Daniel Lamprecht

Using ontologies to model human navigation behavior in information networks

Master's Thesis

Graz University of Technology

Institute of Knowledge Management
Head: Univ.-Prof. Dipl.-Inf. Dr. Stefanie Lindstaedt

Supervisor: Dipl.-Ing. Dr.techn. Univ.-Doz. Markus Strohmaier

Graz, March 2013

Daniel Lamprecht

Modellierung von menschlichem Navigationsverhalten in Informationsnetzwerken unter Verwendung von Ontologien

Masterarbeit

Technische Universität Graz

Institut für Wissensmanagement
Vorstand: Univ.-Prof. Dipl.-Inf. Dr. Stefanie Lindstaedt

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

Graz, März 2013

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.

Graz, ____________________ ________________________

Date Signature

Eidesstattliche Erklärung[1]

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 entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am ____________________ ________________________

Datum Unterschrift

[1] Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

Abstract

In the evaluation of information systems, simulating the behavior of different user groups is a useful activity to understand the implication of design decisions. This thesis presents ontology-based Decentralized Search (OBDS), a novel method to model the navigation behavior of users equipped with different types of background knowledge. Ontology-based Decentralized Search combines decentralized search, an established method for navigation in social networks, and ontologies to model navigation behavior in information networks. OBDS uses ontologies as an explicit representation of background knowledge to inform the navigation process and guide it towards navigation targets. By using different ontologies, users equipped with different types of background knowledge can be represented. This thesis demonstrates the method using four ontologies from the biomedical domain and their associated Wikipedia articles. The obtained simulation results are compared with random walks, randomly generated ontologies and optimal solutions as base lines. To further verify the usefulness of the results, this thesis juxtaposes the simulation results with a user study. In conclusion, the findings of this thesis supports that Ontology-based decentralized search produces click paths similar to those originating from human navigators. These results indicate that the method can be used to model human navigation behavior in systems that are based on information networks (such as Wikipedia).

Kurzfassung

Die Simulation des Verhaltens unterschiedlicher Benutzergruppen stellt ein hilfreiches Werkzeug in der Evaluierung von Informationssystemen dar, und hilft dabei, die Auswirkungen von Designadaptionen besser zu verstehen. Diese Masterarbeit stellt Ontologiebasierte Dezentrale Suche (OBDS), einen neuen Ansatz zur Simulation von Benutzerverhalten vor. OBDS kombiniert Dezentrale Suche, eine etablierte Navigationsmethode in sozialen Netzwerken, und Ontologien zur Modellierung von Navigationsverhalten in Informationsnetzwerken. Die Methode verwendet Ontologien als eine explizite Repräsentation von Hintergrundwissen um die Navigation zu steuern und zum Ziel zu leiten. Durch die Verwendung von unterschiedlichen Ontologien können Benutzer mit verschiedene Arten von Hintergrundwissen repräsentiert werden. Im Anschluss an die Vorstellung der Methode wird in dieser Arbeit Ontologiebasierte Dezentrale Suche anhand des Beispiels von vier Ontologien aus der biomedizinischen Domäne und deren zugehörigen Wikipediaartikeln demonstriert. Die resultierenden Simulationsergebnisse werden mit Irrfahrten (Random-Walks), zufällig erstellten Ontologien und der jeweils bestmöglichen Lösung verglichen. Um den Simulationsanspruch zu untermauern, werden in einer Benutzerstudie die Ergebnisse der Simulationen mit menschlichen Klickdaten verglichen. Die Resultate dieser Masterarbeit zeigen, dass Ontologiebasierte Dezentrale Suche Klickpfade produziert, die denen von menschlichen Benutzern sehr ähnlich sind. Diese Ergebnisse legen nahe, dass OBDS für die Modellierung menschlichen Navigationsverhaltens in Informationssystemen (wie beispielsweise Wikipedia) eine geeignete Methode ist.

Acknowledgements

First of all, I would like to thank my advisor Dr. Markus Strohmaier for supervising me in the process of writing this thesis. His vision was what drove this research, and all of this would not have been possible without his constant feedback, motivation, and encouraging discussions.

I would further like to thank Prof. Mark Musen for making it possible for me to spend three unique months at Stanford University, California and to participate in cutting-edge research, including the preparation of a journal paper.

To the staff at the Stanford Center for Biomedical Informatics Research, thank you for your feedback, inspiring discussion, encouragement, creativity and patience. I would like to especially thank Natasha F. Noy, Tania Tudorache and Csongor Nyulas for their support.

The research visit to Stanford University was made possible with the generous grant of a Marshall Plan scholarship by the Austrian Marshall Foundation, for which I am very grateful.

I would like to express my thanks to all of my colleagues and friends at the Knowledge Management Institut of Graz University of Technology, who provided me with constant feedback and motivation and created a very warm and encouraging atmosphere in our research group. Thank you also to Dr. Denis Helic for fruitful discussions and insights into decentralized search.

Finally, I want to thank my parents for alway encouraging and allowing me to realize my potentials and talents. Thank you for all your support and love.

Daniel Lamprecht

Graz, March 2013

Contents

List of Figures

List of Tables

1. Introduction

1.1. Motivation

With the advent of the World Wide Web, navigating information networks has become an important factor in everyday human life. Being able to effectively search and navigate on the Web is now as important as using a car or a telephone. Due to its vast number of web pages, tools such as search engines are often needed to aid in exploring the Web. Users generally only know a small portion of the Web, and as such navigation becomes an important factor in cases when search engines cannot help with tasks such as exploration or finding the concept on the tip of one's tongue.

The Web has become increasingly popular with humans of all ages and backgrounds. As a result, one of the key challenges of building information systems is the need to develop interfaces suited to a range of different types of users. Different types of users, such as novices, experts, generalists or specialists will, in general, display considerably different knowledge about a given domain. This specific knowledge in turn influences their interactions with an information system. Gaining insight into human navigation behavior supports the construction of easy to use software and information systems that are ready to accommodate a broad range of user types.

This master's thesis investigates ways of modeling navigational behavior of human users in information networks. Humans navigating an information network (such as Wikipedia) are generally not familiar with the global network structure but navigate based on assumptions, intentions and locally available information only. Experiments by Stanley Milgram and others [TMTM69] [Mil67] have shown that humans are very effective at finding short paths based on local information in offline as well as online *social networks*.

One of the motivating factors driving this thesis was the curiosity gain insight into the details of similar Wikipedia navigation sessions. Figure 1.1 is an excerpt from the popular web comic XKCD[1] addressing this topic.

[1] http://www.xkcd.com

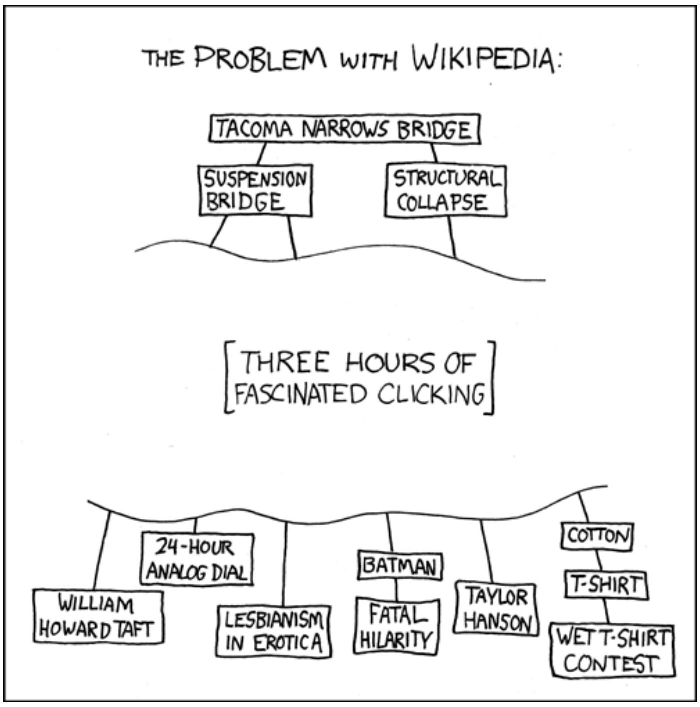


Figure 1.1.: **The problem with Wikipedia** (CC BY-NC 2.5 from `http://xkcd.com/214/`)

This thesis presents a novel method (ontology-based decentralized search) for simulating human navigational click behavior in *information networks* and examine its suitability to model actual human navigation behavior. The method, called Ontology-based decentralized search (OBDS), builds on decentralized search, a well-established navigation method in social networks, which is based on local information only. Decentralized search is applied to navigation in information networks to model the behavior of users with varying levels of domain knowledge and produce simulated click data. OBDS uses decentralized search with existing well-established ontologies as background knowledge to inform the search process and point it towards the direction of the target.

This method is new in that it uses an explicit representation of background knowledge in the form of an ontology. Previous research in psychology suggests that humans store concepts in their minds hierarchically [FE07], similar to ontologies. Ontology-based Decentralized Search models different groups of users by using different ontologies as background knowledge.

1.2. Research Questions and Contributions

The following lists the three main research questions that drove this thesis, as well as summarized answers to each of them. All of the research questions are addressed in Chapters 5 and 6 and discussed in Chapter 7.1.

Research Question 1 **Can existing ontologies contribute useful information to navigation in information networks? If yes, how do existing ontologies perform in comparison to randomly generated ontologies and random walks?**

The results show that ontologies can indeed inform navigation in information networks. Their performance depends on the specific domain and the quality of the mappings to the information network. In the evaluations, OBDS works well for the data set containing diseases and less well for the the data set containing genes and gene products.

Research Question 2 **Does ontology-based decentralized search (OBDS) produce valid results, i.e., are the simulated navigation paths similar to those produced by human navigation?**

This question is addressed by comparing the resulting navigation paths of the simulations to a user study. The results show that that the click paths produce by OBDS perform well above pure random walks. For one of the data sets, OBDS perform substantially better than randomly generated ontologies as well.

Research Question 3 **When using OBDS, what ontology is bested suited to produce human-like navigation results?**

From the results, ICD-10 and MeSH seem to be best suited to be used as a replacement for human behavior when navigating in an information network (all used ontologies are described in detail in the following sections). However, the overall differences between the ontologies are not very strong, and it is subject of ongoing research to further identify differences in the performances of different ontologies.

To demonstrate Ontology-based Decentralized Search, this thesis makes use of the information network formed by a set of Wikipedia articles from the biomedical domain and the connections (hyperlinks) between them. The research presented in this thesis shows that several different ontologies from the biomedical domain can be used as background knowledge to inform navigation simulations, much as humans can use their acquired knowledge for navigation on the encyclopedia.

The main contribution of this thesis is the demonstration of the general suitability of existing real-world ontologies to inform decentralized search on information networks such as Wikipedia. By comparing the navigational paths generated by the simulation with several baseline approaches and with data obtained from a user study, the outcome shows that the method yields results similar to those produced by actual human users. These results suggest that OBDS can indeed be used to simulate human navigational behavior in information networks. This might be useful for addressing issues arising in the development of systems that are based on networked information. These findings are relevant for new methods of applying ontologies and for modeling navigation in information networks using ontologies as background knowledge.

1.3. Thesis Outline

This thesis consists of eight chapters. The introductory chapter is followed by Chapter 2, in which the context of previous related work to this thesis is discussed in the context of navigation models and ontologies. Chapter 3 presents Decentralized Search and its extension Ontology-based Search, the main algorithm of this thesis. Chapter 4 describes the materials and methods and discusses the general setup, the used Ontologies, the Wikipedia articles, the navigation scenarios and the setup of the user study. Chapter 5 details the results of the case study of applying OBDS to a dataset of Wikipedia articles and ontologies in the biomedical domain. Chapter 6 further expands on the results and compares them with the findings of the user study. Chapter 7 discusses the results. Finally, Chapter 8 concludes the thesis and proposes future work in relation to the research discussed in this thesis. The attached Appendix provides supplementary material used in the course of this thesis (Appendix A B C).

Parts of this thesis are in the process of being submitted to a journal for publication.

2. Related Work

In the context of this thesis, the areas of Navigation Models and Ontologies are of particular importance. This chapter gives an overview of the related work in these fields. Navigation models are first discussed in the domain of social networks, followed by navigation models in information networks. The second part of this chapter introduces ontologies as instruments for knowledge representation and reviews the developments in the fields of ontology languages, ontologies in the biomedical domain and ontologies used for navigational purposes.

2.1. Navigation Models

This thesis studies navigation in the field information networks, more specifically using the example of Wikipedia. As much of the theory of network navigation is based on social networks, this chapter starts with an overview over this field first, followed by a section on navigation in information networks.

2.1.1. Navigation in social networks

This thesis particularly addresses navigation in social networks via decentralized search algorithms. Fundamentally, decentralized search describes a way of solving a pathfinding problem in a social network. Starting from an arbitrary start node (i.e., a person) within the network, the objective of decentralized search is to find a way to a given target node. The algorithm, however, does not possess global knowledge of the network and can therefore only take local decisions. The term "decentralized" stems from the fact that the search proceeds by forwarding the search problem from one node to another, which, in a social network, involves a different person taking the decisions at every node.

The idea of decentralized search, as used in the navigation simulations of this thesis, was made popular by Stanley Milgram's widely discussed small-world experiment

[TMTM69] [Mil67] in the 1960s. In the experiment, participants in Boston and Nebraska received a letter containing information about a specific target person. This target person was a stock broker in Boston, Massachusetts. The participants were then asked to forward the letter to one of their friends, which, in the experiment, were defined as an acquaintance known on a first-name basis. The objective of forwarding the letter was to bring it to someone closer to the target person, who would then forward it to another person believed to be yet closer to the Boston stock broker.

The results showed a median chain length of six intermediates for successful chains of letters and coined the term of "six degrees of separation". Letters from both participants in both Nebraska and Boston reached the target person via only very few intermediates, demonstrating that geographic distance showed only little influence. Shortly before the target, multiple letters were forwarded through a small number of intermediaries acting as channels towards the target person. This is depicted in Figure 2.1. By taking only the limited knowledge of each participant into account at each step, the search effectively constituted a form of decentralized search. The result illustrated the so-called *small world phenomenon*, as it seemed possible to connect two arbitrary persons across the United States through a very small number of hops.

The experiment has been criticized for a number of methodological flaws such as selection bias, as participants were solicited through newspaper advertisements, and a bias towards shorter successful chains, as longer chains were more difficultly obtained with participation relying on volunteers [Sch09] [Eri79]. However, subsequent studies successfully repeated the experiment. Dodds et al. [DMW03] were able to repeat the experiment using e-mail in 2002, connecting individuals across continents with similar results.

In 2011, Backstrom et al. [BBR+11] reported the average distance between pairs of worldwide Facebook users to be between four and five hops, and the average distance between Facebook users within a single country as around three. However, unlike in the original Milgram experiment, the study of the Facebook graph did not rely on user participation but instead calculated the shortest distances between all pairs of users in the graph (i.e., always chose the optimal person to forward the search problem to).

In 1998, Watts and Strogatz [WS98] formally characterized networks exhibiting small-world characteristics as having a high clustering coefficient and a low characteristic path length (i.e., a low average path length between pairs of nodes in the graph). In contrast to pure random graphs, real-world networks exhibiting the small-world phenomenon showed these two characteristics for a range of examples. Graphs created by randomly connecting nodes on the other hand typically showed a low clustering coefficient and a high characteristic path length. The idea of these properties was to

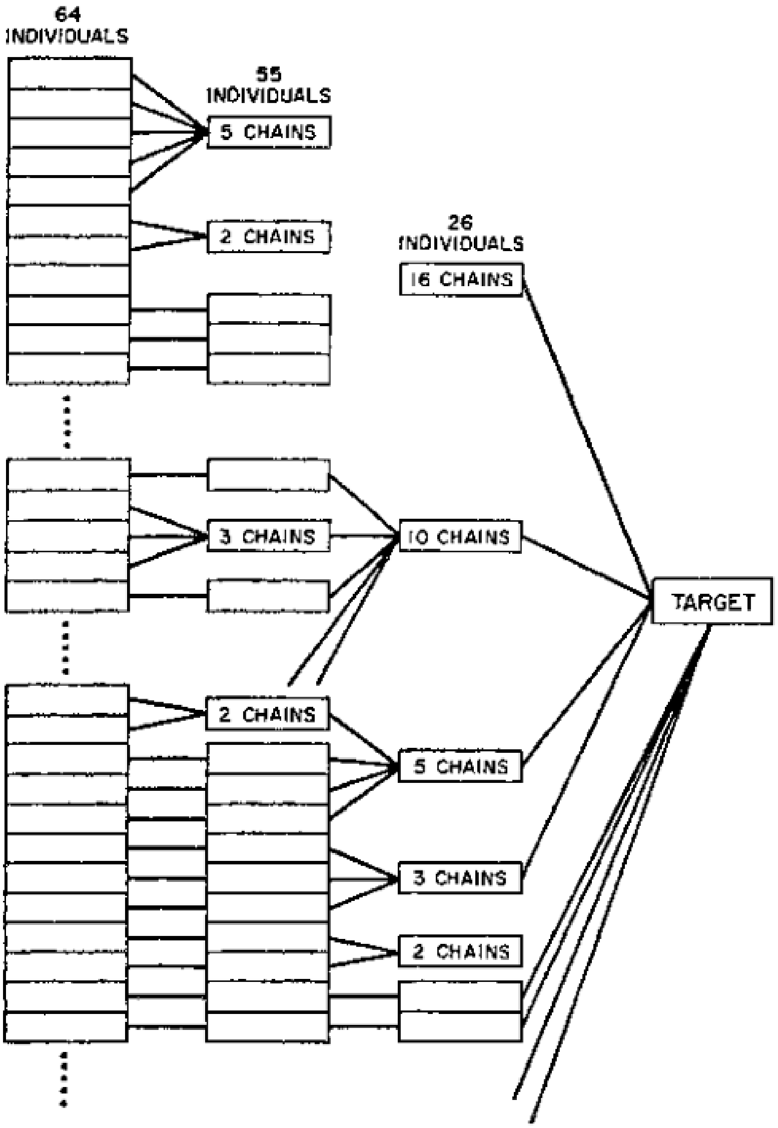


Figure 2.1.: **Common channels before the target in the Milgram experiment** The figure shows the successful letter chains converged to the target person through relatively few channels, i.e., intermediaries who forwarded multiple letters. This image originally appeared in [TMTM69, p. 439].

model the high clustering in social networks together with the typically short paths connecting any two individuals, which were modeled by a few random connections. Watts and Strogatz introduced a process of automatically creating networks for which these properties occurred by starting from a regular ring lattice graph and rewiring a small fraction (0.5% - 5%) of the edges to randomly chosen nodes. The resulting graph was shown to fulfill the postulated requirements for a small world network. Furthermore, Watts and Strogatz [WS98] demonstrated the actual existence of this type of small-world networks in a film actor collaboration network, the power grid of the western United States and the neural network of Caenorhabditis elegans, a small roundworm.

