Wikipedia article data for hypotheses generation. In particular, we use the Wikipedia dump dated on 2011-10-07[4].

Yelp dataset. Second, we study human trails over successive businesses reviewed by users on the reviewing platform *Yelp* (`yelp.com`) – we have used this setting as an example throughout this article (e.g., see Figure 1). For generating these trails we use a dataset publicly offered by Yelp[5]. Overall, we generate $125,365$ trails – where each trail describes the subsequent review history of one single user – over $41,707$ distinct businesses with an average trail length of 8. The data also includes further information about the businesses like the geographic location or category markers assigned, which we will use for hypotheses generation.

Last.fm dataset. Third, we study human trails that capture consecutive songs listened to by users on the music streaming and recommendation website *Last.fm* (`lastfm.com`). We use a publicly available dataset for generating the trails at hand focusing on listening data stemming from one day (2009-01-01). Overall, the dataset consists of 275 trails – where each trail captures the successive songs listened to by one user on a given day – over $11,166$ distinct tracks with an average trail length of 52.8. For generating hypotheses, we consult the *MusicBrainz* (`musicbrainz.org`) API as describe later.

4. EXPERIMENTS

To demonstrate HypTrails and its general applicability, we perform experiments with both synthetic as well as empirical datasets (as introduced in Section 3).

4.1 Experiments with Synthetic Data

Our first experiments focus on applying HypTrails to three synthetic trail datasets, generated by the following mechanisms: teleportation, a random walk and a popularity random walk (see Section 3). In our experiments, we look at these three datasets and compare three corresponding hypotheses (uniform, structural, popularity) that capture the generative mechanisms of each dataset. As we know from theory, HypTrails ranks the hypothesis that best captures the underlying mechanisms as the most plausible one. Next, we introduce the hypotheses in greater detail, before we discuss the experimental results.

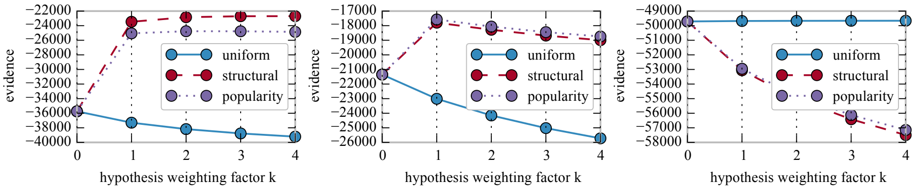
Hypotheses. We now describe how we express the three hypotheses as matrices Q. Note that we only have to specify the hypothesis matrix Q (see Section 2.3) while the concrete pseudo count distribution for generating proper priors is handled by our approach.

Uniform hypothesis. This hypothesis has the intuition that trails have been purely generated by random teleportation and all transitions are equally likely. Thus, we equally believe in each transition and set each element of Q to an equal value (here 1).

Structural hypothesis. This hypothesis captures our belief that the trails have been generated by (only) following the underlying topological link structure. Hence, we believe that agents would always choose a random link leading from one node to another while traversing the network. We express this by setting $q_{i,j}$ of Q to 1 if a directed link between state s_i and state s_j exists in the topological network.

[4] This Wikipedia dump closely resembles the information available to players of the game for our given time period.

[5] `yelp.com/dataset_challenge`



(a) Structural random walk

(b) Popularity random walk

(c) Random teleportation

Figure 4: *Experiments with synthetic data.* This figure depicts the results obtained when applying HypTrails to three synthetically generated trail corpora with known mechanisms (structural random walk (a), popularity random walk (b) and random teleportation (c)) comparing three different hypotheses: (i) uniform (solid, blue lines), (ii) structural (dashed, red lines) and (iii) popularity (dotted, purple lines). In each figure, the x-axis depicts the strength (weighting factor k) one assigns to a given hypothesis as defined in Equation 5 ($k = 0$ refers to a uniform prior) while the y-axis shows the corresponding evidence (marginal likelihood) value. For simplicity, we can compare hypotheses with each other by comparing the evidence values (larger values mean higher plausibility) for the same values of k as all Bayes factors are decisive. The results illustrate what we know from theory as for each dataset the hypothesis that captures the mechanisms according to which the data has been produced best, is declared as the most plausible one.

Popularity hypothesis. This hypothesis also believes that the trails have been generated by following the links of the underlying link structure, but we have stronger beliefs in choosing large in-degree nodes compared to low in-degree nodes. Hence, we set $q_{i,j}$ to $\deg^-(s_j)$ if a directed link between state s_i and state s_j exists in the topological network.

Results. Figure 4 depicts the results for each hypothesis and dataset at interest. The x-axis denotes the weighting factor k for the number of pseudo counts assigned (cf. Equation 5). The y-axis denotes the corresponding Bayesian evidence (marginal likelihood); for $k = 0$ the evidence is the same for all hypotheses as in that case the pseudo counts are uniformly distributed and no informative aspect is considered. The larger k gets, the more pseudo counts are assigned to the prior according to the given hypothesis and hence, the stronger our belief in specific transitions of a given hypothesis. We can compare hypotheses with each other by comparing the y-values (evidence) for the same x-values. According to Kass and Raftery's interpretation table of log-Bayes factors [21], we find that *all differences are decisive* which is why we refrain from presenting explicit Bayes factors. Hence, the larger the evidence for a given hypothesis is, the more plausible it is in comparison to the other hypotheses at interest. Across all three synthetic datasets, we can observe what we know from theory: the hypothesis that captures the underlying known mechanisms of the synthetic trails best is found to be the most plausible one. In the following we discuss the results of each dataset in detail:

Structural random walk. In Figure 4(a) we can see that the structural hypothesis is ranked as the most plausible one as it exhibits the highest evidences for $k > 0$. This result is as expected from theory as the trails are also produced according to a structural random walk only considering the underlying topological link network as expressed by the structural hypothesis. The reason why the popularity hypothesis is more plausible than the uniform hypothesis is because the former also incorporates structural information while the latter does not.

Popularity random walk. We show the results for our popularity random walk generated trails in Figure 4(b). In this case the popularity hypothesis which incorporates the in-degree (popularity) of potential structural target nodes can be identified as the most plausible one as it captures the mechanisms according to which the trails have been generated.

Random teleportation. Finally, in Figure 4(c) we demonstrate the results for our trails generated via random teleportation. As expected, the uniform hypothesis is the most plausible one which accounts to our prior belief that all target nodes are equally likely to come next given a current node. Contrary, the structural and popularity hypotheses which both incorporate structural knowledge are less plausible hypotheses.

4.2 Experiments with Empirical Data

Our second kind of experiments focus on demonstrating the general applicability of the HypTrails approach by applying it to three real-world, empirical human trail datasets (Wikigame, Yelp and Last.fm) as introduced in Section 3. We compare *universal* as well as *domain-specific* hypotheses for each dataset which we describe next, before we discuss the experimental results.

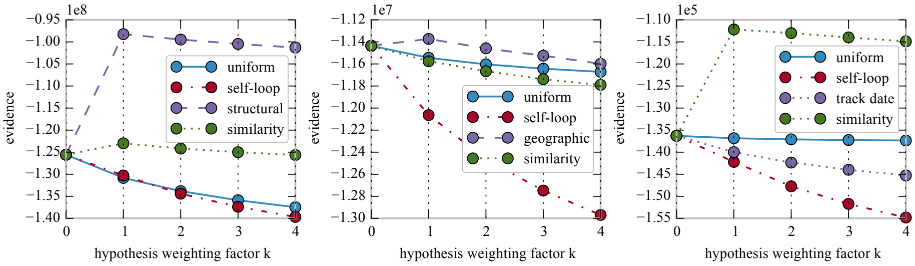
Hypotheses. We now describe the universal and domain-specific hypotheses studied and how we express them. These are just exemplary hypotheses for illustrative purposes, researchers are *completely free to formulate other / their own hypotheses* accordingly.

Uniform hypothesis. We use the universal uniform hypothesis in a similar fashion as for our experiments with synthetic data in order to express our prior belief that each state is equally likely given a current state. Hence, we assign 1 to each element of the hypothesis matrix Q. We can see this hypothesis as a baseline for other hypotheses; if they are not more plausible than the uniform hypothesis, we can not expect them to be good explanations about the behavior that is producing the underlying human trails.

Self-loop hypothesis. With the universal self-loop hypothesis we express our prior belief that humans never switch to another element in a trail. For example, for a navigational scenario this would mean that if a user currently is on a specific Wikipedia page, she would always just refresh the current one and never switch to another one. We set the diagonal to 1 in the corresponding hypothesis matrix Q and leave all other elements zero.