In 2000, Jon Kleinberg proved that for the type of small-world networks proposed by Watts and Strogatz [Kle00], no effective decentralized search algorithm could exist that always found a path connecting two arbitrary two nodes in subpolynomial time. However, Kleinberg presented a more specialized version of the model that, instead of rewiring uniformly at random, chose the random node to attach to following an inverse power distribution based on the distance from that node. This meant that the probability of rewiring to a distance of 1 was 0.5, to a distance of 2 it was 0.25 and so forth. By rewiring according to a distribution, the network model allowed for more effective decentralized search, as it permitted a decentralized search algorithm to reduce the distance to the target by an order of magnitude every few steps, independently of the total distance. Kleinberg consequently proved that a decentralized algorithm capable of finding short paths existed for this and only this class of networks.

One year later, Kleinberg extended his model of decentralized search to include hierarchies [Kle01], where the term *hierarchy* denotes a tree that includes all network nodes. He showed that when the network nodes were embedded as the leaf nodes of a hierarchy and links in a network were formed proportional to distances in this hierarchy, the resulting network was also efficiently searchable by making use of the hierarchy as a background knowledge. To form an effectively searchable graph, nodes were connected with a probability proportional to their distance in the tree, i.e., the height of their closest common ancestor. Provided the hierarchy information, the search could then proceed to the target effectively. This thesis makes use of ontologies as this type of background knowledge.

Miao et al. [MTC+12] have studied decentralized search in collaboration networks. Collaboration networks differ from information or social networks in that the information flow in them is driven by tasks. This means that the edges in the network are formed by collaboration on tasks. In their study, the tasks were software bugs. Developers who were assigned a bug they could not eliminate themselves forwarded it to another developer who they believed could handle it. By establishing several forwards in a row, this of work flow consisted a type of decentralized search, as all decisions about the next hop were taken independently by multiple participants. Miao et al. studied this in the context of the development networks of the Eclipse and Netbeans software, as well as in an IT service management system and developed algorithms to simulate both the network creation and the information flow.

Adamic and Adar [AA04] studied decentralized search in the e-mail network of HP labs. They constructed a network of 436 company employees from the e-mail communication log and examined three different decentralized search strategies: a) best connected, b) according to organizational hierarchy and c) physical proximity in the office (cubicle distance). While strategy a) proved unsuccessful, strategies b) and c) were shown to be a good approach to navigating the e-mail network.

Decentralized search is also used in peer-to-peer file sharing protocols such as Gnutella or KaZaA. The Gnutella network displayed small-world characteristics in 2003 [LZHH03], with a low characteristic path length and a high cluster coefficient.

2.1.2. Navigation in information networks

In this thesis, decentralized search, a navigation model originally developed for social networks, is applied to information networks. The procedures and methods of this are described in detail in section 3. This section provides an overview of previous work related to navigating information networks.

One of the most prominent related model to search in information networks is information foraging [Pir07]. Information foraging is based on foraging theory in biology. In order to survive, animals have adopted methods which maximize the energy gained from food sources. In the theory of information foraging, search in information networks is not guided by background knowledge but by information scent. Information is assumed to be available in patches, just as food is often available in larger quantities (e.g., a bush providing berries). Search in information networks is viewed as being guided by information scent, with each article and link emanating a distinct scent, which is dependent on the target of the search. For instance, when searching for information on penguins, a link leading to an article about Antarctica would provide more scent than a link leading to an article about the Sahara desert.

In this thesis, information networks are studied using the example of Wikipedia. However, real navigation paths from Wikipedia are very hard to obtain, as the goals of users navigating are often hidden and not explicitly visible and logs of click trails are difficult to obtain. Furthemore, recent research [GCFG10] has shown that when visiting a Wikipedia page, users have a mere 30 - 40% chance of following a link on that page. Users are hence more likely to jump to some other page directly. Jumping to another page is referred to as *teleporting*, e.g., by using the search function or typing in another address manually. In general, users on the web are estimated to follow a hyperlink on the current page in about 60 - 70% of their clicks [GCFG10]. In the original page rank formula and calculations, teleportation was assumed to occur in 15% of all clicks [PBMW98]. With $60 - 70\%$, the fraction of teleports is hence significantly higher on Wikipedia than on general web sites. This might be due to the fact that users visit Wikipedia to satisfy specific information demands rather than to browse articles. However, there exist valid reasons to navigate Wikipedia, which will be detailed in the description of the navigation scenarios in Chapter 4.4.

As a consequence of the high teleportation factor, navigation paths are often short in nature. Due to this and the difficulty of obtaining Wikipedia navigation paths, wiki games have been a popular replacement for Wikipedia navigation paths in recent research. Wiki games, such as Wikispeedia[1], WikipediaMaze[2] or Wiki Game[3] allow users to play games on the network formed by the Wikipedia articles and links. In their most simple versions, games consist of finding a path between two given articles, e.g., from Dik-dik (a small antelope) to Albert Einstein. An example for this is shown in figure 2.2. Click trails from wiki games have enabled researchers to gain insight into navigational behavior on Wikipedia. In 2009, West et al. [WPP09] used wiki game data to infer semantic distances between concepts by studying game click paths. In 2012, West and Leskovec [WL12b] found that in wiki games, players tend to navigate to hubs (articles with a large number of outlinks) first, and subsequently home in on targets node.

In the research group at the Institute of Knowledge Management (where this thesis was written), decentralized search with non-ontological background knowledge has already been studied in different contexts. The work presented in the course of this thesis builds on a navigation simulation framework that permits the simulation of decentralized search. The framework is currently being extended in the course of three master's theses and is based on the SNAP [sna12] framework developed at Stanford University.

In 2011, research at the Institute of Knowledge Management compared the navigability of different tag hierarchy generation algorithms on data from Bibsonomy, CiteULike, Delicious, Flickr and LastFm [HS11]. The paper evaluated the suitability of tag hierarchies for navigation on tagging networks and proposed a novel tag hierarchy generation algorithm.

In 2012, Strohmaier, Helic et al. compared different folksonomy induction algorithms through decentralized search [SHB+12] . They showed that, based on evaluation

[1] www.wikispeedia.net

[2] www.wikipediamaze.com

[3] www.thewikigame.com

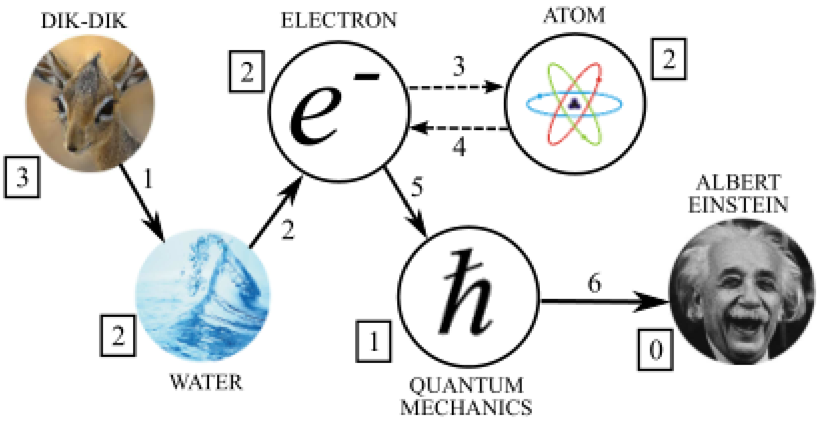


Figure 2.2.: **Example for a wiki game.** Wiki games allow users to play games on the network formed by the Wikipedia articles and links. In their most simple versions, games consist of finding a path between two given articles, e.g., from Dik-dik (a small antelope) to Albert Einstein. Figure reprinted from [WL12b]

through navigation, clustering algorithms developed for social tagging systems performed better than standard hierarchical clustering algorithms.

Helic et al. applied decentralized search to broad and narrow folksonomies on data from Mendeley [HKG+12] and found broad folksonomies better suited to supporting navigation.

Trattner et al [TSHS12] compared decentralized search and human navigation behavior in information networks and showed that the simulation of decentralized search yielded very similar results to actual human navigation data on Wikipedia. In their work, Trattner et al. investigated different types of hierarchies as background knowledge and found that decentralized search based on a hierarchy generated from network features such as in- and outdegree simulated human navigation better than comparable hierarchies generated from external knowledge.

In ongoing research, Helic, Strohmaier et al. are studying the influence of stochasticity and different methods of selecting the next hop in decentralized search [HSGS13].

The previous work did not tap into existing ontologies as background knowledge, but used other approaches (such as automated methods) for this purpose. This thesis goes beyond previous research by extending the simulation framework with ontologies and by applying Ontology-based Decentralized Search to the case of Wikipedia and for concrete ontologies for the biomedical domain.

2.2. Ontologies

The second part of this chapter on related work is concerned with ontologies. The term ontology originally stems from philosophy, where it denotes the study of ”what is, of the kinds and structures of objects, properties, events, processes, and relations in every area of reality” [Smi08]. In philosophy, the word ontology is uncountable and is used to represent an entire branch of philosophy.

In computer science on the contrary, ontologies are countable and an ontology is ”a formal, explicit specification of a shared conceptualization” [SBF98]. This definition states that an ontology should be machine-readable (formal), explicitly specified, commonly agreed upon by its user base (shared) and a conceptualization. A conceptualization is ”an abstract, simplified view of the world that I wish to represent for some purpose” [GN87]. Strictly speaking, an ontology only defines the schema, and an ontology with instances of concepts forms a knowledge base (though this is rarely rigorously separated) [NM+01]. In this thesis, the ensemble of the scheme and instances are referred to as an ontology.

Ontologies met the desire to formalize a common view of the world to be used in artificial intelligence and knowledge systems. Other reasons for their development included the reuse of domain knowledge, the separation of domain knowledge and operational knowledge, the easier analysis of domain knowledge [NM+01] and the unification of different views of the world [UG+96]. An ontology consists of a set of ”concepts, their definitions and their inter-relationships” [UG+96].

In 2001, Tim Berners-Lee proposed the semantic web as a new version of the Web, accessible to and improved by automated agents [BLHL+01]. The semantic web relies on the so-called *semantic web pyramid of languages* in order to exchange machine-processable data. It uses XML for data exchange and to ”provide a serialized syntax for tree structures” [FVHH+01]. Building on XML, the Resource Descriptor Framework (RDF) is used for assertions [DFVH03] . RDF is ”a [...] W3C recommendation designed to standardize the definition and use of meta-data descriptions of web-based resources” [DFVH03, p. 12] and defines a simple, machine-readable data format [FVHH+01]. RDF consists of (subject, predicate, object) triples. The sentence ”Daniel reads his thesis”, for instance, would be represented as ”(Daniel, read, thesis”) in the logic of RDF. These triples, when added together, make up the so-called RDF graph. RDF may be serialized to XML or a variety of other data formats. RDF Schema (RDFS) extends RDF with a basic system that supports the creation of simple ontologies [DFVH03].

The desire to remedy some of the shortcomings of RDFS provided the foundations for the development of the Ontology Inference Layer (OIL), a more capable ontology language for the web. OIL was funded by the European Union, provided ”the full power of an expressive description logic” [DFVH03, p. 18] and consists of several different complexity levels. In its most simple form (Core OIL), the language is almost the same as RDFS [DFVH03].

OIL was subsequently unified with the DARPA Agent Markup Language (DAML), a similar-purpose language for the description of ontologies developed at the United States Defense Advanced Research Projects Agency. The unification of the two languages was called DAML+OIL. This markup language was ”more tightly integrated with RDFS” [DFVH03, p. 26]. DAML+OIL is only serializable to RDFS and defined a more formal semantic, enabling better automated reasoning [Zäc03].

In 2001, the W3C started working on a new ontology language named *Web Ontology Language* and abbreviated as OWL. The objective of OWL was none less than to become the standard ontology language used on the Web. OWL built on DAML+OIL and RDF [Stu09] and became a formal W3C recommendation in 2004 [w3c04]. OWL is specified in three variants (OWL Lite, OWL DL and OWL Full), where OWL Lite is a subset of OWL DL, which in turn is a subset of OWL Full [w3c04].

While not strictly a superset of RDF and RDFS, OWL makes large use of them [AH09]. The most important components of an OWL ontology are classes and properties[4]. Classes define the basic concepts in an ontology. Classes may have properties specifying attributes and relations to other classes [AH09]. Since 2004, OWL has become a widely-used standard for ontologies. All four ontologies studied in this thesis are available in OWL.

Besides ontologies, similar knowledge systems are controlled vocabularies, taxonomies and thesauri. Rubin et al. define a controlled vocabulary (which is also called a terminology) as consisting of a list of concepts, their descriptions and lexical terms [RSN08]. The entries in a controlled vocabulary should be unambiguous, and synonyms should be aggregated. A taxonomy is a hierarchically organized controlled vocabulary [Pid03]. A thesaurus is a collection of controlled vocabulary terms which ”uses associative relationships in addition to parent-child relationships” [Pid03]. In the literature, however, definitions for ontologies and related concepts often contradict each other or are not used in the strict sense of the word (e.g., controlled vocabularies are often denoted as ontologies).

Uschold et al. define three application areas for ontologies: Communication, Inter-Operability and Systems engineering as follows [UG^{+}96, p. 7-13]:

[4] http://www.w3.org/TR/owl-guide/

1. *Communication* Ontologies provide a unified view of a shared domain and can help to settle different viewpoints. They also define unambiguous definitions of concepts and keep track of the connections between them.
2. *Inter-Operability* Ontologies can help with exchanging different data formats and act as Inter-Lingua, providing an intermediate view of the world to which other views can be translated.
3. *Systems engineering* Ontologies can serve as the basis for specifications in engineering and simplify automated checking of the implementation of specifications and facilitate reuse.

Evaluating ontologies and estimating the suitability of an ontology for a specific task remain open problems in computer science. Different ontologies may be best suited for different purposes, and the assessment of an ontology as ”good” or ”fit” may not hold in general. Often, ontology evaluation cannot be done automatically and human support is required.

In 2005, Brank et al. [BGM] surveyed ontology evaluation techniques and identified four main methods:

1. *Comparison to a gold standard* by lexical comparison to an ontology considered a good representation
2. *Data-driven evaluation* by measuring the fit of an ontology to data, e.g., a set of documents
3. *Manual assessment* by humans, who analyze the ontology with regard to requirements and standards.
4. *Task-based evaluation* by measuring the fitness of an ontology for a given task or application

2.2.1. Ontologies in the biomedical domain

In the biomedical domain, ontologies have been adapted more frequently than in other disciplines [NT08]. Biomedical ontologies play an important role in biomedical research [B+08] and are used for a range of purposes.

Aggregated by their function, ontologies in this domain can be grouped in six classes [RSN08]:

1. *Search and query of biomedical data* Ontologies such as the GeneOntology, Medical Subject Headings or the NCI Thesaurus act as controlled vocabularies to facilitate information retrieval.

2. *Data exchange* Ontologies can specify data exchange formats between applications. Examples for this are MAGE-ML and MAGE-OM, which define information models for microarrays.
3. *Information integration* Ontologies can simplify relating multiple databases by providing ontological descriptions for their content.
4. *Natural language processing* NLP methods in combination with ontologies facilitate information extraction.
5. *Representation of encyclopedic knowledge* Encyclopedic information can be combined with ontological structure to enable automated access.
6. *Computer reasoning* Knowledge systems may permit automated reasoning over their contents and derive new facts.

Bodenreider et al [B+08] functionally surveyed ontologies in the biomedical domain according to three classes:

Knowledge Management Ontologies in the biomedical domain provide a controlled vocabulary. Ontologies such as Medical Subject Headings (MeSH) and the Gene Ontology (GO) are applied to annotate data, thus enabling better access to medical documents. The International Classification of Diseases (ICD) and SNOMED CT are used to index clinical documents and make use of codes for diseases. The Unified Medical Languages System (UMLS) provides mappings between multiple ontologies of the same domain.

Data Integration, Exchange and Semantic Interoperability Ontologies support data integration by either providing a common data format or mappings. UMLS, SNOMED CT and Logical Observation Identifiers Names and Codes (LOINC) are used for these purposes.

Decision Support and Reasoning Ontologies help in research to select groups of patients in medical trials and to identify common characteristics according to different treatments. These methods are often based on ICD. Ontologies also play a supporting role in clinical decision support by providing a controlled vocabulary and a formal, machine-accessible knowledge base. Furthermore, ontologies back up natural language processing tasks and knowledge discovery.

BioPortal [WNS+11], developed by the National Center for Biomedical Ontology (NCBO) of the United States, is a web portal which provides access to a range of ontologies. BioPortal permits access to its ontologies in a number of data formats and supports searching in ontologies as well as API access.

2.2.2. Ontologies and Navigation

Ontologies have been used in previous research to facilitate navigation in digital libraries. For example, Papazoglou and Hoppenbrouwers [PH99] have used ontologies to retrieve related works when searching digital libraries.

The research of Rajapakse et al. [RKA+08] shows efforts to navigate the digitally available literature related to dengue fever. The dengue domain consists of heterogeneous data and is, in general, difficult to search. The authors created an ontology text-mining techniques and were able to simplified information-retrieval on the domain.

Villela Dantes et al. [VDMF10] have studied the ontology-guided insertion of links into web pages. In their work, they classified web pages according to an ontology and subsequently inserted links to related topics into web pages to facilitate navigation.

Mohanraj et al. [MCS+11] have examined self-adapting ontologies in the case of recommendation systems. Their objective was to predict the next step in a user's navigation. The first step of their approach consisted of a genetic algorithm (following the honey bee foraging behavior) and was used to find recommendation candidates for the user's next click. In a second step, they selected one recommendation among the candidates using an adaptive ontology, which they updated according to the user's previous navigation steps.

These research papers share the effort to use ontologies to aid navigation. The objective of this thesis lies in explaining and modeling user behavior by using ontologies as background knowledge. The ontologies are hence not used to guide human users but to simulate and possibly explain their behavior.

3. Decentralized Search

3.1. Introduction

Decentralized search is a method of solving a pathfinding problem in a network without a central control unit. Starting from an arbitrary start node within the network, the objective of decentralized search is to find a way to a given target node. The term *decentralized* stems from the fact that the search proceeds by forwarding the search problem from one node to the next, until the target is reached. In Stanley Milgram's small world experiment [TMTM69], decentralized search was established through humans forwarding letters to acquaintances in order to find a target person. Each human along the chain of letters acted independently of all others and thus made the search decentralized, i.e., acting without a central control unit involved in the decisions at every step. Further examples for decentralized search include bug forwarding in a developer network, where software bugs are assigned to a starting person, and then forwarded to other developers until it is fixed [MTC+12], or job recommendations in social networks [AA04].

In the theory of network navigability, Jon Kleinberg showed that networks that are formed according to a background hierarchy (i.e., a tree) are efficiently navigable [Kle01], provided the search agent has access to that background hierarchy during the search. Some sort of background knowledge about the network however, is a general necessity for efficient navigability when searching a network. In Stanley Milgram's small-world experiment, participants were provided with certain pieces of background information about the target person, such as geographic location and profession [Mil67] and were able to exploit these facts when forwarding letters.

According to Jon Kleinberg's work [Kle01], Decentralized Search uses a second network as the background knowledge. This background knowledge is called a *hierarchy* because of its tree structure. The actual navigation takes place on the network N and uses distance information from the background knowledge H, which contains all nodes from N but, in general, different connections between them. While the network N is a directed graph, the hierarchy H is undirected.

This thesis applies decentralized search, which was originally developed for social networks, and apply it to information networks. There exist substantial differences between navigation in these networks, as described in [HSGS13]: In a social network, navigation is executed by multiple agents that are a part of the network (search problems are forwarded to another person), and the search is hence genuinely *decentralized*. In an information network on the contrary, the navigation is performed by a single agent that is generally not a part of the network. Furthermore, the cost of consulting candidate nodes is expensive in social networks (meaning that the process of contacting social entities is time-consuming), while it is comparatively cheap to consult candidates (e.g., web sites) in an information network [HSGS13]. These changes are summarized in Table 3.1.

Nevertheless, Decentralized Search is still an interesting and scientifically rewarding method to apply to information networks: While the method becomes less decentralized in a certain sense, the study of Decentralized Search reveals intriguing aspects of information networks such as their navigability, an aspect often neglected in the presence of powerful search engines. Moreover, Decentralized Search can be applied to simulate user behavior in information networks, an aspect which makes it attractive to automatically examine the ramifications of network structure changes to navigation behavior.

In a social network, the decision of where to forward the problem to is generally based on the expected knowledge and capability of that particular next node (person). For the simulations in this thesis, it is assumed that all nodes shared a common background knowledge. This assumption made the algorithm less *decentralized* in a certain sense, because all the decisions are now made by one and the same entity (the simulation). Just like in the original decentralized search algorithm however, at each node the simulation could only access information about that particular node's local

	Social Networks	**Information Networks**
Agents per search	multiple agents	single agent
Type of routing	decentralized (with local knowledge)	centralized (with local knowledge)
Searcher	part of the network (endogenous)	not part of the network (exogenous)
Routing decisions	social intuitions	topical intuitions
Local knowledge	rich	limited
Consultation of candidates	costly	cheap

Table 3.1.: **Potential differences and commonalities between navigation in social and information networks** This table originally appeared in [HSGS13]

network neighborhood. The background knowledge represented additional knowledge about the network necessary to effectively find a short path to the target.

To provide an example: *When looking for an employee in a company for instance, the employees and their acquaintance relations could form the network, and the background knowledge could be represented by the organizational hierarchy, i.e. managers, team members and so forth. In this case, the restriction for the navigation would be that it could only be forwarded to acquainted employees. This reflects the reality of personal recommendations when searching for employment.*

The following section discusses design parameters of Decentralized Search, followed by a section on Hierarchical Decentralized Search, which is directly based on Jon Kleinberg's work of hierarchies and navigability. After that follows a description of Ontology-based Decentralized Search, the new algorithm presented by this thesis.

3.2. Design Parameters of Decentralized Search

This chapter gives an overview of the possible design choices in concrete implementations of the Decentralized Search algorithm and discusses their implications. Any version of Decentralized Search is bound to make a choice for each of these design parameters.

3.2.1. Background knowledge utilization

At every step, Decentralized Search queries the background knowledge graph to obtain the next node to proceed to.

Ranking The background knowledge is used to establish a ranking of potential successor nodes at each step. There exist several measures for this: The most straightforward measure is the calculation and ranking of the nodes by their geodesic distance to the target on the background knowledge. In case the background knowledge forms a tree, another possible measure is the height of the lowest common ancestor between the current node and the target node. This ranking represents the intuitions of the agent navigating on the network, provided by the background knowledge. In the Milgram experiment, this could for instance be the geographic distance to the target person.

Completeness A background knowledge is called *complete*, if it contains information about all the nodes present in the network. A *complete* background knowledge contains every node of the network but does not necessarily provide the correct information, as it generally contains different edges. In general, the background knowledge does not need to contain information about any node (in which case the search becomes a random walk) or may contain perfect knowledge (in which case the search follows the shortest path). In case of a target node not in the background knowledge, the search can either abort or proceed randomly.

3.2.2. Node selection

Given a ranking of nodes provided by the background knowledge, the next step for the Decentralized Search algorithm is the selection of the successor node. In addition to the ranking, the selection process may include further restrictions or additional information about the potential successor nodes.

Selection strategy Generally, nodes may be selected either deterministically or stochastically [HSGS13]:

- A *deterministic greedy* selection always opts for the top-ranked node as provided by the background knowledge.
- A *stochastic* node selection may be implemented in a variety of ways, such as
  - ϵ-greedy or variants thereof, such as ϵ-beginning or ϵ-decreasing [BBG12]
  - with probabilities proportional to the ranking, e.g., softmax [HSGS13]

The most simple node selection strategy (deterministically greedy) is to always follow the information provided by the background knowledge and pick the top-ranked node. However, the search strategy may also take several further aspects of the network into account. In general, Decentralized Search also has access to information about some properties of the potential successor nodes, such some measure of textual similarity to the current node or in- and outdegree. This may be applied to form a node-selection strategy which considers only nodes with certain properties, such as high degree or high textual similarity to the current node.

Search strategies may also combine several strategies, such as initially navigating to a well-ranked high-degree node first and subsequently selecting successor nodes based on textual similarity to the target node. [WL12a] lists several potential search strategies for the information network of Wikipedia.

Node revisitation Another crucial aspect is the revisitation of nodes. If network nodes can be visited an unlimited number of times, this may lead to looping behavior. In order to avoid this, implementations of Decentralized Search may restrict nodes to be visited only a predetermined number of times (e.g., a maximum of two visits) or only once.

Handling of multiple targets The case of Decentralized Search with multiple targets raises the question of how the navigation handles them. With an increasing number of targets, finding an optimal way of visiting multiple targets (based on the information provided by the background knowledge) becomes increasingly hard, as it constitutes an instance of a traveling salesman problem (TSP). There exist several possibilities, such as

- Determining the search order beforehand via the information provided by the background knowledge (i.e., solving the TSP or approximating it).
- Determining the next node to navigate to at every step, i.e., calculating the distances to all target nodes and subsequently taking one step towards the closest target.

3.2.3. Termination Condition

Depending on the search strategy, Decentralized Search may run until it has explored all nodes in the network, or loop indefinitely. As this may not always be the desired behavior, a termination condition is often useful. The most simple version is the termination after a maximum number of steps, if no target has been found. Another interesting aspect is the implementation of an *attrition rate* which assigns a certain probability of termination at every step (just as human participants had a chance of dropping out at every step in the Milgram experiment). The attrition rate can be modeled following a certain function, e.g., linear, quadratic or hyperbolic.

3.2.4. Backtracking

A further essential design parameter is backtracking during the navigation. Backtracking models the back button functionality of a web browser, and can be useful to escape dead ends and unknown network areas. As a design choice, backtracking may be limited to a certain number of uses by the algorithm.

3.3. Hierarchical Decentralized Search

This section presents Hierarchical Decentralized Search, one possible implementation of Decentralized Search out of the design space described in the previous chapter. What makes this particular implementation interesting is that it has found widespread use in research (e.g., [TSHS12] [HSGS13]) and directly follows Jon Kleinberg's work. For these reasons and to provide a concrete example before going into the details of Ontology-based Decentralized Search, the algorithm is described in this section.

Generally speaking, Hierarchical Decentralized Search is a decentralized search algorithm in a network and uses a tree-shaped hierarchy as its background knowledge. Table 3.2 describes the chosen design parameters.

Background knowledge	complete
ranking	geodesic distance
Node selection	deterministically greedy
Node revisitation	no
multiple targets	not supported
Termination condition	maximum number of steps
Backtracking	yes (unlimited)

Table 3.2.: **Design parameters of Hierachical Decentralize Search.** The table shows the design parameters used in the implementation of HDS, as described in Chapter 3.2.

Algorithm 1 shows the basic algorithm for hierarchical decentralized search. The algorithm is initially started as `HDS(N, H, s, NULL, t)`, where N is the network, H the background hierarchy, s the starting node, $NULL$ the parameter for the parent node (of which there is none, initially), and t the target node. The function then proceeds by selecting a successor node, which is chosen among the links pointing away from the starting node s. The background knowledge is used to gage the fitness of each link and estimates a distance for each potential successor node. As the distance, the geodesic graph distance on the background knowledge H between the nodes s and t is used (written as $d_H(s,t)$). The function then recursively proceeds to the best successor node. This process is repeated as long as there are unexplored nodes in the network that the background knowledge estimates to be more profitable to explore than backtracking.

To avoid loops, each node in the network is explored only once. However, the algorithm may backtrack to the last visited nodes (up until the starting node, if desired). This is used in case of dead ends (articles with no unvisited outgoing links) or at articles providing only links leading further away from the target according to the background hierarchy information.

Algorithm 1 is adapted from [TSHS12] and the standard recursive implementation of depth-first search. In fact, if the selection of the successor node is simplified to always choose the next unmarked node in the list of neighbors without considering the background hierarchy, the algorithm becomes the standard recursive implementation of depth-first search. As for depth-first search, a non-recursive implementation using a stack is evidently also achievable as a similar implementation.

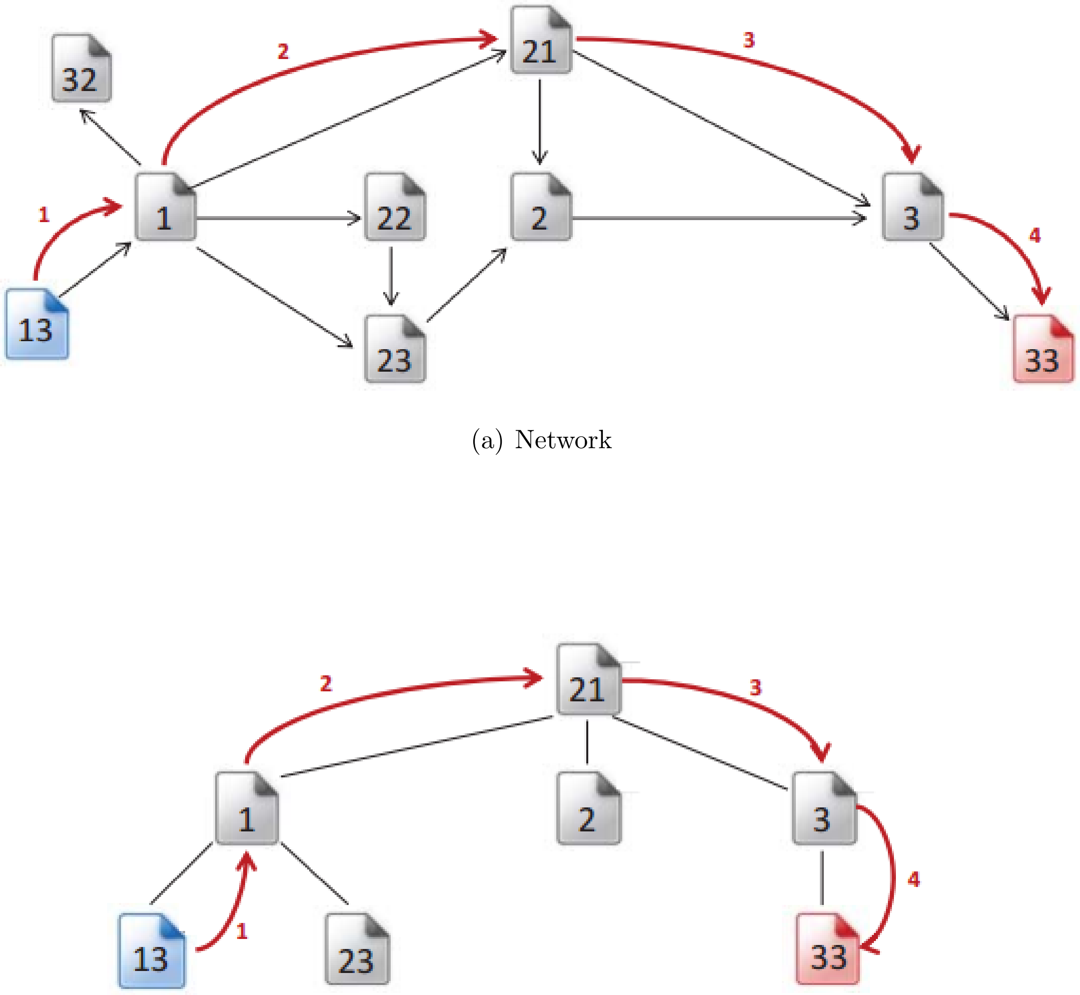
Algorithm 1 Hierarchical Decentralized search. The basic algorithm for hierarchical decentralized search with backtracking. The algorithm is initially called as `HDS(N, H, s, NULL, t)` and recursively calls itself until the target is found or no more exploration is deemed useful by the hierarchy. $d_H(s,t)$ is the geodesic graph distance on the background knowledge H between the nodes s and t. This algorithm is adapted from [TSHS12] and the basic algorithm for depth-first search.

Input: network N, hierarchy H, start node s, predecessor node p, target node t

```
function HDS(N, s, p, t)
    mark $s$ in $N$
    if $s = t$ then
        return True
    repeat
        successor $\leftarrow p$
        $d_{min} \leftarrow \infty$
        for $n \in \Gamma(s) \cup \{p\}$ do          ▷ $\Gamma(s)$ are the neighbors of $s$ in $N$
            if $n$ unmarked and $d_H(n,t) < d_{min}$ then
                $d_{min} \leftarrow d_H(n,t)$
                successor $\leftarrow n$
        if successor $\neq p$ then
            if DecentralizedSearch($N$, $H$, successor, $p$, $t$) $=$ True then
                return True
    until $s = t$
    return False
end function
```

The algorithm described in Algorithm 1 requires the following prerequisites:

1. The hierarchy H is complete, i.e., it possesses knowledge about the entire network N, that is it contains all nodes from the network N.
2. The hierarchy H is connected undirected simple graph.



(a) Network

(b) Background knowledge (hierarchy)

Figure 3.1.: **Example for Hierarchical Decentralized Search** Figure a) shows the graph of an example network, where Hierarchical Decentralized Search is started at node 13 and progresses to node 33. Figure b) shows the corresponding background knowledge. HDS proceeds along the links which seem most profitable (i.e., that are closest to the target node) according to the background knowledge. Figures reprinted from [TSHS12].

3. The network N is a directed simple graph to which no connectivity constraints are imposed.
4. For all node pairs (s, t), where s is the start node and t the target node, the network N contains a path from s to t.

The hierarchy H may also contain additional nodes not present in the hierarchy.

In order to actually obtain a path from this algorithm, a log procedure or a print statement at each step would be required. The pseudo-code for this has been left out for the sake of simplicity.

Figure 3.1 provides an example of Hierarchical Decentralized Search for a small network and a corresponding hierarchy. In the figures, the navigation starts at the node labeled *13* and proceeds to the target node labeled *33*. During the navigation, the next hop to select is determined by the information provided by the hierarchy.

In comparison to previous work with the decentralized search framework such as [HSGS13] and [TSHS12], Algorithm 1 has been improved by the inclusion of backtracking. Backtracking potentially leads to longer paths (because the search does not easily abort) and a higher number of found targets.

Algorithm 1 evidently does not yield optimal solutions, as the value of its information to the navigation process depends on the fitness of the hierarchy. For a hierarchy in tree-form with n nodes and $n - 1$ edges, this is in general also not possible, as a directed simple graph can contain up to $n(n-1)$ edges. In other words, the hierarchy contains only information about $\Theta(n)$ edges, while the network contains $\mathcal{O}(n^2)$ edges, where n is the number of nodes in both the hierarchy and the network.

3.4. Ontology-based Decentralized Search

Ontology-based decentralized search (OBDS) represents a different, more general approach to Decentralized Search. This section provides a general description and motivation of Ontology-based Decentralized Search, while the specific version of OBDS used by this thesis and its chosen design parameters are described in detail in Chapter 4 in the context of the experimental setup.

As the name indicates, Ontology-based Decentralized Search uses ontologies as background knowledge. In contrast to Hierarchical Decentralized Search, the background ontology need not be strictly hierarchical (i.e., not in tree form), but a regular undirected graph is also acceptable. However, all of the ontologies used in this thesis retained a structure similar to a tree, with a clear root concept but a number of concepts with multiple parents.

The use of existing ontologies represents a substantial change in the motivation of the background knowledge: As opposed to previous work in this area, the background knowledge is now exogenous to the network. What this implies is that the hierarchy is based on knowledge independent of the network that the agent navigates on. In previous HDS experiments, the background hierarchy was often calculated from network features (such as node degree or centrality). This worked well for simulations but had the inherent drawback of using ”unfair” knowledge in the sense that it used parts of the global network topology to calculate the background knowledge. With the use of existing ontologies created without explicit knowledge of the network, this issue can be overcome.

All ontologies used in the application of Ontology-based Decentralized Search in this thesis play a key role for their corresponding domain in their research fields. They are hence representative for a good part of the knowledge in these domains (the ontologies are discussed in Chapter 4.2). This permits OBDS with a foundation to more accurately represent the intuitions of human navigation behavior.

The use of ontologies and the associated semantic information open up a range of new possibilities for the application of the background knowledge. The following paragraphs describe the key ideas.

Filtering by relations and properties Because of its use of a background knowledge enriched by semantic information, Ontology-based Decentralized Search is a more general version of the original Decentralized Search approach. In addition to the information conveyed by the nodes and edges of the background knowledge, ontologies add substantially more information to the background knowledge: Ontologies are (in general) made up of different types of relations (such as *is-a* or *part-of*), which can be used to extract different varieties of background knowledge from one and the same ontology. For example, a hierarchical version of the ontology could be extracted by following only the *is-a* relations. Furthermore, ontologies may assign *properties* to their concepts. A background knowledge can hence also be restricted to ontology concepts with a certain property. For example, an ontology could be filtered to contain only contain concepts related to *geography*, thus providing a background knowledge limited to a single domain. This could then be compared with other filtered versions of the ontology. In conclusion, this approach leads to a variety of new and interesting ways to calculate distances and rank the potential successor nodes.