Similarity hypothesis. We use the similarity hypothesis for expressing our belief that humans consecutively target nodes in trails that are in some way (e.g., semantically) related to each other. We now aim at modeling this hypothesis for all three datasets. However, due to their given nature the similarity hypothesis differs for each dataset at interest which is why we describe the domain-specific similarity hypotheses next:

Wikigame similarity hypothesis. This hypothesis states the belief that humans prefer to consecutively access websites that are semantically related which has been observed and hypothesized in a series of previous works (e.g., [38, 47, 48]). Using the set of Wikipedia pages



(a) Wikigame

(b) Yelp

(c) Last.fm

Figure 5: *Experiments with empirical data.* This figure depicts the results obtained from applying HypTrails to three different empirical human trail datasets (Wikigame (a), Yelp (b) and Last.fm (c)) for comparing a set of hypotheses. The x-axis depicts the strength (weighting factor k) one assigns to a given hypothesis as defined in Equation 5 ($k = 0$ refers to a uniform prior) while the y-axis shows the corresponding evidence (marginal likelihood) value. For simplicity, we can compare hypotheses with each other by comparing the evidence values (larger values mean higher plausibility) for the same values of k as all Bayes factors are decisive. Several domain-specific hypotheses are declared as the most plausible ones for our three datasets: the structural hypothesis for the Wikigame trails (a), the geographic hypothesis for the Yelp trails (b) and the artist similarity hypothesis for our Last.fm trails (c).

that users navigate over, we use the textual information of each site provided by the corresponding Wikipedia dump (see Section 3) for calculating the semantic relatedness [34] between sites. We utilize a vector space model [34] for representing the documents (states) as vectors of identifiers using *tf-idf* [35] for weighing the terms of the vectors. We apply an automatic stop word removal process where we ignore all terms that are present in more than 80% of the documents in our corpus. Also, we use sub-linear term frequency scaling as a term that occurs ten times more frequently than another is not necessarily ten times more important [26]. Additionally, we perform a sparse random projection for reducing dimensionality while still guaranteeing Euclidian distance with some error [1, 24]. The final number of parameters is determined by the *Johnson-Lindenstrauss lemma* [15] that states that given our $360,417$ number of samples (distinct Wikipedia pages), we only need $10,942$ features while preserving the results up to a tolerance of 10% – which is also the tolerance level we use for dimensionality reduction. By doing so, we can reduce the number of tf-idf features from $2,285,489$ to the specified $10,942$. Finally, we calculate similarity between all pairs of pages (states) using *cosine similarity* between the described vector representations which define $q_{i,j}$ of our matrix Q . Each $q_{i,j}$ can now exhibit a final value between 0 and 1 where 1 means complete similarity and 0 means no relatedness at all. To increase sparsity we only consider similarities that are equal or larger than 0.1. Additionally, we set the elements of the diagonal of Q to 1.

Yelp similarity hypothesis. With this hypothesis we express our belief that humans choose their next business they review based on similarity to the current business according to their categories (e.g., subsequently reviewing restaurants but not a barber after a restaurant). On Yelp, businesses can get assigned a list of categories that represent them (e.g., restaurant) with we leverage for calculating similarity between them. Again, we use a vector space model for representing businesses as vectors of binary identifiers (category assigned or not assigned). For calculating all-pair similarity scores between businesses we utilize Jaccard similarity ranging from 0 to 1 which determine $q_{i,j}$ of the prior hypothesis matrix Q. The diagonal is set to zero as we do not believe in humans consecutively reviewing the same business.

Last.fm similarity hypothesis. This hypothesis captures our belief that humans consecutively listen to songs on Last.fm if they are produced by the same artist – e.g., only listening to songs by Eros Ramazzotti. Hence, we set elements of the hypothesis matrix Q between two tracks to 1 only if they are from the same artist – the diagonal is set to zero.

Wikigame structural hypothesis. For Wikigame, we evaluate an additional domain-specific hypothesis that captures our prior belief that users navigate the Web (or in this case Wikipedia) primarily by using the underlying topological link structure. The corresponding hypothesis matrix Q can hence be built by looking whether links between sites of our states space S exist in the underlying topological link network G with directed edges $E(G)$ (derived from the Wikipedia dump as stated in Section 3.2). To be precise, the values of the elements $q_{i,j}$ of Q are determined by the number of hyperlinks linking from page s_i to page s_j; mostly, only one hyperlink links from one page to the other. Additionally, we set the diagonal of the matrix to 1 as users might also subsequently navigate the same page by e.g., clicking the refresh button of the browser.

Yelp geographic hypothesis. On Yelp, we also consider the domain-specific hypothesis that the next business a user reviews is one that is geographically close to the current one – we have used this as an example throughout this article (e.g., see Figure 1(b) or Figure 3). For doing so, we start by calculating the *haversine distance* [39] between the longitude and latitude values of all pairs of businesses. As the resulting value (in km) is smaller for geographic close businesses than for far businesses we normalize the values by dividing them by the maximum distance before subtracting them from 1. This leads to final values that range from 0 to 1 where 1 means geographically identical. We set the values of Q according to the calculated values while leaving the diagonal zero.

Last.fm date hypothesis. Finally, we specify a hypothesis that believes that successive tracks listened to on Last.fm are close regarding their original publication date (e.g., someone prefers to only listen to 80s songs). We determine the date of a track by using the Musicbrainz API and looking for the earliest release date available. Next, we calculate the difference between dates of two songs in years.[6] Similar to the Yelp dataset, we then divide each date difference value by the maximum and subtract it from 1 giving us scores between 0 and 1 where the latter means that two tracks are

[6] We only consider track pairs for which we can retrieve a date for both tracks through the API.

originally published in the same year. We set the transition values of Q according to the calculated values and leave the diagonal zero.

Results. The results for all datasets are shown in Figure 5. Again, all Bayes factors are decisive and we can simply interpret hypotheses having larger y-values (evidence, marginal likelihood) as more plausible. Across all datasets, we can identify some domain-specific hypotheses that are more plausible compared to the universal uniform hypothesis which we can see as a baseline. Hence, these hypotheses seem to capture some mechanisms well that human behavior exhibits while producing the human trails studied. Additionally, we find that throughout all datasets the universal self-loop hypothesis is the least plausible one with a small exception for the Wikigame dataset. In the following, we present the results from the different datasets in greater detail:

Wikigame. In Figure 5(a) we present the results of applying HypTrails to the Wikigame dataset comparing the hypotheses at interest. First and foremost, our approach shows largest evidence for the domain-specific structural hypothesis. This indicates that users playing the Wikigame indeed seem to prefer to navigate Wikipedia by following links of the underlying topological link network. This is not too surprising as the Wikigame per definition only allows users to click on available hyperlinks for trailing through the Wikipedia space. Additionally, we can see that the domain-specific similarity hypothesis is more plausible than both the universal uniform as well as the self-loop hypotheses. This corroborates the theories and assumptions of previous work [38, 47, 48] which observed that humans tend to follow semantically related concepts successively.

Furthermore, we observe that both the universal uniform as well as the self-loop hypotheses are the least plausible hypotheses at interest. Interestingly, for $k = 1$ the self-loop hypothesis exhibits larger evidence compared to the uniform prior which partly also demonstrates that self-loops are indeed an important aspect for this dataset as also observed in previous work [37]. However, with larger k the evidence of the uniform hypothesis surpasses the self-loop hypothesis which may be explained by the fact that we weight the informative part (i.e., only self-loops) too strongly while we ignore all other possible transitions.

Yelp. We depict the results for comparing the hypotheses at interest for our Yelp dataset (business reviews) in Figure 5(b). A first observation is that our approach indicates the domain-specific geographic hypothesis as the most plausible one. Hence, humans indeed seem to prefer to successively review businesses that are geographically close to each other on Yelp as captured by our dataset. Contrary, the other domain-specific similarity hypothesis is less evident compared to the uniform hypothesis which can be seen as a baseline. Consequently, from this exemplary analysis, we can not assume that humans prefer to consecutively review the same businesses based on similar categoric descriptors, at least not based on the similarity of categorical descriptors given on Yelp. Finally, the self-loop hypothesis is indicated as the least plausible one which indicates that humans at maximum very seldom review the same business twice in a row in our dataset.

Last.fm. Finally, in Figure 5(c) we illustrate the results obtained when applying HypTrails for comparing our Last.fm hypotheses. In this case, we can see that the similarity hypothesis, expressing our prior belief that users consecutively listen to songs that stem from the same artist, is the most plausible one. This is visible as the evidence values are larger for all $k > 0$ compared to the other hypotheses of interest. Again, we observe that humans do not seem to prefer to listen to the same song over and over again (self-loop hypothesis) in our dataset. Also, for this example data, the track date hypothesis is indicated with lower evidence compared to the universal uniform hypothesis making it less plausible.

5. DISCUSSION

The HypTrails approach represents an intuitive way of comparing hypotheses about human trails as we have demonstrated on synthetic as well as empirical data. However, there are some aspects – as partly exhibited throughout our experiments – that one should consider when applying HypTrails; we discuss them next.

Specification of hypotheses. HypTrails allows researchers to intuitively express their hypotheses as arbitrary matrices (where higher values indicate higher belief), which are then used for eliciting priors. While this is a very intuitive way of expressing hypotheses, choices have to be made regarding several factors such as (i) which transitions to believe in (e.g., about setting the diagonal) (ii) how to calculate values for representing a hypothesis (e.g., haversine distance for geographic closeness) or (iii) availability of information (e.g., API restrictions). While several ways of doing this are conceivable, our approach does not constraint the researchers' choice in this regard. If in doubt, our advice is to express the uncertainty through another hypothesis (which reduces the problem) or through a set of other hypotheses (which focus on different representations). For example, in this article we first investigate the plausibility of a universal self-loop hypothesis compared to a uniform hypothesis before making a choice about the diagonal of other hypotheses. We find that for navigational trails (Wikigame), self-loops seem to play a role at least occasionally (cf. Figure 5(a)) which is why we also set the diagonals in other hypotheses to larger values than zero, while for both Yelp (cf. Figure 5(b)) as well as Last.fm (cf. Figure 5(c)) we can not observe such behavior. Another choice to make is whether one wants to express hypotheses in a symmetric or asymmetric way – e.g., it might be useful to believe that transitioning from state A to state B is more relevant than from B to A. Following our advice, we would express a symmetric and asymmetric version of the hypothesis and compare them.