Filtering of important concepts As many ontologies used in real life contain a large number of concepts and relations, OBDS could also be used to establish the most important relations or relation types in an ontology. This could be accomplished by extracting and comparing different versions of an ontology via navigation, and extracting the most frequently used concepts and relations. One application of this could be the generation of a condensed representation of an ontology, providing a more simple overview and introduction into the domain.

Modeling different user groups Ontologies could also be used to model different types of users. A good example for this is the case of the ICD-10 ontology, which provides a classification of diseases. In the ontology, the depth of a disease (i.e., its distance from the root node) corresponds to its specificity. This could be used to model the knowledge of different hospital personnel. For instance, a medical specialist could be modeled by the entire depth of knowledge of one section of the ontology, and a depth-limitation in the other sections. A nurse could be modeled by having a certain depth-limitation in all areas, which would still be less limited than the version of the ontology used to represent a layperson. This could be effectively used to simulate different user groups in medical information systems, without having to carry out actual human user studies which are often expensive and difficult to conduct.

Inference Furthermore, ontologies permit *inference* on their entities. For hierarchical relations this could mean for instance, that subconcepts could be assigned the type of their superconcepts (e.g., the perhaps unfamiliar Supraventricular tachycardia is a subconcept of Heart Disease in ICD-10, which is a more familiar disease). In the case of the cut-off background knowledge, more specific ontology concepts could then be substituted by their inferred superconcepts and provide more information to the navigation process than a pure random guess.

4. Experimental Setup

This chapter describes the materials and methods used in the case study of applying Ontology-based Decentralized Search to Wikipedia and biomedical ontologies. Section 4.1 introduces the design parameters and the setup for Ontology-based Decentralized Search. Section 4.2 introduces the four ontologies used. Section 4.3 provides the details for the Wikipedia articles as well as information about the mapping between Wikipedia articles and the ontology concepts. Section 4.4 describes the navigation scenarios simulated in this thesis, followed by Section 4.5 describing the setup of the user study used to evaluate the results.

4.1. Ontology-based Decentralized Search

4.1.1. Design Parameters

The concrete parameters for the Ontology-based Decentralized Search algorithm used in the experimental phase of this thesis are again chosen from the parameter space described in Chapter 3.2. Its parameters are similar to those of Hierarchical Decentralized Search, but include the support for multiple targets. The design parameters are summarized in Table 4.1.

Handling of multiple targets The version of OBDS used in the experiments supports multiple targets as follows: At each step during the navigation, OBDS decides what target is currently estimated to be the closest and then takes a step into that target's direction. This process continues until all targets are found. Note that while this could lead to looping behavior in general, loops are prevented in the algorithm by the marking of nodes as visited and further excluding visited nodes from the exploration.

Background knowledge	complete
ranking	geodesic distance
Node selection	deterministically greedy
Node revisitation	no
multiple targets	decide at each step
Termination condition	maximum number of steps
Backtracking	yes (unlimited)

Table 4.1.: **Design parameters of Ontology-based Decentralized Search.** The table shows the design parameters used in the implementation of OBDS, as introduced in Chapter 3.2.

4.1.2. Further adaptions

Moreover, OBDS, as used in the experiments, additionally includes a *starting portal* and a *home button*.

Starting portal OBDS always starts from the same starting portal in the graph. This starting portal is an artificially introduced network node containing links to multiple other network nodes and serving as an entry point into different areas of the network. The concrete portals in association with the search scenarios used are discussed in section 4.4.

Home Button OBDS contains a *home button* that leads back to the starting portal. In the simulation, this button is available at all times, and the algorithm decides at each step whether to continue the search or to return back to the portal.

4.1.3. Ontology-based design decision

Ontology-based Decentralized Search was applied using four established ontologies from the biomedical domain as background knowledge (these ontologies are discussed in the next section).

In the experiments, Ontology-based Decentralized Search was used with both filtered and unfiltered ontologies. First, OBDS was applied to three different ontologies that were not filtered. This meant, that all concepts and relation types present in the ontologies (and mapping to the data sets) were used as the background knowledge. The results of the simulations were then compared on the same information network, which meant that the three ontologies effectively modeled different user groups on the same set of data. Second, OBDS was applied to three filtered versions of one and the same ontology. These version were filtered by concept properties and formed a partition of the ontology (i.e., no concept was part of more than one subontology). This was then used to compare the performance of these subontologies.

4.1.4. Algorithm

The exact algorithm for Ontology-based Decentralized Search, as used in this thesis, is detailed in Algorithm 2.

Algorithm 2 Ontology-based Decentralized search. The algorithm for ontology-based decentralized search with backtracking. In addition to hierarchical decentralized search, this algorithm starts from a starting portal and searches for multiple targets. The algorithm is initially called as `OBDS(N, O, s, NULL, t)` and recursively calls itself until all the targets are found or no more exploration is deemed useful by the background ontology.

```
Input: network $N$, background ontology $O$, start node $s$, predecessor node $p$, target
    node $t$
function OBDS(N, O, s, p, t)
    mark $s$ in $N$
    if $s \in t$ then
        remove $s$ from $t$
    if $s = \emptyset$ then
        return True
    repeat
        successor $\leftarrow p$
        $d_{min} \leftarrow \infty$
        for $n \in \Gamma(s) \cup \{p$, portal$\}$ do            ▷ $\Gamma(s)$ are the neighbors of $s$ in $N$
            for target $\in t$ do
                if $n$ unmarked and $d_H(n$,target$) < d_{min}$ then
                    $d_{min} \leftarrow d_H(n$,target$)$
                    successor $\leftarrow n$
        if successor $\neq p$ then
            if DecentralizedSearch($N$, successor, $p$, $t$) $=$ True then
                return True
    until $s = t$
    return False
end function
```

The basic structure of Algorithm 2 is the same as the one of Algorithm 1. The algorithm is initially called as `OBDS(N, O, s, NULL, t)`, where N is the network, O is the background ontology, s the starting node, $NULL$ is the parameter for the parent node (initially not set) and t the set of target nodes. Like `HDS`, the algorithm calls itself recursively. The major changes to the basic algorithm for Hierachical Decentralized Search lie in the multiple targets, which are now evaluated at every call of the function, and the inclusion of the starting portal in the list of potential successors at each step during navigation.

Figure 4.1 presents an example of the application of Ontology-based Decentralized Search. The example is taken from the application of the method to the data set of a network of biomedical Wikipedia articles and the ICD-10 ontology, which is described in the following section. The search agent in the examples starts from a hypothetical portal containing links to a number of common diseases. The algorithm then proceeds towards the target, making use of ICD-10 as its background ontology.

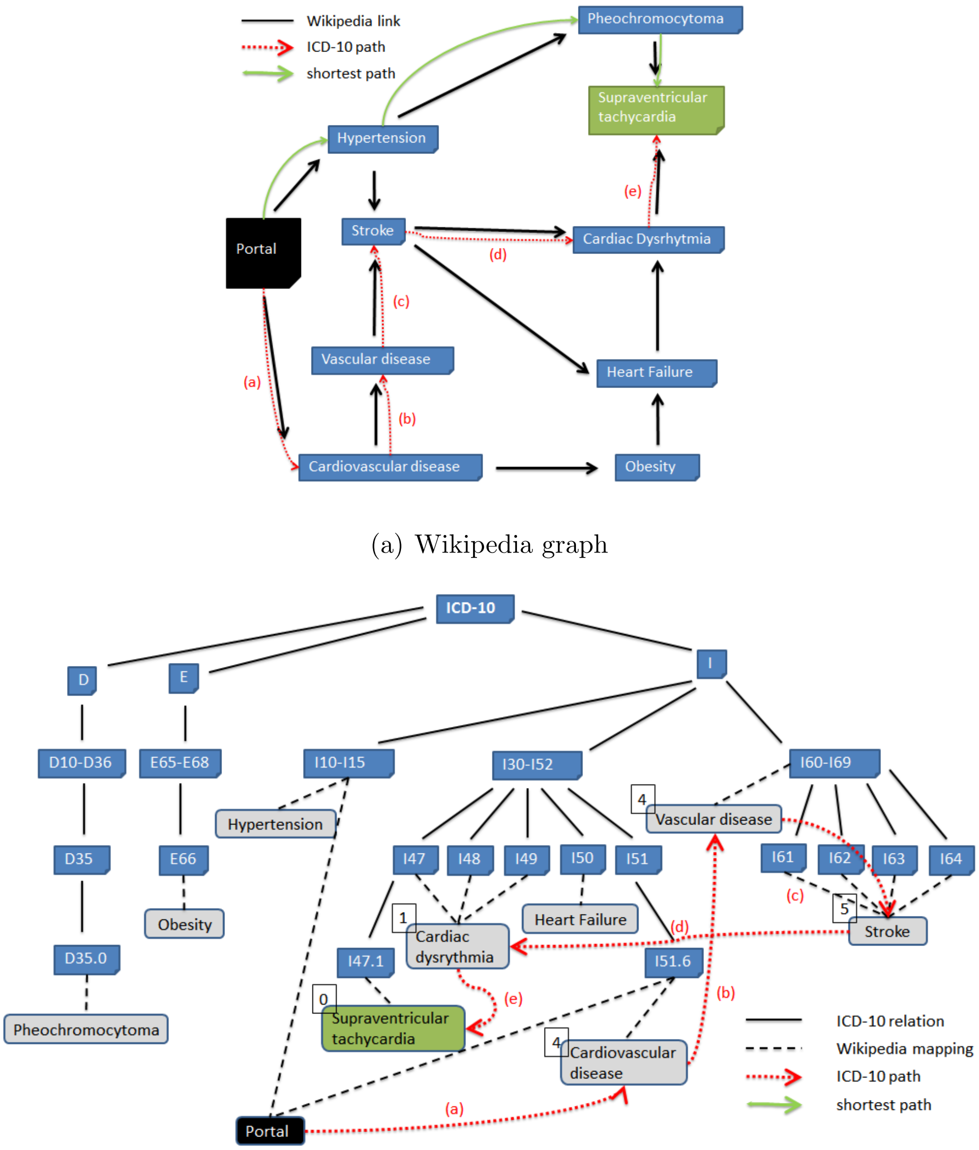
4.2. Ontologies in the biomedical domain

In this thesis, the the following four ontologies are used as background knowledge (all from the biomedical domain) :

The **International Classification of Diseases, tenth revision (ICD-10)** is a classification of diseases, signs and symptoms first published in 1992 and maintained by the World Health Organization (WHO). The ICD-10 had its origins in the classification of causes of deaths and has become the standard for diagnostic classification, presently used by over a hundred countries to report mortality statistics. It is also widely used for epidemiology, health management as well as clinical purposes and is available in 46 languages [ICD12]. It enables the exchange of data between countries and languages. The version used in this thesis contained 12,417 concepts (see Table 4.2 for an overview of the ontologies used in this thesis). ICD-10 consists of 22 top-level nodes[1] termed *chapters* and assigns a code (or a range of codes) to every disease in its domain. Every code starts with a letter indicating the chapter, followed by numbers further specifying the disease. For example, the code `I47.1` denotes the disease Supraventricular tachycardia and `E66` denotes Obesity.

Medical Subject Headings (MeSH) is a controlled vocabulary thesaurus for journal articles in the medical domain. MeSH is maintained by the U.S. National Library of Medicine. The ontology forms a tree-structure with 16 top-level concepts and contains

[1] Top-level nodes are the direct neighbors of the root node in the ontology.



(a) Wikipedia graph

(b) Background knowledge (ICD-10)

Figure 4.1.: **Example for Ontology-based decentralized search** OBDS starts from a hypothetical portal containing links to a number of common diseases. The algorithm then proceeds towards the target, making use of the background ontology. In this figure, the network is represented by Wikipedia articles and the background knowledge is represented by the ICD-10 ontology containing diseases. A possible navigation path in the graph (green, solid) is guided by ICD-10, which differs from the shortest path (red, dotted). The numbers along the ICD-10 path show the distance to the target, according to ICD-10.

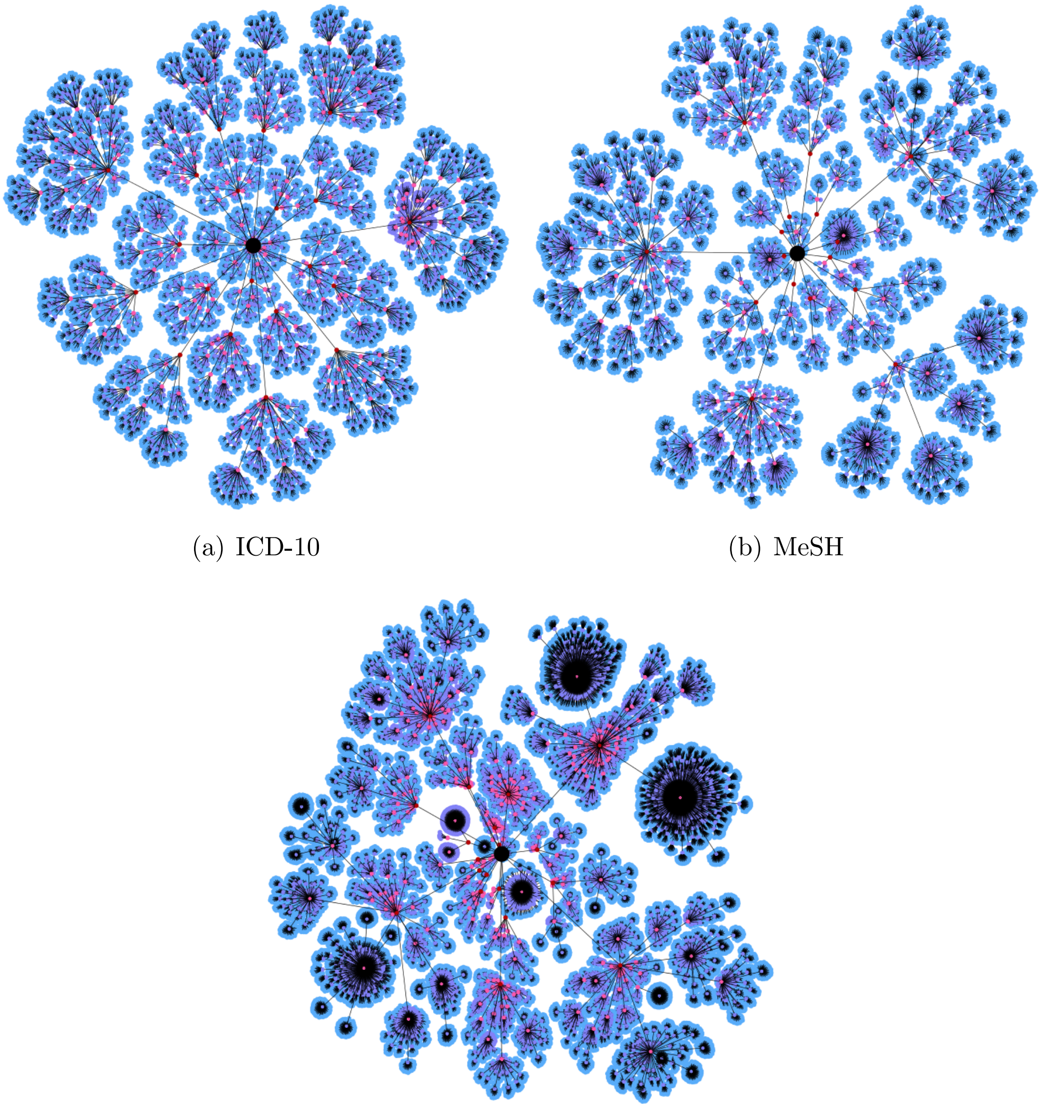
26,142 terms (called *descriptors*) [MeS12] . Descriptors are graph leaves and attached to one or more tree nodes (which are not descriptors). As such, the complete ontology graph used in this thesis contained 80,689 nodes. MeSH extends beyond biomedical concepts and comprises terms from other domains such as Geography, Technology or Publication Characteristics. However, only the biomedical terms were used in this thesis. MeSH assigns unique identification strings of the form `D0136170` to its concepts, but also denotes each concept via a code that references the tree structure. For example, `C14.280.067.845.880` denotes Tachycardia, Supraventricular and `C18.654.726.500` denotes Obesity.

Systematized Nomenclature of Medicine–Clinical Terms (SNOMED CT) [PS00] is a clinical health care terminology used in electronic health record systems. SNOMED CT had its origins in the merge of SNOMED RT (developed in the United States) and Clinical Terms Version (developed in the United Kingdom). The revision used in this thesis contained 295,482 concepts, which made it by far the largest ontology examined in the simulations. SNOMED CT consists of 19 top-level concepts and assigns numeric identifiers to its concepts. For example, Supraventricular Tachycardia is identified with code `6456007` and Obesity with `414916001`.

The **Gene Ontology (GO)** [Ash00] is a controlled vocabulary of terms used for the annotation of genes and gene products. It is part of the Open Biomedical Ontologies (OBO) effort to create controlled vocabularies in biology and medicine. The Gene Ontology consists of 37,779 concepts divided among three different subontologies, which cover the cellular component, the molecular function and the biological process, respectively. In its filtered form used in this thesis, the three subontologies take the form of disjoint trees. In general, genes and gene products are annotated with entries stemming from all three subontologies. Entries in the Gene Ontology are assigned a numeric identification code such as `GO:0000016` for the entry lactase activity.

Figures 4.2 and 4.3 depict aspects of the examined ontology graphically. The figures were created based on the root node of the ontology (which is explicitly specified in the ontology descriptions) and following links to neighbor concepts up until a depth of four. These figures permit the visual inspection of the ontology structures and reveal information about the densities and the number of top-level concepts. For the Gene Ontology (Figure 4.3), it is clearly visible that the biological process subontology includes far more concepts than the other two subontologies.

Table 4.2 displays statistics about the data sets used for this thesis (the Wikipedia statistics in this table are explained in the next section). Given the undirected graph $G = (V, E)$, the density was calculated as

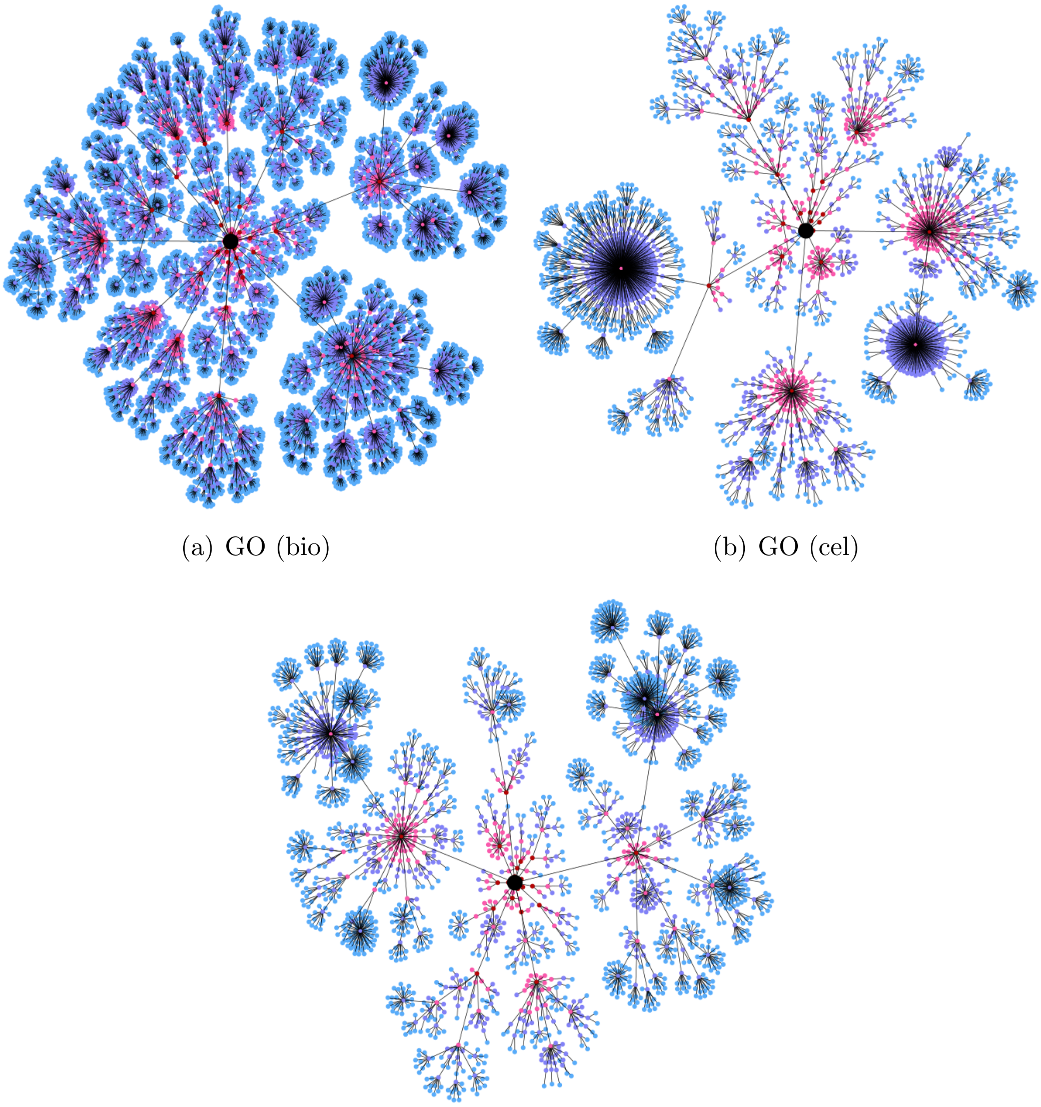


(a) ICD-10

(b) MeSH

(c) SNOMED CT

Figure 4.2.: **Structure of the four top levels of ICD-10, MeSH and SNOMED CT.** The root node is displayed in the middle of each plot. The figures show all ontology concepts up until a distance of four from the root node. Color indicates distance, with red being close to the root and blue being farther away. SNOMED CT (depth 16) is clearly broader than MeSH (depth 14), which stems from the fact that the latter contains roughly four times as many concepts as the former.



(a) GO (bio)

(b) GO (cel)

(c) GO (mol)

Figure 4.3.: **Structure of the four top levels of the Gene Ontology** The figure shows the structure for the three subontologies making up the GeneOntology (biological process, cellular component and molecular function). The root node is displayed in the middle of each plot. The figures show all ontology concepts up until a distance of four from the root node. Color indicates distance, with red being close to the root and blue being farther away. The biological process subontology is visibly broader than its sibling ontologies.

	Description	ICD-10	MeSH	SNOMED CT
Ontology	concepts	12,417	80,689	295,482
	top-level	22	16	19
	links	12,416	112,463	440,408
	density	8.05×10^{-5}	1.73×10^{-5}	5.04×10^{-6}
	depth	4	14	16
	relation	is-a	is-a, part-of	is-a
Wikipedia	articles	2,673	2,584	1,594
	links	31,863	27,350	14,539
	density	4.46×10^{-3}	4.10×10^{-3}	5.73×10^{-3}

	Description	Gene Ontology	bio	cel	mol
Ontology	concepts	37,779	23,691	3,020	9,413
	top-level		25	19	21
	links	67,991	51,397	5,617	10,977
	density	4.76×10^{-5}	9.16×10^{-5}	6.16×10^{-4}	1.24×10^{-4}
	depth		10	7	10
	relation	is-a, part-of, regulates			
Wikipedia	articles	3,445			
	links	15,643			
	density	1.32×10^{-3}			

Table 4.2.: **Characteristics of the data sets used for this thesis** The tables display statistics about the examined ontologies as well as the sets of Wikipedia articles mapped to those articles. As SNOMED CT was only used in conjuction with MeSH and SNOMED CT, the column shows the information for the intersection of Wikipedia articles mapping to all three ontologies. This thesis used data from the four ontologies listed, all of which were in the biomedical domain. For the GeneOntology, the triples (bio - cel - mol) list statistics for the three subontologies making up the GeneOntology.

$$D = \frac{|E|}{|V|(|V|-1)},$$

where $|E|$ is the number of edges in the graph and $|V|(|V|-1)$ is the maximum number of edges in an undirected graph. For the directed Wikipedia network, the density was calculated as

$$D = \frac{2|E|}{|V|(|V|-1)},$$

as an undirected graph has a maximum of $\frac{2}{|V|(|V|-1)}$ edges.

Although SNOMED CT has a larger amount of edges than ICD-10 and MeSH in absolute terms and is denser for its first levels away from the root node (see Figure 4.2), its density is actually lower because of its relatively low number of edges compared to its nodes. The same is the case with the biological process subontology of the Gene Ontology (see Figure 4.3).

4.3. Wikipedia Articles

The English Wikipedia was chosen as the information network to base the investigations of this thesis on. The idea behind this was to use a widely used information network with data available and accessible for research purposes.

Wikipedia is authored and edited by millions of users world-wide. The investigation of navigation on Wikipedia represents a stepping stone towards verifying the results of Ontology-based decentralized search on general information systems or the Web as a whole.

To investigate the ontologies described in the previous section, matching Wikipedia articles from the biomedical domain are required. The Wikipedia articles used in this thesis were obtained from a dump of the English Wikipedia and represent the articles of the encyclopedia as of December 1, 2011[2].

To extract articles from the biomedical domain corresponding to ontology concepts, the articles were mapped to the ontologies by parsing the articles' info boxes.

Disease articles on Wikipedia commonly make use of a `Template:Infobox disease`[3], which offers several options to reference medical ontologies such as ICD-10 or MeSH (see Figure 4.4 for an example). These template fields in the `Infobox disease`, as well as two other infobox templates, were then used to map Wikipedia articles to their ontology counterparts in ICD-10 and MeSH for their use in this thesis. All infoboxes used are listed in Appendix A.

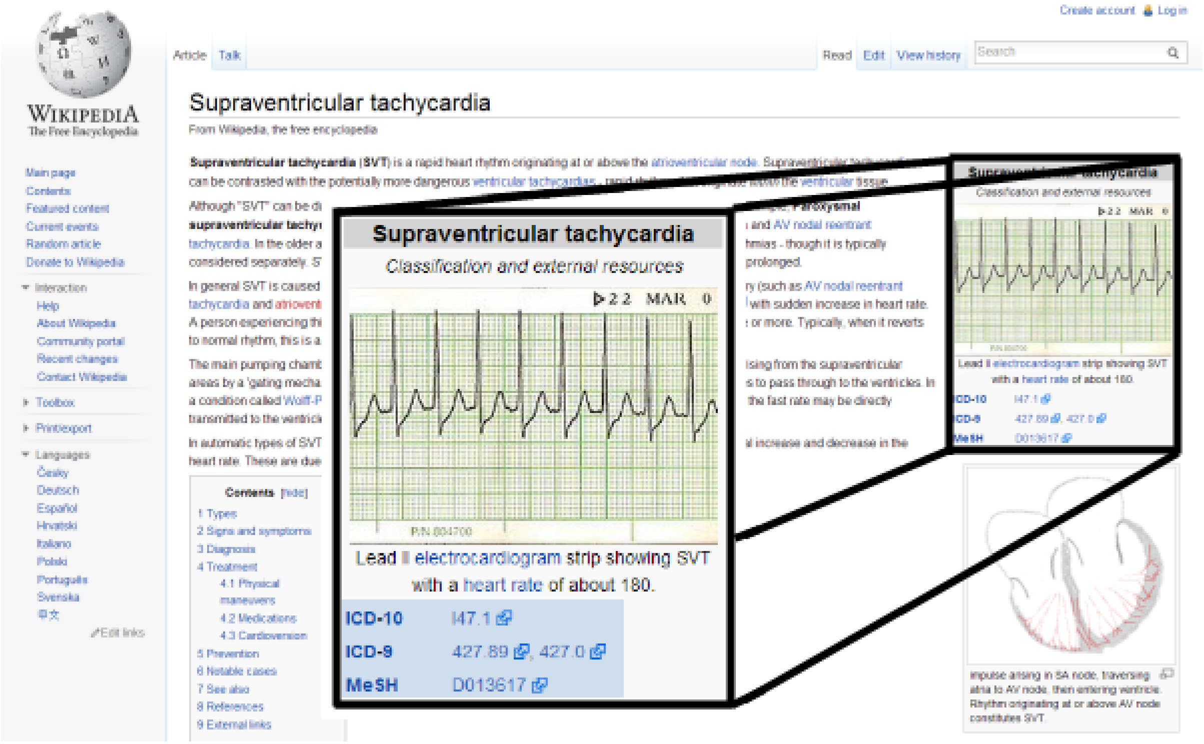
SNOMED CT is proprietary and as such not present in Wikipedia info boxes. As a consequence, its articles could not directly used to relate Wikipedia articles to the ontology concepts. As an alternative, semantic mappings from BioPortal [WNS+11] were used to map Wikipedia articles to SNOMED CT. In total, 1,594 Wikipedia articles occuring in both ICD-10 and MeSH were mapped to SNOMED CT with this

[2] http://dumps.wikimedia.org/enwiki/20111201/

[3] http://en.wikipedia.org/wiki/Template:Infobox_disease

method. Hence, all of the ICD-10, MeSH and SNOMED-CT ontologies were used in the experiments, and the ontologies were not filtered by properties or relations.

The Gene Ontology is different in that it is not used for 1:1 mappings but for the *annotation* of Wikipedia articles. Articles are assigned different annotations from the three subontologies making up the controlled vocabulary of the GeneOntology. For instance, Insulin is annotated with protease binding, hormon activity and protein binding, stemming from the Molecular function part of the GeneOntology. This was taken as a motivation for the split of the Gene Ontology into three subontologies in the experiments of this thesis. Hence, the Gene Ontology was filtered by concept properties into three different ontologies, which still used all of the available relation types.



Supraventricular tachycardia

From Wikipedia, the free encyclopedia

Supraventricular tachycardia

Classification and external resources

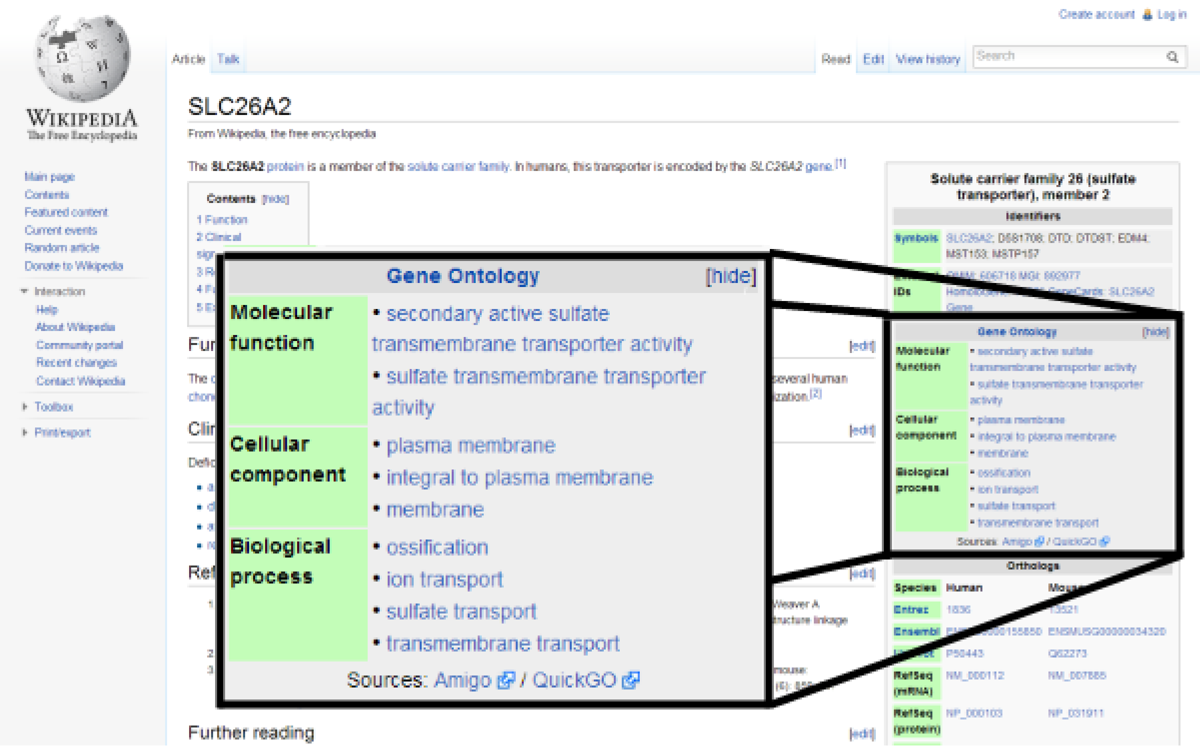
Lead II electrocardiogram strip showing SVT with a heart rate of about 180.

ICD-10	I47.1
ICD-9	427.89, 427.0
MeSH	D013617

Figure 4.4.: **Example for an infobox template used in disease articles on Wikipedia.** Disease articles commonly make use of an Infobox disease template, which offers fields for ontology codes. The template fields in the infoboxes were used to map Wikipedia articles to their ontology counterparts.

As a result, the Wikipedia articles were linked to all (up until 50 or more) related concepts from all three subontologies of the Gene Ontology. For this, the corresponding fields in templates created by the ProteinBoxBot[4] were inspected in order to

[4] http://en.wikipedia.org/wiki/User:ProteinBoxBot



SLC26A2

From Wikipedia, the free encyclopedia

The **SLC26A2** protein is a member of the solute carrier family. In humans, this transporter is encoded by the *SLC26A2* gene.[1]

Solute carrier family 26 (sulfate transporter), member 2

Identifiers	
Symbols	SLC26A2; D5S1708; DTD; DTDST; EDM4; MST153; MSTP157

Gene Ontology	[hide]
Molecular function	• secondary active sulfate transmembrane transporter activity • sulfate transmembrane transporter activity
Cellular component	• plasma membrane • integral to plasma membrane • membrane
Biological process	• ossification • ion transport • sulfate transport • transmembrane transport

Sources: Amigo / QuickGO

Orthologs		
Species	Human	Mouse
Entrez	1836	13521
Ensembl	[illegible]00155850	ENSMUSG00000034320
UniProt	P50443	Q62273
RefSeq (mRNA)	NM_000112	NM_007885
RefSeq (protein)	NP_000103	NP_031911

Further reading

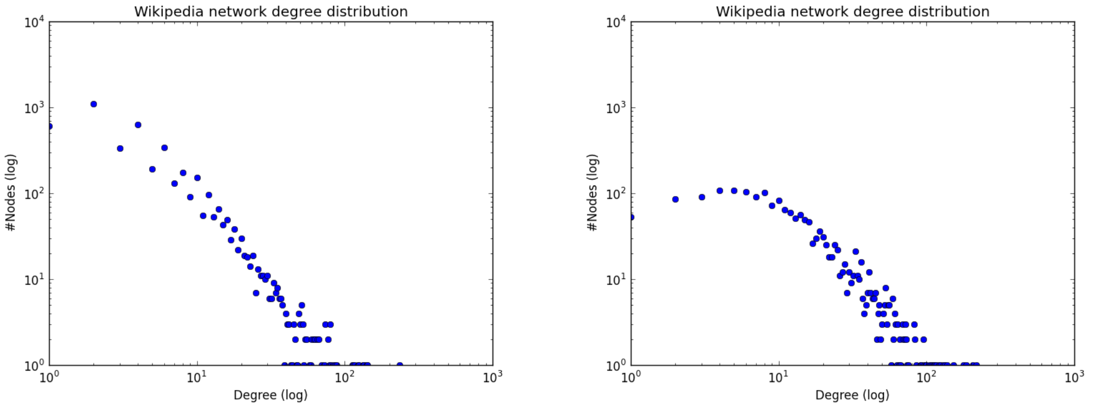
Figure 4.5.: **Example for Gene Ontology infobox template used in disease articles on Wikipedia.** Gene articles commonly make use of a Gene Ontology template, which offers fields for ontology codes. The template fields in the infoboxes were used to map Wikipedia articles to their ontology counterparts.

extract the relevant mappings to the Gene Ontology. The `Portal: Gene Wiki` page on Wikipedia contains around 10,000 articles on human genes and proteins. Articles in this domain are usually either created or annotated by the ProteinBoxBot, using information from the Gene Ontology and other projects. An example for a template can be seen in Figure 4.5.

As a great number of these articles are very domain-specific and only very few editors are knowledgeable enough to add to them, there is a large number of stubs (very short articles) and orphans (articles not linked to by any other Wikipedia article). This is also reflected in the low number of links between the articles in this data set, as compared to the other data sets (see the density information in Table 4.2).

Further information about the Wikipedia articles used in this thesis is provided in Table 4.2.