Behavior of hypothesis weighting factor *k*. Throughout our experimental results (see Figure 4 and Figure 5) we frequently observe that the evidence is falling with larger k. As pointed out throughout this article, the evidence is a weighted likelihood average and is largest if both the prior as well as the likelihood concentrate on the same parameter value regions. The larger we choose k, the larger we set the hyperparameters of the Dirichlet distributions and the more they get concentrated. Thus, only a few specific parameter configurations (single draw from the Dirichlet distribution) receive higher prior probabilities while many others receive low ones. As we can not expect our hypotheses to concentrate on the exact same areas as the likelihoods as we did for our empirically aligned toy example in Figure 2(b), we sometimes see falling evidences with larger k as we reduce the scope of the prior. Again, we want to emphasize that hypotheses should not be compared with each other for different values of k.

Memoryless Markov chain property. Currently, HypTrails is memoryless, meaning that the next state only depends on the current one. Previous work has been contradictory in their statements about memory effects of human trails on the Web (see e.g., [14, 37]). While first order models have mostly been shown to be appropriate, it may be useful to extend HypTrails to also support memory effects in the future. This would mean that it would allow us to not only analyze hypotheses about how the current state influences the next one, but also how past ones (potentially) exert influence.

Ideas for future work. While we have showcased a certain variety of datasets and hypotheses that can be analyzed with HypTrails, we would like to encourage researchers to see these examples only as a stepping stone for more detailed experiments to be conducted. In addition, in future work multiple extensions and / or experimental variations are conceivable. For example, it could be useful to look at personalization or user group effects in data. Currently, the examples only demonstrate collective behavior, but one may assume that different groups of users produce human trails differently. One could segment the dataset according to some heuristic criteria and then analyze the same hypotheses on both sub-datasets. If one hypothesis is more plausible in one dataset than the other, one can assume differences in user behavior in different sub populations. One might also believe that human behavior changes over time [51]. This suggests to apply HypTrails to study the temporal evolution of hypotheses (and evidences for them). Furthermore, one can also think about combining hypotheses with each other to form new ones. For example, in Figure 5(a) we show that both the structural as well as the similarity hypotheses are very plausible to explain navigational behavior on Wikipedia. One could use a combination of both by weighing structural transitions according to their similarity.

6. RELATED WORK

Studies of human trails in information systems have been fuelled by the advent of the World Wide Web [4]. A fundamental way of interacting with the Web is navigating from website to website. Such navigational trails have been extensively investigated in the past. An example of early work is by Catledge and Pitkow [11] who studied navigational regularities and strategies for augmenting the design and usability of WWW pages. Subsequent studies, e.g., the work by Huberman et al. [20] or Chi et al. [13], emphasize existing regularities and rationalities upon which humans base their navigational choices. These examples nicely demonstrate the importance of gaining a better understanding of sequential user behavior producing human trails on the Web. Apart from modeling [7, 8, 13, 32, 36, 37] and the detection of regularities and patterns [20, 44, 45], researchers have also been interested in studying strategies humans follow when producing human trails on the Web. We highlight some exemplary findings next.

A prominent theory is the *Information Foraging theory* by Pirolli and Card [31] which states that human behavior in an information environment on the Web is guided by *information scent* which is based on the cost and value of information with respect to the goal of the user [13]. Another behavioral pattern is shown in [30] and [9] where the authors observe that semantics affect how users search visual interfaces on websites; the importance of semantics between subsequent concepts is also emphasized in [12, 38, 47, 48]. Amongst many others, further studies of human trails on the Web focus on the detection of progression stages [51], trail prediction [22], the study of the value of search trail following for users [5, 49], partisan sharing [2] or approaches to capture trends in human trails [27].

While we highlight just a small excerpt of related work, all these studies reveal interesting behavioral aspects that should be translatable into hypotheses about transitions over states. What is difficult, is to compare them within a coherent research approach. In this work we tackle this problem. Fundamentally, HypTrails is based on a Markov chain model which is prominently leveraged for modeling human trails on the Web. Google's PageRank, for example, is based on a first order Markov chain model [8] and a large array of further studies have highlighted the benefits of Markov chain models for modeling human trails on the Web (e.g., [7, 17, 23, 25, 32, 36, 37, 45, 52]). Given these advantages as well as the fact that we are interested in studying hypotheses about memoryless transitions, the Markov chain model represents a sensible choice for our approach. For deriving the parameters of models, we utilize Bayesian inference [37, 40].

The main idea of our approach is to incorporate hypotheses as informative Dirichlet priors into the Bayesian Markov chain inference and compare them with Bayes factors. Bayes factors are known to be highly sensitive on the prior. This property of Bayes factors has been seen as a limitation in the past – as originally pointed out by Kass and Raftery [21]. However, as emphasized by Wolf Vanpaemel [41], if "models are quantitatively instantiated theories, the prior can be used to capture theory and should therefore be considered as an integral part of the model". In such a case, the sensitivity of Bayes factors on the prior *can be seen as a feature* – i.e., instrumental for gaining new insights into the plausibility of theories (or in our case hypotheses about human trails). Thus, marginal likelihoods and Bayes factors can be leveraged as an appropriate measure for evaluating hypotheses about human trails. The process of expressing theories as informative prior distributions over parameters has been discussed in follow-up work by Wolf Vanepaemel in [43] and in [42] where the author tackles this task by using hierarchical methods. In this work, we present an adaptation of the so-called (trial) roulette method, which was first proposed in [19] and further discussed in [16, 29], for this task. With our adaption, we understand the grid as a hypothesis matrix where elements correspond to beliefs about transitions for a given hypothesis. Also, in our case, chips correspond to pseudo counts of Dirichlet priors which we automatically set according to expressed hypotheses of researchers.

7. CONCLUSION

Understanding human trails on the Web and how they are produced has been an open and complex challenge for our community for years. In this work, we have addressed a sub-problem of this larger challenge by presenting HypTrails– an approach that enables scientists to compare hypotheses about human trails on the Web. HypTrails utilizes Markov chain models with Bayesian inference. The main idea is to incorporate hypotheses as Dirichlet priors into the inference process and leverage the sensitivity of Bayes factors for comparing hypotheses. Our approach allows researchers to intuitively express hypotheses as beliefs about transitions between states which are then used for eliciting priors.

We have experimentally illustrated the general mechanics of HypTrails by comparing hypotheses about synthetic trails that were generated according to controlled mechanisms. As derived from theory, HypTrails ranks those hypotheses as the most plausible ones, that best capture the mechanisms of the underlying trails. Additionally, we have studied empirical data to further show the general applicability of HypTrails. We looked at human trails from three different domains: human navigational trails over Wikipedia articles (Wikigame), successive reviews of businesses (Yelp) as well as trails capturing songs that users consecutively listen to (Last.fm). Although the experiments presented in this work mainly served to illustrate how one can apply the HypTrails approach, we hope that they also motivate and encourage researchers to conduct further, more in-depth studies of human trails on the Web in the future.

While we have developed HypTrails for comparing hypotheses about hypertext trails, the approach is not limited to Web data. It can be applied to any form of trails over states at interest in a straightforward manner; e.g., it could also be used to study human trails as recorded by GPS data. Insights gained by such studies can give a clearer picture of the underlying dynamics of human behavior that shape the production of human trails.

Acknowledgements. This work was partially funded by the DFG German Science Fund grant STR 1191/3-2 and by the FWF Austrian Science Fund research project "Navigability of Decentralized Information Networks" (P24866). Furthermore, we want to thank Alex Clemesha for giving us access to the Wikigame data as well as Daniel Lamprecht for valuable input.

References

[1] D. Achlioptas. Database-friendly random projections. In *Symposium on Principles of Database Systems*, pages 274–281. ACM, 2001.

[2] J. An, D. Quercia, and J. Crowcroft. Partisan sharing: facebook evidence and societal consequences. In *Conference on Online Social Networks*, pages 13–24. ACM, 2014.

[3] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[4] T. Berners-Lee and M. Fischetti. *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. HarperInformation, 2000.

[5] M. Bilenko and R. W. White. Mining the search trails of surfing crowds: identifying relevant websites from user activity. In *International Conference on World Wide Web*, pages 51–60. ACM, 2008.

[6] J. Borges and M. Levene. Data mining of user navigation patterns. In *Web usage analysis and user profiling*, pages 92–112. Springer, 2000.

[7] J. Borges and M. Levene. Evaluating variable-length markov chain models for analysis of user web navigation sessions. *IEEE Transactions on Knowledge and Data Engineering*, 19(4):441–452, Apr. 2007.

[8] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *International Conference on World Wide Web*, pages 107–117. Elsevier Science Publishers B. V., 1998.

[9] D. P. Brumby and A. Howes. Good enough but i'll just check: Web-page search as attentional refocusing. In *International Conference on Cognitive Modeling*, pages 46–51, 2004.

[10] V. Bush. As we may think. *The Atlantic Monthly*, 176(1):101–108, 1945.

[11] L. D. Catledge and J. E. Pitkow. Characterizing browsing strategies in the world-wide web. *Computer Networks and ISDN Systems*, 27(6):1065–1073, 1995.

[12] M. Chalmers, K. Rodden, and D. Brodbeck. The order of things: activity-centred information access. *Computer Networks and ISDN Systems*, 30(1):359–367, 1998.

[13] E. H. Chi, P. L. T. Pirolli, K. Chen, and J. Pitkow. Using information scent to model user information needs and actions and the web. In *Conference on Human Factors in Computing Systems*, pages 490–497. ACM, 2001.

[14] F. Chierichetti, R. Kumar, P. Raghavan, and T. Sarlos. Are web users really markovian? In *International Conference on World Wide Web*, pages 609–618. ACM, 2012.

[15] S. Dasgupta and A. Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.

[16] C. Davidson-Pilon. *Probablistic Programming & Bayesian Methods for Hackers*. 2014.

[17] M. Deshpande and G. Karypis. Selective markov models for predicting web page accesses. *ACM Transactions on Internet Technology*, 4(2):163–184, May 2004.

[18] P. H. Garthwaite, J. B. Kadane, and A. O'Hagan. Statistical methods for eliciting probability distributions. *Journal of the American Statistical Association*, 100(470):680–701, 2005.

[19] S. Gore. Biostatistics and the medical research council. *Medical Research Council News*, 35:19–20, 1987.

[20] B. A. Huberman, P. L. T. Pirolli, J. E. Pitkow, and R. M. Lukose. Strong regularities in world wide web surfing. *Science*, 280(5360):95–97, Mar 1998.

[21] R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

[22] S. Laxman, V. Tankasali, and R. W. White. Stream prediction using a generative model based on frequent episodes in event sequences. In *International Conference on Knowledge Discovery and Data Mining*, pages 453–461. ACM, 2008.

[23] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (salsa) and the tkc effect. *Computer Networks*, 33(1):387–401, June 2000.

[24] P. Li, T. J. Hastie, and K. W. Church. Very sparse random projections. In *International Conference on Knowledge Discovery and Data Mining*, pages 287–296. ACM, 2006.

[25] Z. Li and J. Tian. Testing the suitability of markov chains as web usage models. In *International Conference on Computer Software and Applications*, pages 356–361. IEEE Computer Society, 2003.

[26] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[27] Y. Matsubara, Y. Sakurai, C. Faloutsos, T. Iwata, and M. Yoshikawa. Fast mining and forecasting of complex time-stamped events. In *International Conference on Knowledge Discovery and Data Mining*, pages 271–279. ACM, 2012.

[28] T. H. Nelson. Complex information processing: a file structure for the complex, the changing and the indeterminate. In *National Conference*, pages 84–100. ACM, 1965.

[29] J. Oakley. Eliciting univariate probability distributions. *Rethinking Risk Measurement and Reporting*, 1, 2010.

[30] B. J. Pierce, S. R. Parkinson, and N. Sisson. Effects of semantic similarity, omission probability and number of alternatives in computer menu search. *International Journal of Man-Machine Studies*, 37(5):653–677, 1992.

[31] P. L. T. Pirolli and S. K. Card. Information foraging. *Psychological Review*, 106(4):643–675, 1999.

[32] P. L. T. Pirolli and J. E. Pitkow. Distributions of surfers' paths through the world wide web: Empirical characterizations. *World Wide Web*, 2(1-2):29–45, Jan 1999.

[33] D. d. S. Price. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27(5):292–306, 1976.

[34] H. Rubenstein and J. B. Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, 1965.

[35] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988.

[36] R. Sen and M. Hansen. Predicting a web user's next access based on log data. *Journal of Computational Graphics and Statistics*, 12(1):143–155, 2003.

[37] P. Singer, D. Helic, B. Taraghi, and M. Strohmaier. Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PloS one*, 9(7):e102070, 2014.

[38] P. Singer, T. Niebler, M. Strohmaier, and A. Hotho. Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70, 2013.

[39] R. W. Sinnott. Virtues of the haversine. *Sky and Telescope*, 68(2):158, 1984.

[40] C. C. Strelioff, J. P. Crutchfield, and A. W. Hübler. Inferring markov chains: Bayesian estimation, model comparison, entropy rate, and out-of-class modeling. *Physical Review E*, 76(1):011106, Jul 2007.

[41] W. Vanpaemel. Prior sensitivity in theory testing: An apologia for the bayes factor. *Journal of Mathematical Psychology*, 54(6):491–498, 2010.

[42] W. Vanpaemel. Constructing informative model priors using hierarchical methods. *Journal of Mathematical Psychology*, 55(1):106–117, 2011.

[43] W. Vanpaemel and M. D. Lee. Using priors to formalize theory: Optimal attention and the generalized context model. *Psychonomic Bulletin & Review*, 19(6):1047–1056, 2012.

[44] S. Walk, P. Singer, and M. Strohmaier. Sequential action patterns in collaborative ontology-engineering projects: A case-study in the biomedical domain. In *International Conference on Conference on Information & Knowledge Management*. ACM, 2014.

[45] S. Walk, P. Singer, M. Strohmaier, T. Tudorache, M. A. Musen, and N. F. Noy. Discovering beaten paths in collaborative ontology-engineering projects using markov chains. *Journal of Biomedical Informatics*, 51:254–271, 2014.

[46] L. Wasserman. Bayesian model selection and model averaging. *Journal of Mathematical Psychology*, 44(1):92–107, 2000.

[47] R. West and J. Leskovec. Human wayfinding in information networks. In *International Conference on World Wide Web*, pages 619–628. ACM, 2012.

[48] R. West, J. Pineau, and D. Precup. Wikispeedia: An online game for inferring semantic distances between concepts. In *International Joint Conference on Artifical Intelligence*, pages 1598–1603. Morgan Kaufmann Publishers Inc., 2009.

[49] R. W. White and J. Huang. Assessing the scenic route: measuring the value of search trails in web logs. In *Conference on Research and Development in Information Retrieval*, pages 587–594. ACM, 2010.

[50] W. Xie, P. O. Lewis, Y. Fan, L. Kuo, and M.-H. Chen. Improving marginal likelihood estimation for bayesian phylogenetic model selection. *Systematic Biology*, 60(2):150–160, 2010.

[51] J. Yang, J. McAuley, J. Leskovec, P. LePendu, and N. Shah. Finding progression stages in time-evolving event sequences. In *International Conference on World Wide Web*, pages 783–794. ACM, 2014.

[52] I. Zukerman, D. W. Albrecht, and A. E. Nicholson. Predicting users' requests on the www. In *International Conference on User Modeling*, pages 275–284. Springer, 1999.

4. Conclusions

The production of human trails on the Web is ubiquitous and our research community has been interested in modeling aspects of human trails on the Web since the advent of the Web. Understanding the production of human trails can be useful for a series of applications such as for enhancing information network structures, predicting human trails or for recommender systems. However, it has been a complex challenge for years. In this thesis, I have tackled this challenge by mainly concentrating on methodological tools that should provide researchers with straight-forward mechanisms to study human trails on the Web. To that end, I have focused on several sub-problems and methodological contributions. First of all, related research has been contradictory regarding their statements about whether it is useful to incorporate memory effects when modeling human trails on the Web. By presenting *a general framework for detecting the appropriate Markov chain order – i.e., memory effects – given human trail data in a comprehensive way*, I want to encourage and facilitate future studies. Next, this thesis highlights several regularities, patterns and strategies in human trails on the Web as well as demonstrates that *human navigational trails can be successfully leveraged for the task of calculating semantic relatedness between concepts.* This argues for an augmentation of existing methods to also consider human trails. Finally, I have presented *an approach called HypTrails that allows researchers to express and compare hypotheses – beliefs in transitions – about human trails.* Next, this final chapter summarizes the results and contributions in detail before I highlight implications, limitations and potential future work.

4.1. Results and Contributions

To summarize the results and contributions of this thesis, I give answers to the research questions as defined in Section 1.4.

Memory and Structure in human trails on the Web. Previous works have been quite contradictory regarding whether human trails on the Web exhibit memory effects or not. This is of specific interest to the Markov chain model which has been memoryless in a wide range of human trail applications such as Google's PageRank. Thus, the first research question of this thesis to be answered has been *"What is the structure and memory in human trails on the Web?"*. To that end, I have presented *a general framework for detecting the appropriate Markov chain order* in Section 3.2 which is one of the main contributions of this thesis. By utilizing a series of advanced statistical inference methods, this framework allows researchers to comprehensively make informed decisions about the appropriate Markov chain order as well as potential memory effects in human trail data. By applying this framework to human navigational trails, the findings demonstrate what this thesis inferred from theory in Chapter 2: It is indeed difficult to detect the appropriate Markov chain order having insufficient data but a vast amount of states. However, by limiting the state space by replacing pages with corresponding topics, the results suggest that human navigational behavior is guided by memory effects at least on some level. In subsequent work as presented in Section 3.3 and Section 3.4, colleagues and I have demonstrated the general applicability and potential use cases of the Markov chain framework by applying it to human edit trails in collaborative ontology engineering projects. The results demonstrate that the framework can be successfully utilized for eliciting not only memory effects, but also regularities and structural patterns in various kinds of human trails on the Web. Additionally, experiments of Section 3.4 showcase the importance of memory effects by demonstrating that higher order Markov chain models can be more accurate when predicting human trails by still accounting for potential overfitting. By and large, future researchers can benefit from the developed framework by applying it to their problem setting and data of interest.