Figure 4.6 shows the degree distribution for the two graphs extracted from the Wikipedia



(a) Gene Ontology

(b) ICD-10/MeSH/SNOMED CT

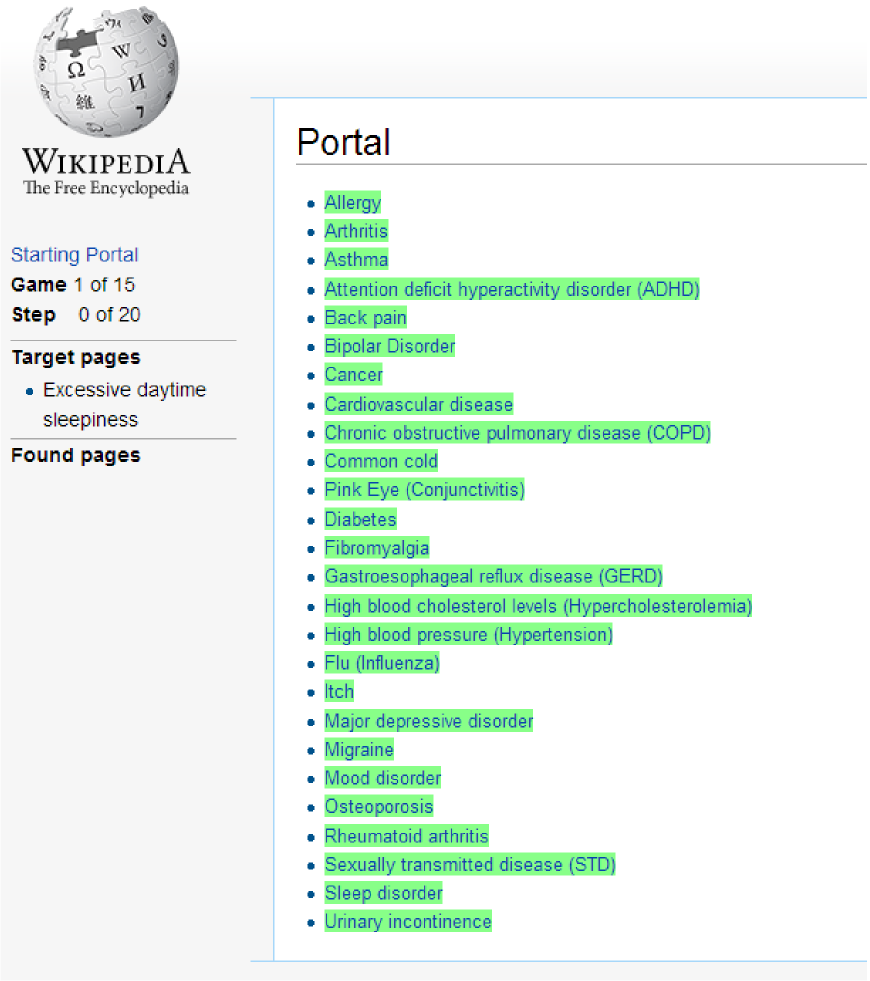
Figure 4.6.: **Wikipedia graph degree distributions** The figure shows the degree distribution for the two graphs extracted from the Wikipedia article network via the Gene Ontology and the intersection of the articles mapping to ICD-10/MeSH/SNOMED CT. Compared to the ICD-10/MeSH/SNOMED CT data set, the Gene Ontology data set contained a higher number of articles with only very few or no links. This is likely due to the higher number of stubs and orphans in the Gene Ontology-related Wikipedia articles.

article network via the Gene Ontology and the intersection of the articles mapping to ICD-10/MeSH/SNOMED CT. Compared to the ICD-10/MeSH/SNOMED CT data set, the Gene Ontology data set contained a higher number of articles with only very few or no links. This is likely due to the higher number of stubs and orphans in the Gene Ontology-related Wikipedia articles.

As they are created and maintained by a large number of editors without a central control institution, the mappings from Wikipedia articles to ontology concepts obtained this way are not necessarily complete or even correct. However, this thesis does not claim to represent perfect data but instead investigates to what extent a human created information network can be explained and modeled via methods relating to decentralized search.

4.4. Navigation Scenarios

For the following investigations and the application of Ontology-based decentralized search in this thesis, two navigation scenarios were considered. Both scenarios started



Starting Portal
Game 1 of 15
Step 0 of 20

Target pages

- Excessive daytime sleepiness

Found pages

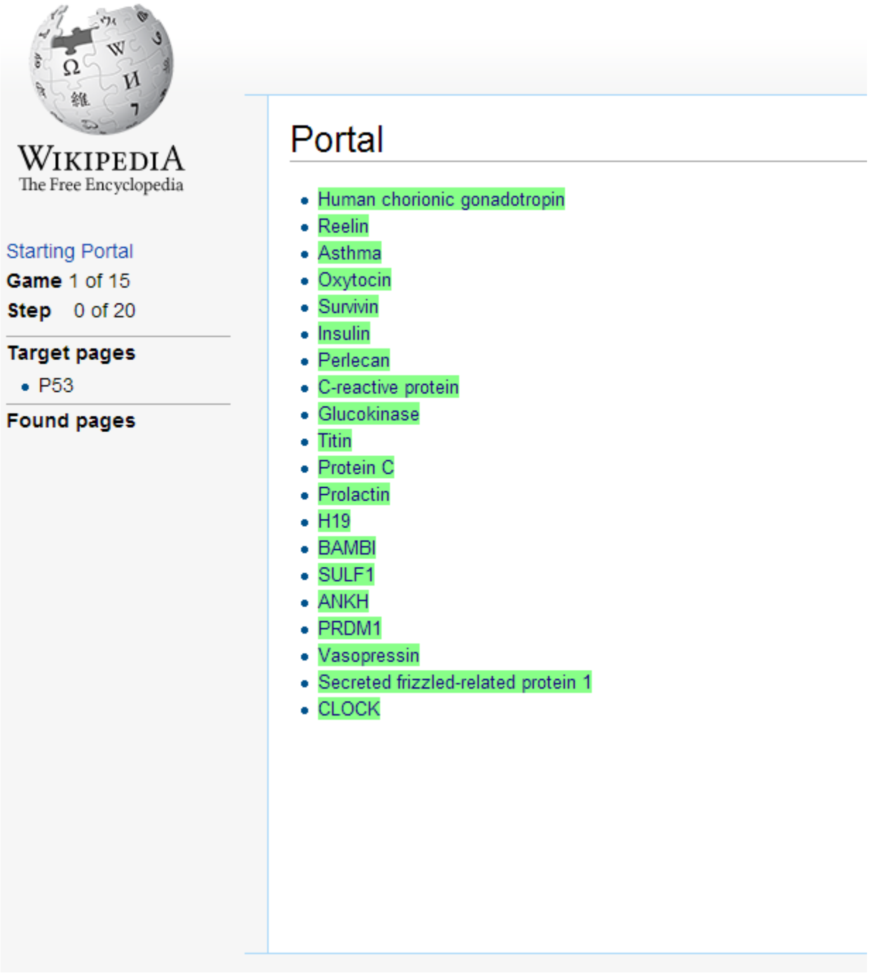
Portal

- Allergy
- Arthritis
- Asthma
- Attention deficit hyperactivity disorder (ADHD)
- Back pain
- Bipolar Disorder
- Cancer
- Cardiovascular disease
- Chronic obstructive pulmonary disease (COPD)
- Common cold
- Pink Eye (Conjunctivitis)
- Diabetes
- Fibromyalgia
- Gastroesophageal reflux disease (GERD)
- High blood cholesterol levels (Hypercholesterolemia)
- High blood pressure (Hypertension)
- Flu (Influenza)
- Itch
- Major depressive disorder
- Migraine
- Mood disorder
- Osteoporosis
- Rheumatoid arthritis
- Sexually transmitted disease (STD)
- Sleep disorder
- Urinary incontinence

Figure 4.7.: **WebMD portal** Starting portal used for ICD-10, MeSH and SNOMED CT. The portal was obtained by mapping navigation bar articles from `WebMD.com` to Wikipedia articles.

from a hypothetical starting portal, which is described in the first section, followed by two sections describing the search scenarios themselves.

A point of criticism in HDS was the artificiality of the navigation scenarios. While there are valid scenarios for point to point navigation in a network, it may not be the most frequent scenario in real information networks. The introduction of a combination of a starting portal and multiple targets addresses this by simulating search starting from an artificially introduced portal. The home button was implemented with the intention of helping users lost in an unknown area of the network to get back to the portal directly.



Starting Portal
Game 1 of 15
Step 0 of 20

Target pages

- P53

Found pages

Portal

- Human chorionic gonadotropin
- Reelin
- Asthma
- Oxytocin
- Survivin
- Insulin
- Perlecan
- C-reactive protein
- Glucokinase
- Titin
- Protein C
- Prolactin
- H19
- BAMBI
- SULF1
- ANKH
- PRDM1
- Vasopressin
- Secreted frizzled-related protein 1
- CLOCK

Figure 4.8.: **Gene Wiki portal]** Starting portal used for the Gene Ontology. The portal uses the longest and most frequently visited Wikipedia articles as listed on the Gene Wiki Portal `http://en.wikipedia.org/wiki/Portal:Gene_Wiki`.

Starting Portal The hypothetical Wikipedia portal included links pointing to a selection of suitable articles.

For ICD-10, MeSH and SNOMED CT, the 25 health conditions listed in the main navigation toolbar of `WebMD.com` were taken as the portal links. WebMD is a popular health information web portal, providing information about a range of medical conditions. The links in the navigation toolbar are a pointing to common diseases and conditions. The intention behind selecting these links was to model a navigation scenario where a user is looking to obtain information about a disease, starting from a web portal providing entry links towards different medical areas. Medical websites, such as WebMD are frequently [LTSM03] used to obtain information about diseases or as a first information before consulting a medical doctor. Each one of these conditions was manually mapped to a Wikipedia article from the dataset. These articles were then used as the outgoing links from the artificial portal.

For the Gene Ontology, the articles listed in the two top 10 lists (ranked by article word and view count) as shown on the `Portal: Gene Wiki`[5] were used. The idea behind choosing these two lists was that users interested in gene-related articles might start their navigation from the Gene Wiki portal on Wikipedia, which provides an overview and an introduction into the domain.

The portals are displayed in Figure 4.7 and Figure 4.8.

Single-target Search The single-target search scenario consisted of navigation to one target article. This was intended to model the scenario of having a concept on the tip of one's tongue, and navigating to rediscover it.

To provide an example, imagine that Alice accompanied her mother to a physician, who diagnosed her mother with a certain disease. Back at home, Alice realizes that she forgot the exact name of the condition. However, she remembers that the disease was somehow related to heart rhythm problems. Trying to recover the exact name, she goes to Wikipedia and navigates from a (hypothetical) starting portal. She first clicks the link on Cardiovascular Disease *and navigates her way towards the lost disease. This example is presented in detail in Figure 4.1, where the search agent is assumed to navigate based on OBDS with ICD-10 background knowledge.*

Multiple-target Search For multiple-target search, the difference to the single-target scenario was in the targets, which consisted of *target sets* of 2 to 10 articles. The other parameters of the simulation (the starting portal, decentralized search and the ontology as background knowledge) were set up in the same way as the single-target search.

Multiple-target search was used to model a scenario of exploratory search. In exploratory search, users explore a space of resources rather than trying to find one specific target [Mar06]. In analogy to Alice's example, her mother might navigate Wikipedia to learn about several potential diseases, causes or secondary effects.

Multiple-target search used clusters of semantically similar Wikipedia articles as the target sets. These clusters were obtained automatically through *k-means clustering*.

[5] http://en.wikipedia.org/wiki/Portal:Gene_Wiki

The features used to cluster the articles were calculated via *TF-IDF* (using scikit-learn [PVG+11]).

TF-IDF is a metric that states the importance of a term relative to a collection of documents. Let D be a collection of documents. The *term frequency* tf for a term t in a document d is then defined as the number of occurrences of the term in the document. The *inverse document frequency idf* is then defined as the

$$idf(t, d) = \log \frac{N}{n},$$

for a term t, a document d, the number of documents N and n the number of documents containing term t. The *term frequency-inverse document frequency (TF-IDF)* is then

$$tf(t) \times idf(t, D).$$

Out of the created clusters, those clusters containing two to fifteen articles were used in in the simulations. Several examples for clusters are given in Table 4.3 (with headings added manually).

In both scenarios, the target article or the set of target articles was directly known to the simulation. This was used to model the somewhat familiar article Alice was trying to reach. Although Alice did not know the exact name of her target, she could roughly place it in a category, to which she then navigated using her own background knowledge. The simulations modeled this by calculating distance directly to the target node on the background knowledge to determine the best link to click.

These clusters were used in all simulations except in the human subject study where manual clusters were developed.

4.5. User Study

To evaluate the simulations and compare the results to human click data, a user study on Wikipedia navigation was carried out. Eight participants without any particular background in medicine were recruited for this study. All of them were graduate students at Stanford University at the time of the user study and were native speakers

Nausea-related	**Stomach-related**
Vomiting	Linitis plastica
Nausea	Stomach cancer
Motion sickness	Gastritis
Morning sickness	Atrophic gastritis
Drooling	Ménétrier’s disease
Hyperemesis gravidarum	Achlorhydria
	Gastroparesis
	Duodenal cancer
	Gastric dumping syndrome
	Stomach disease
Sleep-related	**Nails-related**
Delayed sleep phase syndrome	Leukonychia
Shift work sleep disorder	Psoriatic nails
Sleep disorder	Beau’s lines
Rhythmic movement disorder	Nail biting
Night terror	Nail disease
Parasomnia	Ingrown nail
Irregular sleep–wake rhythm	Subungual hematoma
Hypersomnia	
Night sweats	
Excessive daytime sleepiness	
Cough-related	**Obesity-related**
Bronchitis	Childhood obesity
Chronic bronchitis	Adiposogenital dystrophy
Acute bronchitis	Obesity
Cough	Overweight
Sputum	

Table 4.3.: **Examples for clusters of Wikipedia articles used in exploratory search.** The table shows three examples of clusters used in the simulations. TF-IDF features and k-means clustering was used to automatically group Wikipedia articles into semantically related groups of two to ten articles.

of English[6]. The participants were asked to navigate Wikipedia, modeling the scenario of navigating to find diseases.

The study used the intersection data set of ICD-10, SNOMED CT and MeSH, containing 1,594 Wikipedia articles. Howereve, a large share of these articles turned out to be too specialized for test subjects not particularly familiar with the medical domain (with article names such as Halitosis, Aniseikonia or Milroy's disease, which left users puzzled in a pilot study). As a result, the user study relied on 100 manually selected generally better known targets (such as Pneumonia, Stomach cancer or Asthma), out of which 20 clusters of four articles each were arranged manually. The 100 single targets and the 20 clusters are listed in Appendix C.

Each participant completed a total of 15 navigation tasks. A navigation task consisted of finding a given target node in the subset of the Wikipedia network. The starting point for a task was always the portal, and participants could only click links to move forward. As a starting point, the hypothetical WebMD Wikipedia portal also used by the simulations (see Figure 4.7) was set. To deal with potential frustration, participants were given the possibility to abort the current task if they had not found the target(s) after half of the maximum number of steps (20 for single targets and 40 for multiple targets). As in the simulations, backtracking (using the back button in the browser) and jumping back to the portal by clicking a home link were enabled at all times.

Participants volunteered for participation and received no payment in return, but were offered free food and beverages. The tasks were given without a time limit.

Instructions given to participants The verbatim copy of the instructions given to the participants of the user study is detailed in the Appendix in Section B. Each participant was briefed about these instructions beforehand. Specific attention was paid to use the same briefing structure and words for every participant.

[6] Since many biomedical terms may be hard to understand for users who speak English as a second language, the study of native English speakers represents a more accurate image of the obtained user behavior

5. Applying Ontology-based Decentralized Search

Based on the ontologies, the Wikipedia articles, the starting portals and the two navigation scenarios described in the previous chapter, Ontology-based Decentralized Search was then simulated and evaluated. This chapter starts with a discussion of the evaluation metrics used to assess the methods, followed by the description of the upper and lower bounds used to compare the results against. In the third section, the results are detailed and discussed.

5.1. Evaluation Metrics

Building upon [KPK+10] and previous related research [HSGS13] [TSHS12], three main metrics are used to evaluate the resulting navigation paths.

5.1.1. Success Ratio

In accordance with [HSGS13], the *success ratio* s is defined as the fraction of target nodes found by the navigator. Let P be the set of of target nodes and W be the set of target nodes that were successfully navigated to by the simulations. The success ratio s is then defined as

$$s = \frac{|W|}{|P|}.$$

The success ratio measures the extent to which an agent is successful in finding a target. For instance, a success ratio of 0.9 or 90% states that 90% of the targets have been found.

5.1.2. Stretch

The *stretch* is the average ratio of found path lengths to shortest path lengths [HSGS13]. Let $l(t)$ be the length of the shortest path from the portal to the target node t and let $h(t)$ be the length of the path to the target found by the agent. The stretch τ is then defined as

$$\tau = \frac{1}{|W|} \sum_{t \in W} \frac{h(t)}{l(t)}.$$

The stretch measures the efficiency of search. For example, a stretch of 1.2 states that the paths an agent was able to find are - on average - 20% longer than the shortest paths for these targets.

In order to allow for a more granular application of these metrics, the paths resulting from the simulations are split according to the underlying actual shortest path length to the target node. This approach follows the work of [HS11] and [TSHS12].

5.1.3. Accumulated Success Ratio

In addition to these two established metrics, a further extension to them was the *accumulated success ratio as*, which is the fraction of nodes found up until a certain number n of steps.

$$as(n) = \frac{|W_n|}{|P|},$$

where W_n is the set of target nodes reached by the simulation in n steps or less.

These metrics give a means of analyzing what paths were found by the simulations and how much longer than the shortest paths they were. For all evaluations, a maximum number of 20 clicks for the single-target scenario and 40 clicks for the multiple-target scenario were assumed. While these limits are certainly variable in the setting of navigation, the chosen numbers represent a reasonable limit of potential clicks that a user might undergo in order to find the target article on an information network such as Wikipedia.

5.2. Upper and Lower Bounds

In order to place the results within boundaries, the comparison with upper and lower bounds was useful. These boundaries were established by including a random walk and randomly generated ontologies as lower bounds and a shortest-path solution as an upper bound. All three of them are described in the following sections.

5.2.1. Random Walk

The random walk consisted of following a random link or tracking back to the previous node at each step, where each link and the backtrack had a uniform probability. The random walk did not take already visited nodes or potential targets among the neighboring nodes into account. This comparison showed how much more information the OBDS approach provided to the navigation in comparison to a complete random behavior.

5.2.2. Randomly generated Ontologies

For comparing with a randomly generated ontology, a randomly generated ontology counterpart for every ontology used in the simulations was constructed. To this end, the number of nodes and edges was used as input for the configuration model approach of generating a random graph. The configuration model takes the number of nodes and the degree sequence of a given graph as inputs, and produces a randomly connected graph with the same number of nodes and edges as output [New03, p. 200]. As the resulting graph is not necessarily connected, it was subsequently necessary to randomly connected all graph components and then remove the number of edges created in this process from other parts of the resulting graph (without deconnecting it).

This comparison allowed to assess how much information the OBDS approach gained by taking the structure of the ontologies into account, but not yet the correct mappings to the Wikipedia articles. Furthermore, evaluating with randomly generated ontologies took the structured search behavior of decentralized search into account: Decentralized search, in the implementation used in this thesis, did not re-explore already visited nodes, could backtrack and always recognized links leading to a target node among the current node's neighbors. This gave the comparison with randomly generated ontologies a distinct advantage over the pure random walk.

5.2.3. Shortest-path solution

For the upper bound, a solution using the shortest paths from the portal to the target nodes was computed. In the single-target scenario, this meant that the shortest possible path in the graph for connecting the portal to the target node was used to simulate and evaluate the results.

For the multiple-target scenario, an exact solution would have required solving an instance of the traveling-salesman problem - which is computationally expensive, even for small input sizes. To circumvent this issue, a solution was approximated with a nearest-neighbor approach. This approach always took the shortest possible path to the currently nearest neighbor and hence required only a quadratic number of distance calculations.

This upper bound showed the (approximately) best possible solution. It is important to note that the best solution was only possible with global knowledge of the graph topology, which search agents generally do not posses in a decentralized search scenario.

5.3. Results

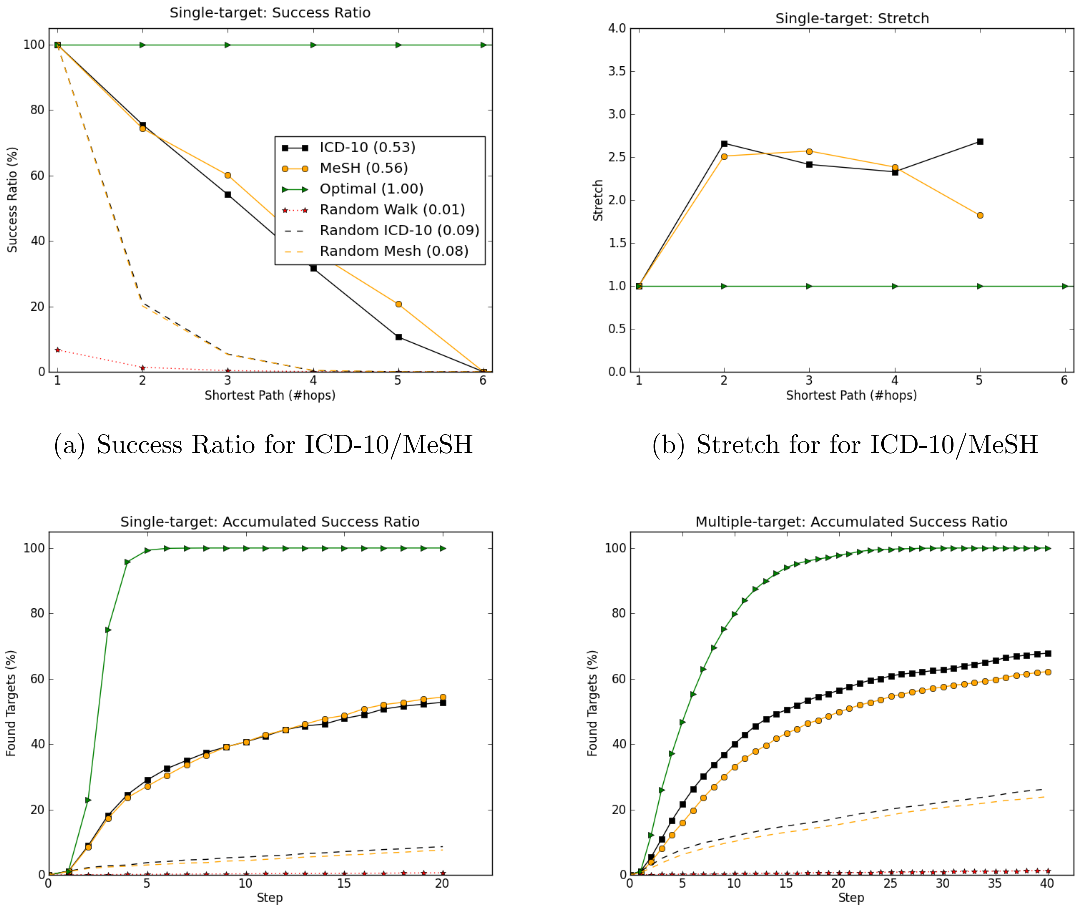
5.3.1. Evaluation Approaches

The results are presented in two different evaluation approaches.

1. Firstly, for the **domain-specific** evaluation each ontology was mapped to the maximum number of articles available on Wikipedia. Subsequently, the performance for each ontology on its *domain-specific set of articles* (that is, each ontology on a different set of articles) was evaluated.
2. Secondly, the **cross-domain** performance of several ontologies was evaluated. For this, the set of Wikipedia articles was reduced to the intersection, i.e., the *set of articles mapping to all examined ontologies*. This allowed a direct comparison of the performance of the different ontologies.

5.3.2. Domain-specific Evaluation

For the domain-specific evaluation, ICD-10 was compared to MeSH. The data for this consisted of two different sets of Wikipedia articles, namely the whole sets of $2,673$



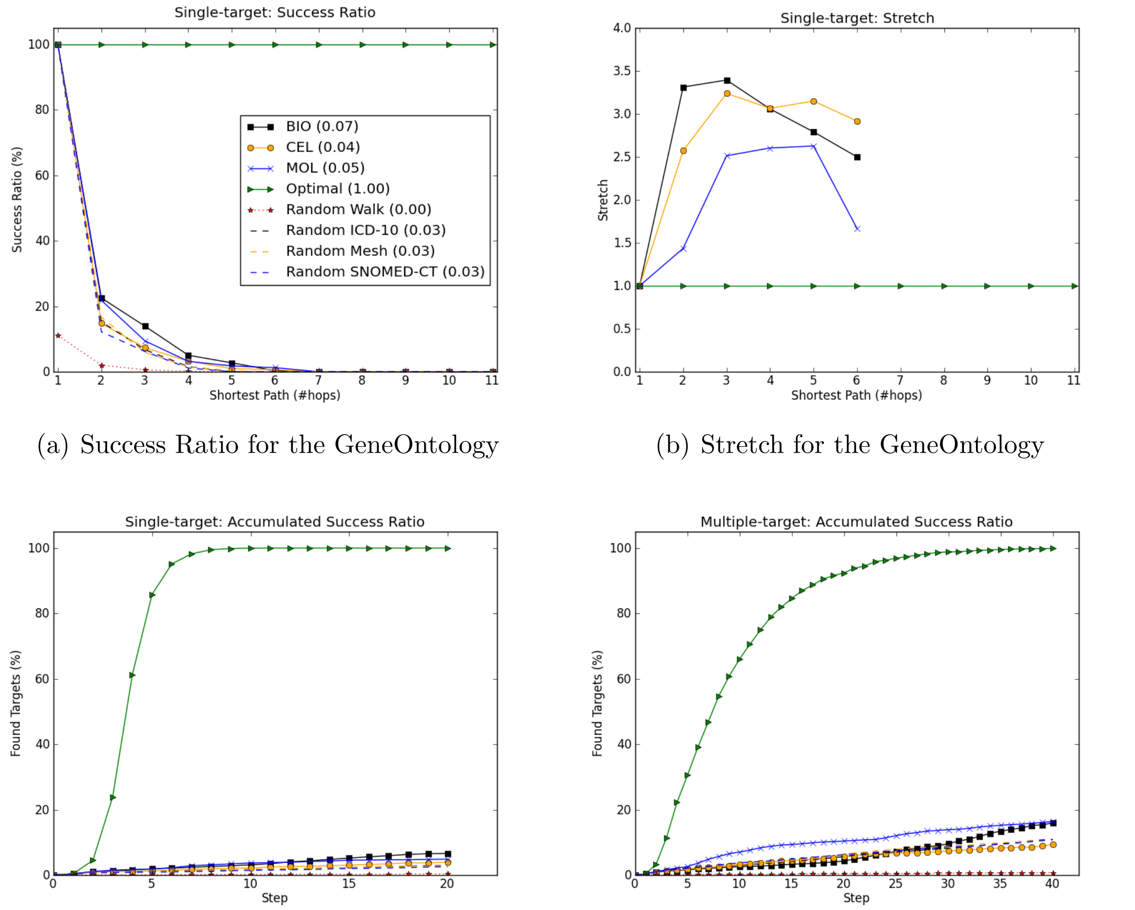
(a) Success Ratio for ICD-10/MeSH

(b) Stretch for for ICD-10/MeSH

(c) Accumulated Success Ratio (single-target) for ICD-10/MeSH

(d) Accumulated Success Ratio (multiple-target) for ICD-10/MeSH

Figure 5.1.: **Domain-specific evaluation: Success ratio, stretch and accumulated success ratio for ICD-10 and MeSH.** The columns show *stretch*, *success ratio* and *accumulated success ratio*, respectively. The numbers in parentheses display the overall values for the success ratio. The legend displayed in Subfigure a) is valid for all four Subfigures. Note that the stretch plots do not include lower bounds, as this measure can only be usefully applied to compare simulations with a similar number of found paths. The figures show that the results produced by Ontology-based Decentralized Search are noticeably better than the results for randomly generated ontologies and the random walk.



(a) Success Ratio for the GeneOntology

(b) Stretch for the GeneOntology

(c) Accumulated Success Ratio (single-target) for the GeneOntology

(d) Accumulated Success Ratio (multiple-target) for the GeneOntology

Figure 5.2.: **Cross-domain evaluation: Success ratio, stretch and accumulated success ratio for the GeneOntology.** The columns show *stretch*, *success ratio* and *accumulated success ratio*, respectively. The numbers in parentheses display the overall values for the success ratio. The legend displayed in Subfigure a) is valid for all four Subfigures. Note that the stretch plots do not include lower bounds, as this measure can only be usefully applied to compare simulations with a similar number of found paths. The figures show that the results produced by Ontology-based Decentralized Search are as least as good as the results for randomly generated ontologies.

articles mapping to ICD-10 and the $2,584$ articles mapping to MeSH. The results are displayed in Figure 5.1.

For the single-target scenario, OBDS displayed about the same performance for both ICD-10 and MeSH. The results also show that OBDS was able to guide the decentralized search towards a little over half of the targets, which was significantly above the performance of both the random walk and the randomly generated ontologies.

For the multiple-target scenario, the outcomes for both ontologies were again side by side, with ICD-10 performing slightly better (68% found targets after the maximum number of steps) than MeSH (62% found targets).

The results of this domain-specific evaluation shows that both ICD-10 and MeSH are well-suited to guide navigation on the set of their corresponding data sets. Although the ontologies serve different purposes (ICD-10 is a disease classification and MeSH is a controlled vocabulary for journal indexing), both produce the same results in the evaluation metrics concerned.

5.3.3. Cross-domain Evaluation

For the cross-domain evaluation, multiple ontologies were evaluated on the same set of Wikipedia articles. Cross-domain evaluation permitted the inspection of multiple ontologies side by side, facilitating comparisons.

The data sets used for this were

i) the set of articles mapping to all three subontologies of the GeneOntology.
ii) the set of the articles mapping to ICD-10, MeSH and SNOMED CT

Figure 5.2 shows the results for i). Comparing the success ratio and accumulated success ratio with the upper and lower bounds, it becomes apparent that all three subontologies of the GeneOntology performed only slightly better than randomly generated ontologies. Overall, the success ratios were fairly low, and the ontologies were able to guide the navigation only towards finding between 4 and 7% of the target nodes for the single-target scenario and between 10 and 18% for the multiple-target scenario. However, the ontologies did lead the OBDS to a better performance than pure random walks, illustrating that the basic algorithm of Ontology-based decentralized search contains an inherent improvement over random search behavior.

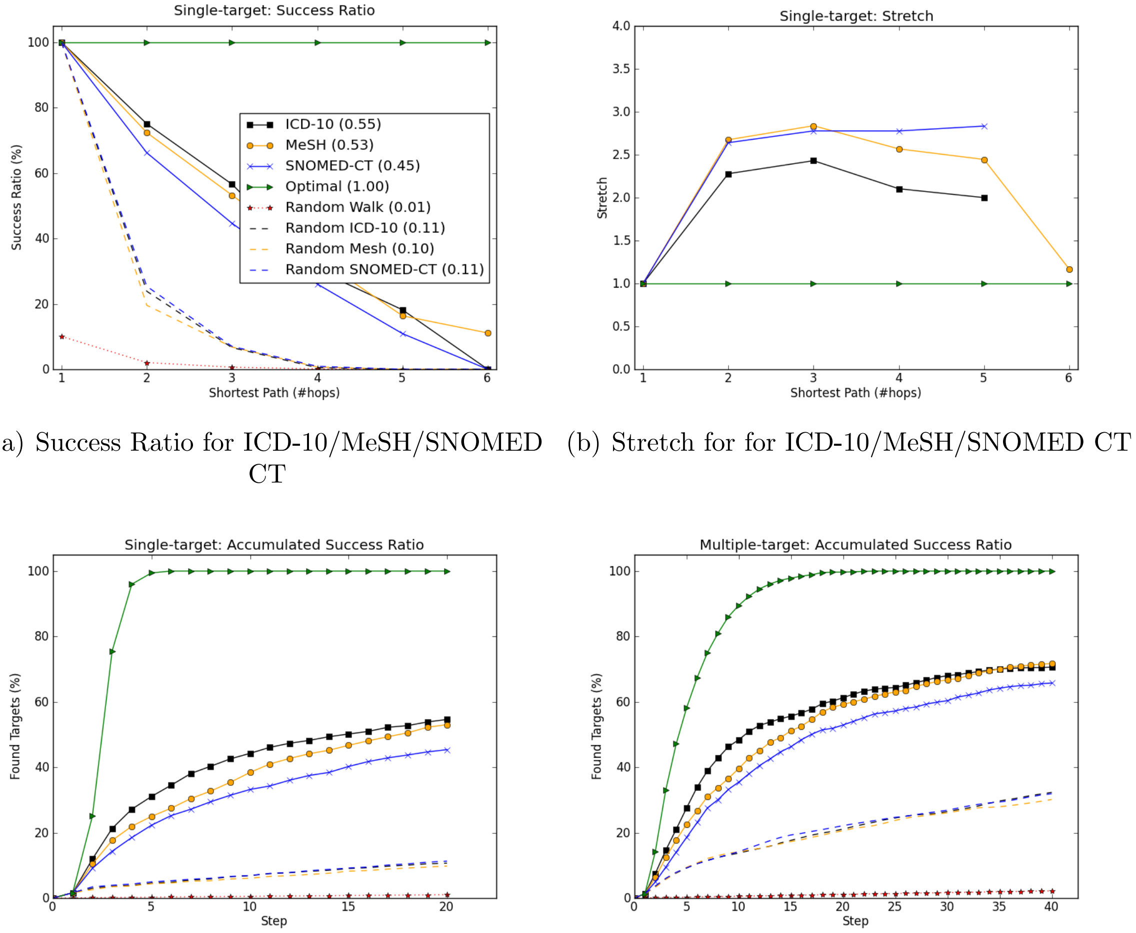
Figure 5.3 displays the results for ii). In this case, the ontologies were able to inform the navigation better than for i). The results show that the success ratios were well above both the random walk and the randomly generated ontologies, and the same is the case for the accumulated success ratios.

When comparing the performance of the ontologies, the results show that ICD-10 performed best overall, followed by MeSH and SNOMED CT for the success ratios. For the stretch, SNOMED CT fared slightly better than MeSH (with an average stretch of 2.45 resp. 2.49).

5.4. Distance distribution on the background ontologies

Another interesting aspect of the simulation results was the distance distribution on the background ontologies. For these distributions, all ontology concepts mapping to a Wikipedia article in the data set were considered, and then the pairwise distance between all of them was calculated. As such, the resulting distribution covers only a part of the distance distribution of the ontology concepts, namely the fraction of distances relevant to the results. Figure 5.4 shows the resulting distribution of these distances.

The results show that the distances were distributed more evenly and over a larger range of distances for the ICD-10/MeSH/SNOMED CT data set. This can be explained by the large number of mappings to the GeneOntology that each of the gene-related Wikipedia articles had, which increases the probability of shorter paths. However, finding these exact shortest paths might be difficult, as the simulation would need precise information from the ontology to exploit them. This might explain why the GeneOntology data set fared worse than the ICD-10/MeSH/SNOMED CT data set.



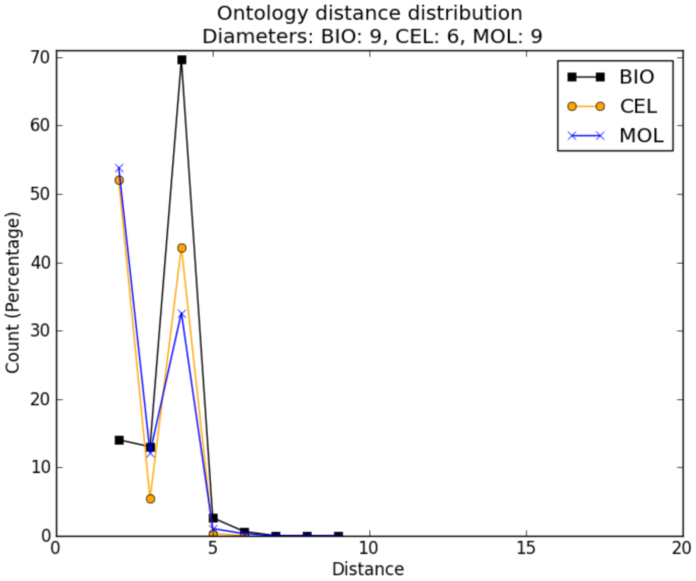
(a) Success Ratio for ICD-10/MeSH/SNOMED CT

(b) Stretch for for ICD-10/MeSH/SNOMED CT

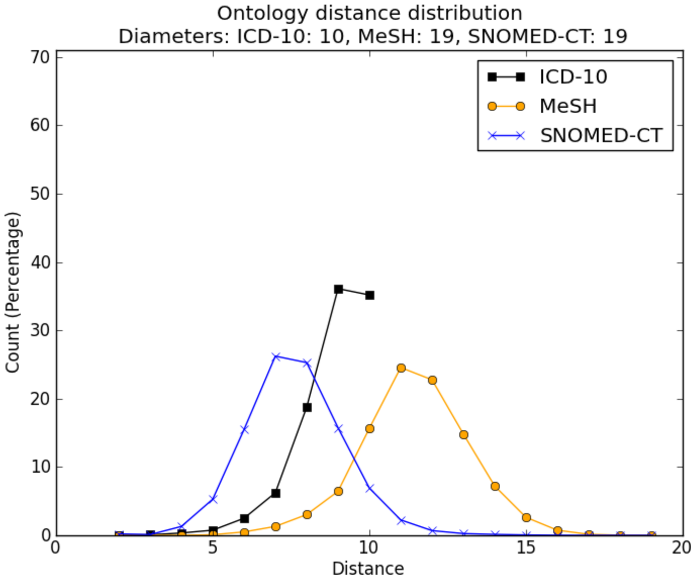
(c) Accumulated Success Ratio (single-target) for ICD-10/MeSH/SNOMED CT

(d) Accumulated Success Ratio (multiple-target) for ICD-10/MeSH/SNOMED CT

Figure 5.3.: **Cross-domain evaluation: Success ratio, stretch and accumulated success ratio for ICD-10/MeSH/SNOMED CT.** The columns show *stretch*, *success ratio* and *accumulated success ratio*, respectively. The numbers in parentheses display the overall values for the success ratio. The legend displayed in Subfigure a) is valid for all four Subfigures. Note that the stretch plots do not include lower bounds, as this measure can only be usefully applied to compare simulations with a similar number of found paths. The figures show that the results produced by Ontology-based Decentralized Search are noticeably better than the results for randomly generated ontologies.