Leveraging human trails. Based on the findings regarding memory and structure in human trails on the Web, I have hypothesized that it may be also beneficial to leverage human trails on the Web for tasks that are usually solved by using Web content only. As an exemplary use case, the second research question of this thesis has questioned *"Can we leverage human navigational trails for the task of calculating semantic relatedness between concepts?"*. This question has been tackled and answered in Section 3.5 by studying human navigational trails through Wikipedia pages captured by logs of the Wikigame. The main idea has been that closeness of concepts in trails can be an indicator for their semantic relatedness. For example, if humans frequently navigate between the Wikipedia pages of Austria and Graz, we might consider them as being semantically related to some degree. To automatically capture this in an intuitive way, the presented work has utilized a vector space model using co-occurrence information between concepts. By evaluating the semantic relatedness scores on a set of gold-standards and baseline corpora, the results indeed have shown that we can successfully leverage human navigational trails for the task of calculating proper semantic relatedness scores between concepts. Additionally, by extending the neighborhood considered for co-occurrence information, we can improve the overall quality of determined semantic relatedness scores. However, this work has also highlighted that not all navigational trails are equally useful; intelligent selection of trail corpora can enhance accuracy. This thesis suggests that we can indeed harness human trails on the Web for knowledge inferring tasks based on the exemplary application of calculating semantic relatedness between concepts. This is the second main contribution of this thesis and argues for existing and future methods to also consider human trails on the Web for their tasks.

Comparing hypotheses about human trails. Given the patterns, regularities and strategies such as that humans seem to navigate over semantically similar websites found in this thesis, the third and final research question has been *"How can we compare hypotheses about human trails on the Web"*. For tackling this question, I have presented an approach called HypTrails in Section 3.6. This approach allows researchers to intuitively express and compare an array of hypotheses about human trails. Hypotheses can be seen as beliefs about common transitions.

HypTrails utilizes a first-order Markov chain model for modeling the data. For inference, it resorts to Bayesian statistics and the main idea has been to incorporate hypotheses as informative Dirichlet priors into the inference process. By then using marginal likelihoods and Bayes factors, the approach can make informed decisions about the plausibility of given hypotheses. Technically, it makes use of the sensitivity of Bayes factors on the prior. For demonstrating the general mechanics and applicability of HypTrails, this thesis has applied it to a set of distinct human trail datasets from various domains: (i) business reviews on Yelp, (ii) human navigational trails on Wikipedia and (iii) successive songs listened to on Last.fm supplemented with experiments on (iv) synthetic trails produced according to know mechanisms. Overall, HypTrails is the final main contribution of this thesis and allows researchers to further understand the production of human trails on the Web.

In order to facilitate reproducibility and future applications as well as fuel future studies on human trails on the Web, most of the methods developed in this thesis are made available open-source and online:

- Framework for comprehensively detecting the appropriate Markov chain order given human trails on the Web based on several advanced statistical inference methods as introduced in [Singer et al., 2014c] (Section 3.2): `https://github.com/psinger/PathTools`.

- Additional test called *runs test* for studying regularities and randomness in categorical trails as utilized in [Walk et al., 2014a] (Section 3.4): `https://github.com/psinger/RunsTest`.

- An implementation of several methods for computing statistical significance tests on both dependent and independent correlation coefficients as used in [Singer et al., 2013a] (Section 3.5): `https://github.com/psinger/CorrelationStats`

- The vector space method using co-occurrence information in human trails as applied in [Singer et al., 2013a] (Section 3.5): `https://github.com/psinger/PathTools`

- Finally, the HypTails approach for comparing hypotheses about human trails as introduced in [Singer et al., 2014b] (Section 3.6): `https://github.com/psinger/HypTrails`

4.2. Implications and Potential Applications

A better understanding of human trails as well as the development of new modeling tools is necessary for improving various aspects on the Web such as user interfaces, information network structure or recommender systems. This thesis provides a further stepping stone for this larger challenge. The main contributions of this thesis are methodologically. I am confident that future research can benefit from the tools and insights offered by this thesis for better understanding the production of human trails on the Web. In the following, I want to discuss some implications and potential applications.

Incorporating memory effects into models and applications. When studying memory in human trails on the Web in this thesis, I have not only focused on Markov chain models of varying order for detecting these effects, but also on Markov chain models as an example of an application that can benefit from incorporating them. However, memory effects can be a useful extension for other kinds of models and applications. For example, recommender systems might be improved by considering longer (or compound) histories of behavioral patterns for future recommendations. As a further potential application I want to mention network models such as spreading models that predominantly focus on neighboring effects only. While it is difficult to make general suggestions about memory effects as they are highly dependent on the choice of data, sample size and complexity as showcased throughout this thesis, the framework offered in this thesis allows every researcher to evaluate memory effects given their own problem setting. By doing so, they can make informed decisions about whether it is useful to incorporate memory effects into their applied models.

Augmenting existing methods with human trail data. Throughout this thesis, I have made several arguments about the inherent regularities, patterns and strategies in human trails on the Web. Based on these observations, I have shown that we can also leverage these patterns in human trails on the Web for inferring knowledge. As an example, I have showcased that we can successfully harness human navigational trails on the Web for calculating accurate semantic relatedness scores between concepts. Hence, I argue that researchers should consider to enrich existing methods by also utilizing human trails on the Web. While the calculation of semantic relatedness is a prominent example, several other methods could benefit from such an augmentation. For instance, machine learning algorithms applied to Web data are mostly limited to content produced by a small set of people. By considering the arguments of this thesis and by supplement this kind of data with pragmatic usage patterns such as human trails, we might be capable of improving the accuracy of corresponding models. Also, note that while this thesis has focused on human navigational trails for this task, the ideas can be extended to other types of human trails on the Web.

Actions based on a better understanding of human trails on the Web. Finally, the tools presented in this thesis allow researchers to get a much clearer picture about the production of human trails at interest. This starts with the structural patterns that can be identified with the Markov chain model and ends with the HypTrails approach that allows researchers to directly compare hypotheses about human trails with each other. Such insights can not only be important for decisions about models, but also for implementing actions. Suppose a researcher has several hypotheses about how humans behave on her platform. By comparing them with HypTrails, the researcher can evaluate them. Now, she can decide whether she wants to further steer these behavioral aspects or maybe counteract them by making changes to the platform. Furthermore, these insights can unfold unexplored behavioral aspects that may be leveraged similar to the calculation of semantic relatedness between concepts.

4.3. Limitations and Future Work

Limitations. This thesis comes with certain limitations which I want to list next.

- **Lack of generality of empirical findings.** While this thesis has heavily focused on demonstrating the developed tools, the empirical findings lack generality. This is a consequence from the fact that the empirical observations are driven by the specific datasets of human trails studied. However, the main contributions of this thesis focus on providing tools to researchers for studying human trails on the Web. Hence, the empirical findings should be seen as an encouragement for further studies that can be conducted by the methods, frameworks and approaches developed in this thesis. While being developed for hypertext trails, the methods are not limited to Web data. For example, one can apply them to trails of whereabouts of humans as captured by GPS data.
- **Data restrictions.** Apart from the lack of generality, it is difficult to acquire appropriate human navigational trails as they mostly have to be captured through logs of websites which are subject to certain privacy restriction. As a consequence, I have heavily focused on studying human navigational trails derived from game data (Wikispeedia and Wikigame). While such game-based data can only be seen as a proxy for real navigation, it also brings some advantages such as that we know the start and target nodes of trails. I have also made use of this by comparing such goal-oriented navigation with free form navigation. Additionally, I frame this thesis to study *human* trails on the Web. However, given the massive amount of bots, crawlers and automated scripts operating on the Web, it might happen that some trails are not produced by actual humans. Hence, future studies should keep that in mind when generating their data to study. Yet, I also see an opportunity in this potential limitation; if we are able to identify non-human trails in our data, we might also be able to learn more about trails that are produced artificially by algorithms. It might even be possible to compare this behavior to human behavior to see how they differ.

- **Limitation of data size.** As highlighted throughout this work, the detection of the appropriate Markov chain order given finite data is a difficult task. The reason for this is the much higher complexity of higher order Markov chain models. Hence, the more states and parameters we are interested in, the more difficult it gets to find statistically significant improvements of higher order models. Thus, we do not necessarily know what the results would be if applied to much larger datasets. However, I want to again emphasize that this thesis offers the tools to conduct such studied in a straight-forward way in future.

- **Methodological restrictions.** Finally, next to empirical generality, this thesis is also limited in regards to the methodological concepts utilized. For most of the experiments, this thesis has applied Markov chain models. I argue for the usefulness of this approach based on previous studies as well as on the benefits this model has shown in the past for various applications such as Google's PageRank. However, we might be able to explain and model human trails on the Web in a better or simply different way by utilizing other models.

Future work. Finally, I want to highlight some potential future works that are partly influenced by mentioned limitations of this thesis.

- **Extending the type of trails studied.** As mentioned, I have limited the empirical investigations to a set of human trail data. There are still a lot of various other types of human trails that may be beneficial to better understand. For example, recent studies have been interested in better understanding diffusion on the Web. If we see diffusion processes as trails, we can apply the methodological concepts developed in this thesis in a straight-forward manner. This may allow for a better understanding of these processes. But also other types of human trails might be of interest to study; to just name a few: (i) trails of bug reports, (ii) trails of friends added on social media platforms or (iii) trails of persons employed as derived from professional social networks such as LinkedIn.

- **Extending the methodological concepts.** This thesis has heavily focused on the application of Markov chain models for tackling the research problems. While perfectly suited for this thesis, I plan on extending the methods in future. For example, I want to tap into the usefulness of further variations of Markov chain models such as the hidden Markov model, varying order Markov models or semi Markov models. Furthermore, the HypTrails approach currently is limited to a first-order Markov chain model. In future, I want to extend this approach to also consider memory effects when comparing hypotheses with each other.

- **Considering memory effects for other models.** In this thesis, I have mainly studied the usefulness of incorporating memory effects into Markov chain models. However, as mentioned above, memory effects might also play an important role in other models. In future, I plan on exploring this by studying models like network spreading models. By doing so, we may be able to capture further patterns and effects that may improve corresponding models.

- **Comparing sub-corpora with each other.** When calculating semantic relatedness between concepts, this thesis has shown that not all trails are equally useful and intelligent selection of trail corpora can enhance accuracy. This argues that not all humans behave similarly and warrants further studies. Hence, in future I also want to compare different sub-corpora of human trails with each other. These sub-corpora can be built according to certain characteristics of humans such as their experience in the system.

- **Web applications.** On the one hand, this thesis has developed methodological tools and on the other hand, it has applied them to real-world human trail data. What is missing, is to use the methods and insights for enhancing Web applications such as pointed out throughout this work. For example, we might aim to incorporate memory effects into an active recommendation algorithm on a Web platform. This would not only allow us to potentially enhance human satisfaction, but also to further evaluate the models at hand.

With this thesis, I hope to facilitate and encourage future research studying human trails on the Web. Our research community can benefit from the tools developed in this thesis. Also, I am confident that human trails on the Web should be utilized beyond what previous work has done. In this thesis, I have provided several incentives for doing so.

List of Figures

List of Tables

Bibliography

Aguilar, C. M. and Medin, D. L. (1999). Asymmetries of comparison. *Psychon. Bull. Rev.*, 6(2):328–337.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *International Symposium on Information Theory*.

Archak, N., Mirrokni, V. S., and Muthukrishnan, S. (2010). Mining advertiser-specific user behavior using adfactors. In *International Conference on World Wide Web*.

Baigorri, A., Gonçalves, C., and Resende, P. (2009). Markov chain order estimation and relative entropy. *arXiv:0910.0264 [math.ST]*.

Banerjee, S. and Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *International Joint Conference on Artificial Intelligence*.

Bartlett, M. (1951). The frequency goodness of fit test for probability chains. In *Mathematical Cambridge Philosophical Society*.

Bellman, R. (1957). A markovian decision process. *Journal of Mathematics and Mechanics*, 6(5):679–684.

Berners-Lee, T. and Fischetti, M. (2000). *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. HarperInformation.

Bestavros, A. (1995). Using speculation to reduce server load and service time on the www. In *International Conference on Information and Knowledge Management*.

Bilenko, M. and White, R. W. (2008). Mining the search trails of surfing crowds: identifying relevant websites from user activity. In *International Conference on World Wide Web*.

Borges, J. and Levene, M. (2000). Data mining of user navigation patterns. In *Web usage analysis and user profiling*, pages 92–112. Springer.

Box, G. E. and Tiao, G. C. (2011). *Bayesian inference in statistical analysis*. John Wiley & Sons.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. In *International Conference on World Wide Web*.

Budanitsky, A. and Hirst, G. (2001). Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources*.

Budanitsky, A. and Hirst, G. (2006). Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

Bühlmann, P., Wyner, A. J., et al. (1999). Variable length markov chains. *The Annals of Statistics*, 27(2):480–513.

Burnham, K. P. and Anderson, D. R. (2002). *Model selection and multimodel inference: a practical information-theoretic approach*. Springer.

Burnham, K. P. and Anderson, D. R. (2004). Multimodel inference understanding aic and bic in model selection. *Sociological Methods & Research*, 33(2):261–304.

Bush, V. (1945). As we may think. *The Atlantic Monthly*, 176(1):101–108.

Cadez, I., Heckerman, D., Meek, C., Smyth, P., and White, S. (2003). Model-based clustering and visualization of navigation patterns on a web site. *Data Mining and Knowledge Discovery*, 7(4):399–424.

Catledge, L. D. and Pitkow, J. E. (1995). Characterizing browsing strategies in the world-wide web. *Computer Networks and ISDN Systems*, 27(6):1065–1073.

Chalmers, M., Rodden, K., and Brodbeck, D. (1998). The order of things: activity-centred information access. *Computer Networks and ISDN Systems*, 30(1):359–367.

Chi, E. H., Pirolli, P. L. T., Chen, K., and Pitkow, J. (2001). Using information scent to model user information needs and actions and the web. In *Conference on Human Factors in Computing Systems*.

Chierichetti, F., Kumar, R., Raghavan, P., and Sarlos, T. (2012). Are web users really markovian? In *International Conference on World Wide Web*.

Chierichetti, F., Kumar, R., and Tomkins, A. (2010). Stochastic models for tabbed browsing. In *International Conference on World Wide Web*.

Cohen, J. (1994). The earth is round (p¡.05). *American Psychologist*, 49(12):997.

Csiszár, I. and Shields, P. C. (2000). The consistency of the bic markov order estimator. *The Annals of Statistics*, 28(6):1601–1619.

Cunha, C. R. and Jaccoud, C. F. (1997). Determining www user's next access and its application to pre-fetching. In *Symposium on Computers and Communications*.

Davidson-Pilon, C. (2014). *Probablistic Programming & Bayesian Methods for Hackers*.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Doerfel, S., Zoller, D., Singer, P., Niebler, T., Hotho, A., and Strohmaier, M. (2014). How social is social tagging? In *International Conference on World Wide Web Companion*.

Downey, D., Dumais, S., Liebling, D., and Horvitz, E. (2008). Understanding the relationship between searchers' queries and information goals. In *International Conference on Information and Knowledge Management*.

Falconer, S., Tudorache, T., and Noy, N. F. (2011). An analysis of collaborative patterns in large-scale ontology development projects. In *International Conference on Knowledge Capture*.

Fienberg, S. E. et al. (2006). When did bayesian inference become" bayesian"? *Bayesian Analysis*, 1(1):1–40.

Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 222:309–368.

Fisher, R. A. (1925). *Statistical methods for research workers*. Genesis Publishing Pvt Ltd.

Gabriel, K. and Neumann, J. (1962). A markov chain model for daily rainfall occurrence at tel aviv. *Quarterly Journal of the Royal Meteorological Society*, 88(375):90–95.

Gabrilovich, E. and Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *International Joint Conference for Artificial Intelligence*.

Garthwaite, P. H., Kadane, J. B., and O'Hagan, A. (2005). Statistical methods for eliciting probability distributions. *Journal of the American Statistical Association*, 100(470):680–701.

Gates, P. and Tong, H. (1976). On markov chain modeling to some weather data. *Journal of Applied Meteorology and Climatology*, 15(11):1145–1151.

Gilks, W. R. (2005). *Markov chain monte carlo*. Wiley Online Library.

Gonçalves, B., Meiss, M. R., Ramasco, J. J., Flammini, A., and Menczer, F. (2009). Remembering what we like: Toward an agent-based model of web traffic. *arXiv:0901.3839 [cs.HC]*.

Goodman, S. (2008). A dirty dozen: Twelve p-value misconceptions. In *Seminars in Hematology*.

Goodman, S. N. (1999). Toward evidence-based medical statistics. 1: The p value fallacy. *Annals of Internal Medicine*, 130(12):995–1004.

Gore, S. (1987). Biostatistics and the medical research council. *Medical Research Council News*, 35:19–20.

Harispe, S., Ranwez, S., Janaqi, S., and Montmain, J. (2013). Semantic measures for the comparison of units of language, concepts or entities from text and knowledge base analysis. *arXiv:1310.1285 [cs.CL]*.

Hayes, B. et al. (2013). First links in the markov chain. *American Scientist*, 101(2):92.

Helic, D., Strohmaier, M., Trattner, C., Muhr, M., and Lerman, K. (2011). Pragmatic evaluation of folksonomies. In *International Conference on World Wide Web*.

Huberman, B. A. and Adamic, L. A. (1998). Novelty and social search in the world wide web. *arXiv:cs/9809025 [cs.MA]*, cs.MA/9809025.

Huberman, B. A., Pirolli, P. L. T., Pitkow, J. E., and Lukose, R. M. (1998). Strong regularities in world wide web surfing. *Science*, 280(5360):95–97.

Huelsenbeck, J. and Andolfatto, P. (2007). Inference of population structure under a dirichlet process model. *Genetics*, 175(4):1787–1802.

Ito, M., Nakayama, K., Hara, T., and Nishio, S. (2008). Association thesaurus construction methods based on link co-occurrence analysis for wikipedia. In *International Conference on Information and Knowledge Management*.

Jiang, J. and Conrath, D. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *International Conference on Research in Computational Linguistics*.

Karplus, K., Barrett, C., and Hughey, R. (1998). Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14(10):846–856.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795.

Katz, R. W. (1981). On some criteria for estimating the order of a markov chain. *Technometrics*, 23(3):243–249.

Kijima, M. and Komoribayashi, K. (1998). A markov chain model for valuing credit risk derivatives. *The Journal of Derivatives*, 6(1):97–108.

Kindermann, R., Snell, J. L., et al. (1980). *Markov random fields and their applications*. American Mathematical Society Providence, RI.

Kolmogoroff, A. (1936). Zur theorie der markoffschen ketten. *Mathematische Annalen*, 112(1):155–160.

Kozima, H. (1993). Computing lexical cohesion as a tool for text analysis. Technical report.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Kurzweil, R. (2012). *How to create a mind: The secret of human thought revealed*. Penguin.

Landauer, T. K., Foltz, P. W., and Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284.

Li, S. Z. (1995). *Markov random field modeling in computer vision*. Springer-Verlag New York, Inc.

Li, Y., Bandar, Z. A., and McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):871–882.

Loftus, G. R. (1996). Psychology will be a much better science when we change the way we analyze data. *Current Directions in Psychological Science*, pages 161–171.

MacKay, D. J. (1992). *Bayesian methods for adaptive models*. PhD thesis, California Institute of Technology.

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

Manabu, O. and Takeo, H. (1994). Word sense disambiguation and text segmentation based on lexical cohesion. In *Conference on Computational Linguistics*.

Markov, A. A. (1906). Rasprostranenie zakona bol'shih chisel na velichiny, zavisyaschie drug ot druga. *Izvestiya Fiziko-Matematicheskogo Obschestva Pri Kazanskom Universitete*, 15(135-156):18.

Markov, A. A. (2006). An example of statistical investigation of the text eugene onegin concerning the connection of samples in chains. *Science in Context*, 19(04):591–600.

Medin, D. L., Goldstone, R. L., and Gentner, D. (1993). Respects for similarity. *Psychological Review*, 100:254–278.

Meiss, M. R., Gonçalves, B., Ramasco, J. J., Flammini, A., and Menczer, F. (2010). Agents, bookmarks and clicks: a topical model of web navigation. In *Conference on Hypertext and Hypermedia.*

Metzler, D. and Croft, W. B. (2005). A markov random field model for term dependencies. In *International Conference on Research and Development in Information Retrieval.*

Miller, G. A. (1995). Wordnet: a lexical database for english. *Commun. ACM*, 38(11):39–41.

Milne, D. (2008). Computing semantic relatedness using wikipedia link structure. In *New Zealand Computer Science Research Student Conference.*

Morrison, D. E. and Henkel, R. E. (2006). *The significance test controversy: A reader.* Transaction Publishers.

Murphy, K. P. (2002). The encyclopedia of cognitive science.

Murray, I. and Ghahramani, Z. (2005). A note on the evidence and bayesian occam's razor.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193(0):217 – 250.

Nelson, T. H. (1965). Complex information processing: a file structure for the complex, the changing and the indeterminate. In *National Conference.*

Neyman, J. and Pearson, E. S. (1992). *On the problem of the most efficient tests of statistical hypotheses.* Springer.

Nicholson, A. E., Zukerman, I., and Albrecht, D. W. (1998). A decision-theoretic approach for pre-sending information on the www. In *Pacific Rim International Conference on Artificial Intelligence.*

Niebler, T., Singer, P., Benz, D., Körner, C., Strohmaier, M., and Hotho, A. (2013). How tagging pragmatics influence tag sense discovery in social annotation systems. In *European Conference on Information Retrieval.*

Nix, A. E. and Vose, M. D. (1992). Modeling genetic algorithms with markov chains. *Annals of Mathematics and Artificial Intelligence*, 5(1):79–88.

Nonnecke, B. and Preece, J. (2000). Lurker demographics: Counting the silent. In *Conference on Human Factors in Computing Systems.*

Nuzzo, R. (2014). Scientific method: Statistical errors. *Nature*, 506(7487):150–152.

Oakley, J. (2010). Eliciting univariate probability distributions. *Rethinking Risk Measurement and Reporting*, 1.

Olston, C. and Chi, E. H. (2003). Scenttrails: Integrating browsing and searching on the web. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 10(3):177–197.

Padmanabhan, V. N. and Mogul, J. C. (1996). Using predictive prefetching to improve world wide web latency. *ACM SIGCOMM Computer Communication Review*, 26(3):22–36.

Patwardhan, S. (2006). Using wordnet-based context vectors to estimate the semantic relatedness of concepts. In *Workshop on Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics Together.*

Pedersen, T., Pakhomov, S. V., Patwardhan, S., and Chute, C. G. (2007). Measures of semantic similarity and relatedness in the biomedical domain. *Journal of Biomedical Informatics*, 40(3):288–299.

Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). Wordnet::similarity: measuring the relatedness of concepts. In *Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics.*

Perkowitz, M. and Etzioni, O. (1997). Adaptive web sites: an ai challenge. In *International Joint Conference on Artificial Intelligence.*

Perneger, T. V. and Courvoisier, D. S. (2010). Interpretation of evidence in data by untrained medical students: a scenario-based study. *BMC Med Res Methodol*, 10(1):78.

Pierce, B. J., Parkinson, S. R., and Sisson, N. (1992). Effects of semantic similarity, omission probability and number of alternatives in computer menu search. *International Journal of Man-Machine Studies*, 37(5):653–677.

Pirolli, P. L. T. and Card, S. K. (1999). Information foraging. *Psychological Review*, 106(4):643–675.

Pirolli, P. L. T. and Pitkow, J. E. (1999). Distributions of surfers' paths through the world wide web: Empirical characterizations. *World Wide Web*, 2(1-2):29–45.

Posch, L., Wagner, C., Singer, P., and Strohmaier, M. (2013). Meaning as collective use: Predicting hashtag semantics on twitter. In *International Conference on World Wide Web Companion.*

Pöschko, J., Strohmaier, M., Tudorache, T., Noy, N. F., and Musen, M. A. (2012). Pragmatic analysis of crowd-based knowledge production systems with icat analytics: Visualizing changes to the ICD-11 ontology. In *Spring Symposium on Wisdowm of the Crowd.*

Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Rabiner, L. and Juang, B.-H. (1986). An introduction to hidden markov models. *ASSP Magazine, IEEE*, 3(1):4–16.

Rendle, S., Freudenthaler, C., and Schmidt-Thieme, L. (2010). Factorizing personalized markov chains for next-basket recommendation. In *International Conference on World Wide Web*.

Resnik, P. (1998). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130.

Rosvall, M., Esquivel, A. V., Lancichinetti, A., West, J. D., and Lambiotte, R. (2014). Memory in network flows and its effects on spreading dynamics and community detection. *Nature Communications*, 5.

Royall, R. (1997). *Statistical evidence: a likelihood paradigm*. CRC press.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Sarukkai, R. R. (2000). Link prediction and path analysis using markov chains. *Computer Networks*, 33(1):377–386.

Scaria, A. T., Philip, R. M., West, R., and Leskovec, J. (2014). The last click: why users give up information network navigation. In *International Conference on Web Search and Data Mining*.

Schöfegger, K., Körner, C., Singer, P., and Granitzer, M. (2012). Learning user characteristics from social tagging behavior. In *Conference on Hypertext and Social Media*.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Sen, R. and Hansen, M. (2003). Predicting a web user's next access based on log data. *Journal of Computational Graphics and Statistics*, 12(1):143–155.

Singer, P. (2014). Understanding, leveraging and improving human navigation on the web. In *International Conference on World Wide Web Companion*.

Singer, P., Flöck, F., Meinhart, C., Zeitfogel, E., and Strohmaier, M. (2014a). Evolution of reddit: From the front page of the internet to a self-referential community? In *International Conference on World Wide Web Companion*.

Singer, P., Helic, D., Hotho, A., and Strohmaier, M. (2014b). Hyptrails: A bayesian approach for comparing hypotheses about human trails. *arXiv:1411.2844 [cs.SI]*.

Singer, P., Helic, D., Taraghi, B., and Strohmaier, M. (2014c). Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PloS One*, 9(7):e102070.

Singer, P., Niebler, T., Strohmaier, M., and Hotho, A. (2013a). Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70.

Singer, P., Niebler, T., Strohmaier, M., and Hotho, A. (2013b). Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70.

Singer, P., Wagner, C., and Strohmaier, M. (2012). Understanding co-evolution of social and content networks on twitter. In *Workshop on Making Sense of Microposts*.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380.

Song, C., Qu, Z., Blumm, N., and Barabási, A.-L. (2010). Limits of predictability in human mobility. *Science*, 327(5968):1018–1021.

Srihari, R. K., Zhang, Z., and Rao, A. (2000). Intelligent indexing and semantic retrieval of multimodal documents. *Inf. Retr.*, 2(2-3):245–275.

Steuer, R., Kurths, J., Daub, C. O., Weise, J., and Selbig, J. (2002). The mutual information: detecting and evaluating dependencies between variables. *Bioinformatics*, 18(suppl 2):S231–S240.

Stigler, S. M. (2002). *Statistics on the table: The history of statistical concepts and methods*. Harvard University Press.

Strelioff, C. C., Crutchfield, J. P., and Hübler, A. W. (2007). Inferring markov chains: Bayesian estimation, model comparison, entropy rate, and out-of-class modeling. *Physical Review E*, 76(1):011106.

Strohmaier, M., Helic, D., Benz, D., Koerner, C., and Kern, R. (2012). Evaluation of folksonomy induction algorithms. *ACM Transactions on Intelligent Systems and Technology*, 3(4):74:1–74:22.

Strohmaier, M., Walk, S., Pöschko, J., Lamprecht, D., Tudorache, T., Nyulas, C., Musen, M. A., and Noy, N. F. (2013). How ontologies are made: Studying the hidden social dynamics behind collaborative ontology engineering projects. *Web Semantics: Science, Services and Agents on the World Wide Web*, 20:18–34.

Strube, M. and Ponzetto, S. P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *Conference on Artificial Intelligence*.

Talmi, D. and Moscovitch, M. (2004). Can semantic relatedness explain the enhancement of memory for emotional words? *Memory & Cognition*, 32(5):742–51.

Teevan, J., Alvarado, C., Ackerman, M. S., and Karger, D. R. (2004). The perfect search engine is not enough: a study of orienteering behavior in directed search. In *Conference on Human Factors in Computing Systems*.

Tong, H. (1975). Determination of the order of a markov chain by akaike's information criterion. *Journal of Applied Probability*, 12(3):488–497.

Trattner, C., Singer, P., Helic, D., and Strohmaier, M. (2012). Exploring the differences and similarities between hierarchical decentralized search and human navigation in information networks. In *International Conference on Knowledge Management and Knowledge Technologies*.

Turdakov, D. and Velikhov, P. (2008). Semantic relatedness metric for wikipedia concepts based on link analysis and its application to word sense disambiguation. In *Colloquium on Databases and Information Systems.*

Tversky, A. (1977). Features of similarity. *Psychological Review*, 84:327–352.

Vanpaemel, W. (2010). Prior sensitivity in theory testing: An apologia for the bayes factor. *Journal of Mathematical Psychology*, 54(6):491–498.

Vanpaemel, W. (2011). Constructing informative model priors using hierarchical methods. *Journal of Mathematical Psychology*, 55(1):106–117.

Vanpaemel, W. and Lee, M. D. (2012). Using priors to formalize theory: Optimal attention and the generalized context model. *Psychonomic Bulletin & Review*, 19(6):1047–1056.

Wagner, C., Singer, P., Posch, L., and Strohmaier, M. (2013). The wisdom of the audience: An empirical study of social semantics in twitter streams. In *European Semantic Web Conference.*

Wagner, C., Singer, P., and Strohmaier, M. (2014a). The nature and evolution of online food preferences. *EPJ Data Science.* accepted/to be published.

Wagner, C., Singer, P., and Strohmaier, M. (2014b). Spatial and temporal patterns of online food preferences. In *International Conference on World Wide Web Companion.*

Wagner, C., Singer, P., Strohmaier, M., and Huberman, B. (2014c). Semantic stability and implicit consensus in social tagging systems. *IEEE Transactions on Computational Social Systems*, 1(1):108–120.

Wagner, C., Singer, P., Strohmaier, M., and Huberman, B. A. (2014d). Semantic stability in social tagging streams. In *International Conference on World Wide Web.*

Walk, S., Pöschko, J., Strohmaier, M., Andrews, K., Tudorache, T., Nyulas, C., Noy, N. F., and Musen, M. A. (2013). Pragmatix: An interactive tool for visualizing the creation process behind collaboratively engineered ontologies. *International Journal on Semantic Web and Information Systems*, 9(1):45–78.

Walk, S., Singer, P., and Strohmaier, M. (2014a). Sequential action patterns in collaborative ontology-engineering projects: A case-study in the biomedical domain. In *International Conference on Information and Knowledge Management*.

Walk, S., Singer, P., Strohmaier, M., Tudorache, T., Musen, M. A., and Noy, N. F. (2014b). Discovering beaten paths in collaborative ontology-engineering projects using markov chains. *Journal of Biomedical Informatics*, 51:254–271.

Wang, C. and Huberman, B. A. (2012). How random are online social interactions? *Scientific Reports*, 2.

Wang, H., Tudorache, T., Dou, D., Noy, N. F., and Musen, M. A. (2013). Analysis of user editing patterns in ontology development projects. In *On the Move to Meaningful Internet Systems: OTM Conferences*.

Weakliem, D. L. (1999). A critique of the bayesian information criterion for model selection. *Sociological Methods & Research*, 27(3):359–397.

West, R. and Leskovec, J. (2012a). Automatic versus human navigation in information networks. In *International Conference on Web and Social Media*.

West, R. and Leskovec, J. (2012b). Human wayfinding in information networks. In *International Conference on World Wide Web*.

West, R., Pineau, J., and Precup, D. (2009). Wikispeedia: An online game for inferring semantic distances between concepts. In *International Joint Conference on Artificial Intelligence*.

White, R. W. and Drucker, S. M. (2007). Investigating behavioral variability in web search. In *International Conference on World Wide Web*.

White, R. W. and Huang, J. (2010). Assessing the scenic route: measuring the value of search trails in web logs. In *Conference on Research and Development in Information Retrieval*.

Whittaker, J. A., Thomason, M., et al. (1994). A markov chain model for statistical software testing. *IEEE Transactions on Software Engineering*, 20(10):812–824.

Yang, J., McAuley, J., Leskovec, J., LePendu, P., and Shah, N. (2014). Finding progression stages in time-evolving event sequences. In *International Conference on World Wide Web*.

Yeh, E., Ramage, D., Manning, C. D., Agirre, E., and Soroa, A. (2009). Wikiwalk: Random walks on wikipedia for semantic relatedness. In *Workshop on Graph-Based Methods for Natural Language Processing*.

Zhang, Z., Gentile, A., and Ciravegna, F. (2012). Recent advances in methods of lexical semantic relatedness-a survey. *Natural Language Engineering*, 1(1):1–69.

Zukerman, I., Albrecht, D. W., and Nicholson, A. E. (1999). Predicting users' requests on the www. In *International Conference on User Modeling*.

A. Complete List of Own Publications

A.1. Journal Articles

- [Singer et al., 2014c] Singer, P., Helic, D., Taraghi, B., and Strohmaier, M. (2014c). Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PloS One*, 9(7):e102070
- [Walk et al., 2014b] Walk, S., Singer, P., Strohmaier, M., Tudorache, T., Musen, M. A., and Noy, N. F. (2014b). Discovering beaten paths in collaborative ontology-engineering projects using markov chains. *Journal of Biomedical Informatics*, 51:254–271
- [Wagner et al., 2014c] Wagner, C., Singer, P., Strohmaier, M., and Huberman, B. (2014c). Semantic stability and implicit consensus in social tagging systems. *IEEE Transactions on Computational Social Systems*, 1(1):108–120
- [Singer et al., 2013a] Singer, P., Niebler, T., Strohmaier, M., and Hotho, A. (2013a). Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70

A.2. Pre-Prints

- [Singer et al., 2014b] Singer, P., Helic, D., Hotho, A., and Strohmaier, M. (2014b). Hyptrails: A bayesian approach for comparing hypotheses about human trails. *arXiv:1411.2844 [cs.SI]*

- [Wagner et al., 2014a] Wagner, C., Singer, P., and Strohmaier, M. (2014a). The nature and evolution of online food preferences. *EPJ Data Science*. accepted/to be published

A.3. Conference Proceedings

- [Walk et al., 2014a] Walk, S., Singer, P., and Strohmaier, M. (2014a). Sequential action patterns in collaborative ontology-engineering projects: A case-study in the biomedical domain. In *International Conference on Information and Knowledge Management*

- [Wagner et al., 2014d] Wagner, C., Singer, P., Strohmaier, M., and Huberman, B. A. (2014d). Semantic stability in social tagging streams. In *International Conference on World Wide Web*

- [Singer et al., 2014a] Singer, P., Flöck, F., Meinhart, C., Zeitfogel, E., and Strohmaier, M. (2014a). Evolution of reddit: From the front page of the internet to a self-referential community? In *International Conference on World Wide Web Companion*

- [Wagner et al., 2014b] Wagner, C., Singer, P., and Strohmaier, M. (2014b). Spatial and temporal patterns of online food preferences. In *International Conference on World Wide Web Companion*

- [Doerfel et al., 2014] Doerfel, S., Zoller, D., Singer, P., Niebler, T., Hotho, A., and Strohmaier, M. (2014). How social is social tagging? In *International Conference on World Wide Web Companion*

- [Singer, 2014] Singer, P. (2014). Understanding, leveraging and improving human navigation on the web. In *International Conference on World Wide Web Companion*

- [Singer et al., 2013b] Singer, P., Niebler, T., Strohmaier, M., and Hotho, A. (2013b). Computing semantic relatedness from human navigational paths: A case study on wikipedia. *International Journal on Semantic Web and Information Systems*, 9(4):41–70

- [Posch et al., 2013] Posch, L., Wagner, C., Singer, P., and Strohmaier, M. (2013). Meaning as collective use: Predicting hashtag semantics on twitter. In *International Conference on World Wide Web Companion*
- [Wagner et al., 2013] Wagner, C., Singer, P., Posch, L., and Strohmaier, M. (2013). The wisdom of the audience: An empirical study of social semantics in twitter streams. In *European Semantic Web Conference*
- [Niebler et al., 2013] Niebler, T., Singer, P., Benz, D., Körner, C., Strohmaier, M., and Hotho, A. (2013). How tagging pragmatics influence tag sense discovery in social annotation systems. In *European Conference on Information Retrieval*
- [Trattner et al., 2012] Trattner, C., Singer, P., Helic, D., and Strohmaier, M. (2012). Exploring the differences and similarities between hierarchical decentralized search and human navigation in information networks. In *International Conference on Knowledge Management and Knowledge Technologies*
- [Singer et al., 2012] Singer, P., Wagner, C., and Strohmaier, M. (2012). Understanding co-evolution of social and content networks on twitter. In *Workshop on Making Sense of Microposts*
- [Schöfegger et al., 2012] Schöfegger, K., Körner, C., Singer, P., and Granitzer, M. (2012). Learning user characteristics from social tagging behavior. In *Conference on Hypertext and Social Media*