(a) Ontology distance distribution (Gene Ontology)



(b) Ontology distance distribution (ICD-10/MeSH/SNOMED CT)

Figure 5.4.: **Distance distribution of ontology concepts that are represented in Wikipedia** The distances were calculated between all pairs of Wikipedia articles in our data sets. To this end, the distances on the corresponding ontology concepts for each pair of Wikipedia articles was computed. The results show that the distances were distributed more evenly and over a larger range of distances for the ICD-10/MeSH/SNOMED CT data set.

6. User Study

This chapter describes the results of the user study and its comparison to the simulation results. The user study was conducted in order to evaluate the simulations and compare the results to human click data. Eight participants were asked to navigate Wikipedia, modeling the scenario of navigating to find diseases.

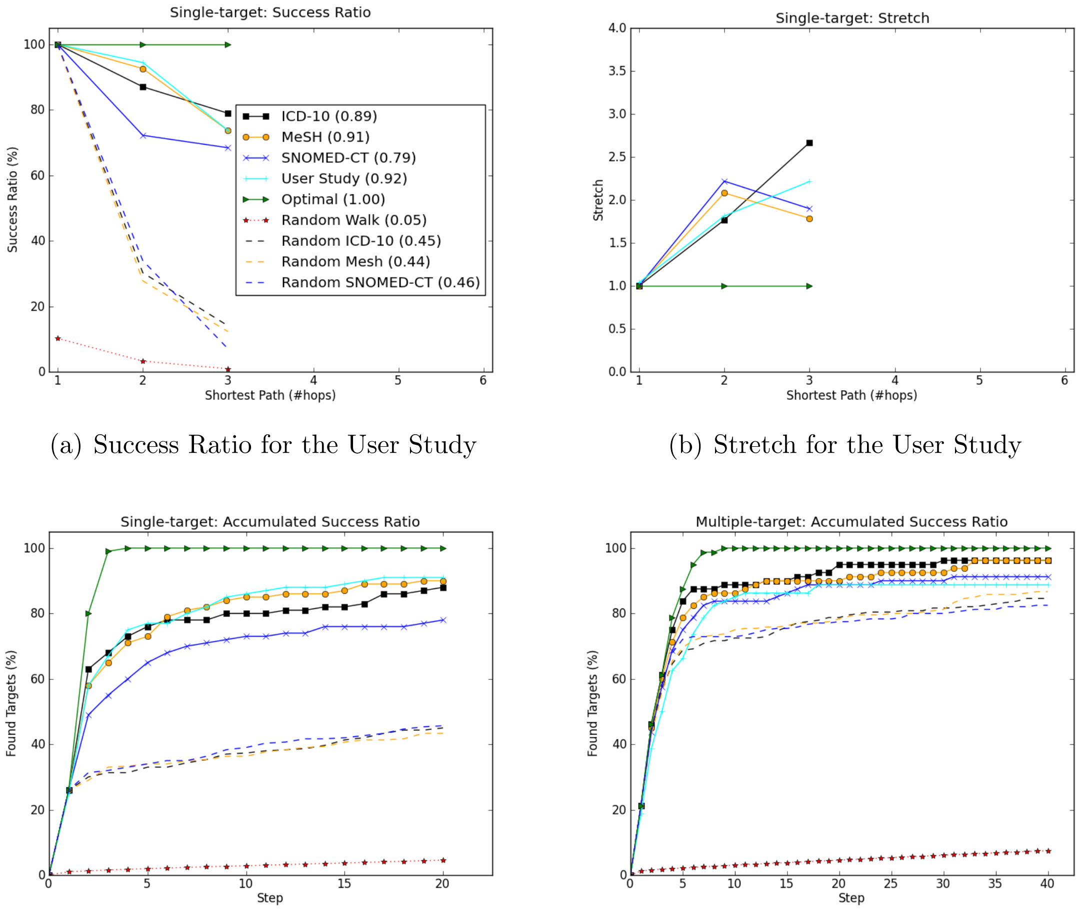
For the user study, the performance of human participants was compared with OBDS on the same data set used for the evaluation of ICD-10, MeSH and SNOMED CT. The targets were 100 manually selected target pages and 20 manually selected clusters. This limitation of targets meant that targets were a maximum distance of three hops away from the portal. The evaluations hence do not include any data points for longer shortest paths. This was a practical constraint of the study.

Figure 6.1 shows the results of the user study in comparison to the simulation results. For the single-target scenario, the overall success ratio was 92% for the user study and ranged from 79 - 91% for the ontologies. With an overall stretch of 1.74 the user study performed slightly better but still very close to the ontologies, which displayed stretches between 1.78 and 1.84. For the accumulated success ratio in the single-target scenario, the user study performed again slightly better than the ontologies.

For the multiple-target scenario, the accumulated success ratio shows that the user study fell within or just below the range of the three ontologies, performing slightly worse than the ontologies after the maximum number of 40 steps.

It is worth noting that for the multiple-target scenario after 20 steps, the users in the study did not find any more targets. This coincides with the point from where on users where given the possibility to abort a search task if they could not find the target. For the single-target scenario, this point was reached at 10 steps. However, users did still find targets after that point.

To further obtain qualitative insight into the navigation process, the following compares the produced path lengths of the user study and the simulations with regard to several aspects.



(a) Success Ratio for the User Study

(b) Stretch for the User Study

(c) Accumulated Success Ratio (single-target) for the User Study

(d) Accumulated Success Ratio (multiple target) for the User Study

Figure 6.1.: **Cross-domain evaluation: Success ratio, stretch and accumulated success ratio for the user study.** The figure shows the results for the user study, which was carried out on a subset of the targets of the ICD-10, MeSH and SNOMED CT data set). The columns show *stretch*, *success ratio* and *accumulated success ratio*, respectively. The numbers in parentheses display the overall values for the success ratio. The legend displayed in Subfigure a) is valid for all four Subfigures. Note that the stretch plots do not include lower bounds, as this measure can only be usefully applied to compare simulations with a similar number of found paths.

6.1. Comparison of path length distributions

The distribution of path lengths produced by both the user study and the simulations is a particularly interesting aspect for the validity of the simulations. This distribution can be seen in Figure 6.2.

To gain more insight into the path length distributions, the Kullback-Leibler divergence from the user study distribution to the other distributions was computed. The Kullback-Leibler divergence measures the number of additional bits needed to encode the path length distribution, if the other distribution is used in place of the original (user study) path length distribution.

For two discrete probability distributions $p(x)$ and $q(x)$ on $\mathcal{X}$, the *Kullback-Leibler divergence* or *relative entropy* is defined as [CT06, p. 19]

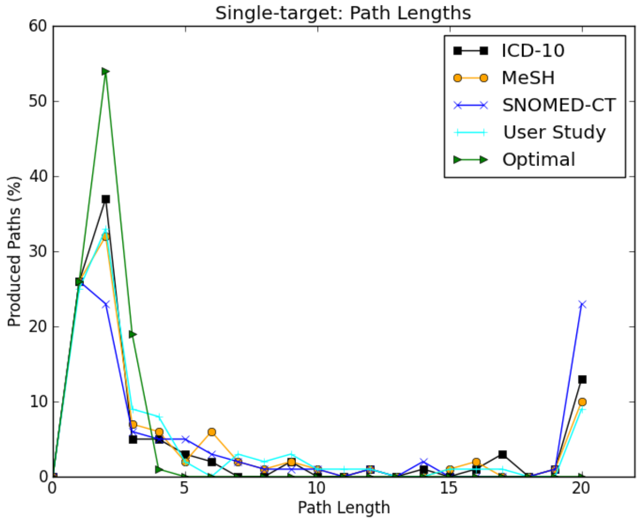
$$\begin{aligned} D(p||q) &= \sum_{x \in \mathcal{X}} p(x) \log_2 \frac{p(x)}{q(x)} \\ &= E_p \log_2 \frac{p(x)}{q(x)}. \end{aligned}$$

The Kullback-Leibler divergence is zero if and only if the probability distributions are equal. To circumvent computational problems when one of the distributions was nonzero and the other zero, a Laplace Smoothing was applied prior to calculating the values.

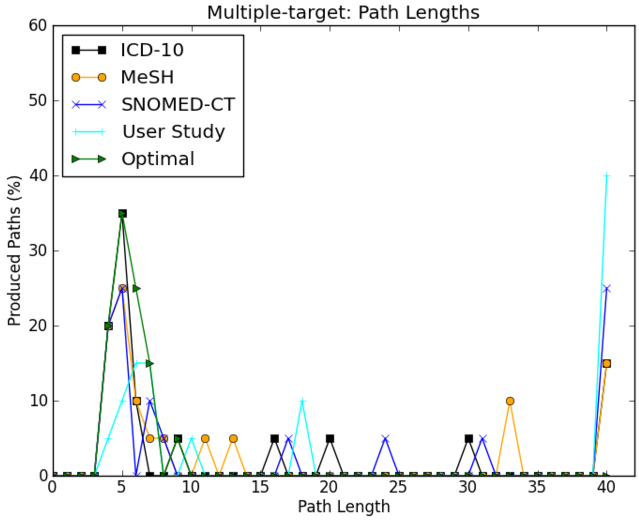
The resulting values can be seen in Table 6.1. For the single-target search scenario, it is clearly visible that only the ontologies produced path length distributions close to the user study: All three ontology path length distributions had a very small Kullback-Leibler divergence (ranging from 0.08 to 0.18 bits) to the user study.

This means that it is reasonable to replace human navigation data with data produced by Ontology-based decentralized Search and a fitting ontology (as far as produced path lengths are considered). The same cannot be said about randomly generated ontologies, the random walk or the optimal solution, which cannot be easily taken in lieu of the ontologies to obtain similar results.

For the multiple-target search scenario, this assertion cannot be made this clearly. However, the path length distribution for the multiple-target scenario was rather sparse, as the data consisted of a mere 20 search scenarios, all of which were very likely to produce a path of a different total length. This meant that a single path



(a) Path lengths produced by the simulations and the user study (single-target scenario)



(b) Path lengths produced by the simulations and the user study (multiple-target scenario)

Figure 6.2.: **Path lengths produced by the user study and the simulations.** The figures show the resulting path lengths for the single-target (a) and multiple-target (b) search scenarios. Navigation was limited at 20 resp. 40 steps, hence the high number of paths for these lengths (i.e., not all targets were found). The path distributions for the random walk and the randomly generated ontologies were left out for reasons of clarity.

accounted for five percent of the path lengths, which is also reflected in Figure 6.2 b). In subsequent research, it might be desirable to repeat the study with domain experts (thus enlarging the possible targets) or to replicate the study on a second, larger data set.

6.2. Further comparisons

In addition to the path lengths, several further aspects of the user study in comparison with the ontologies were investigated. The data for the results is displayed in Table 6.2 and Table 6.3 .

First, the visited Wikipedia pages and the found targets were inspected. To compare these, the nodes were arranged into vectors, which were then used computed cosine similarities.

The cosine similarity is a similarity measure frequently used in information retrieval. For two vectors a and b, the cosine similarity is defined as the cosine of the angle between them, or as

	ICD-10	**MeSH**	**SNOMED CT**
single-target	0.12	**0.08**	0.18
multiple-target	1.01	0.74	0.84

	Optimal	**Randomly Generated Ontology**	**Random Walk**
single-target	0.46	0.97	2.56
multiple-target	1.63	**0.55**	1.29

Table 6.1.: **Kullback-Leibler divergence for the path length distributions produced by the simulations and the user study.** The table shows the KL divergence from the user study to the ontologies and the upper and lower bounds. The KL divergence measures the number of additional bits required to encode the original distribution, if another distribution is used in its place. The randomly generated ontology column was computed using an average over the three randomly generated ontologies considered. The table shows, that the user study was more similar to the ontologies than to the base lines for the single-target scenario.

$$\text{cosine similarity}(a,b) = \frac{a \cdot b}{\|a\|\,\|b\|}.$$

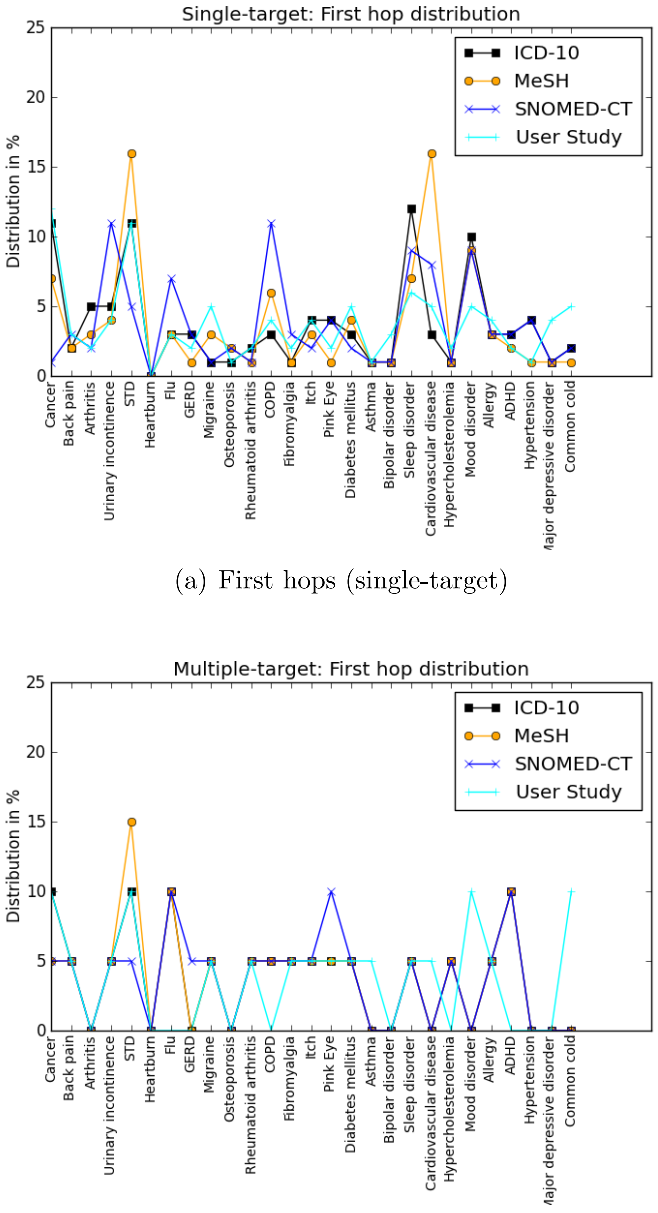
The cosine similarity is always between -1 and +1. In the case of vector entries greater or equal to zero, as in the following evaluations, the cosine similarity is always between 0 and 1 (where 1 denotes identical vectors).

Found Targets For the targets found, all three ontologies displayed high cosine similarity values. This is caused by high success ratios for the limited target set used in the user study which leads to the majority of the vectors containing ones at the same positions and reflects the results from Figure 6.1. The targets found by the user study were most similar to those found by OBDS with the support of MeSH, followed closely by ICD-10.

First Hops The term *first hop* refers to the very first click on a link on the portal at the beginning of each search scenario. For the first hops, the clicks were distributed rather evenly. A truly random distribution would see each link clicked with an expected value of 3.7%. The results showed distributions ranging from 1 to 17% and were thus fairly evenly distributed, explaining the values of the cosine similarity being close together. For the first hops, ICD-10 displayed the most similar values to the user study. The results for the first hops are displayed in graphic form in Figure 6.3. Again, due to the limited nature of the multiple-target search in the conducted user study, the outcome for the first hop distribution appears with only four values (0, 5, 10 and 15).

Backtracking and Home Button Uses In addition to calculating similarities, the average per-step probability of backtracking or clicking the home button was also further investigated.

Both the simulation and the users had access to a back button (leading to the previously visited page) and a home button (leading back to the portal) at all times. However, the computer-generated simulations used the home button only immediately after having found a target in multiple-target search. In all other cases, the best strategy given by the simulation constraints turned out to be backtracking. Furthermore, to avoid jumping back and forth from and to the portal, the portal was always set as the last available options in the case of ties (between the selection of a node, backtracking and jumping to the portal) in the decision process of the OBDS algorithm.



(a) First hops (single-target)

(b) First hops (multiple target)

Figure 6.3.: **First hops produced by the user study and the simulations.** The figures show the distribution of clicks on links pointing away from the starting portal for the single-target (a) and multiple-target (b) search scenarios. The article names are the links users and the simulation could click on, as taken from the WebMD portal described in Chapter 4.4

The user study showed different behavior from the simulations in several aspects: For single-target search, users backtracked less frequently (9% of clicks were back button clicks, versus 11-13% for the simulations) but used the home button in 2% of clicks. For the multiple-target search, users backtracked more frequently (27% versus 17-18% for the simulations) and used the home button less frequently (1% versus 2-3%).

In conclusion, backtracking was the most widely applied strategy for navigating out of dead ends and backtrack from less promising areas of the network. The intuitions that the simulations and the users would make frequents use of the home button to return from an unknown area of the network were hence not met. This was especially true for the user study.

		ICD-10	**MeSH**	**SNOMED CT**	**Optimal**	**Randomly Gen. Ont.**	**Rand. Walk**
Found targets	Single	0.93	**0.95**	0.89	**0.95**	0.78	0.72
	Multiple	**0.94**	**0.94**	0.91	**0.94**	0.90	0.67
First Hops	Single	**0.89**	0.85	0.69	0.88	0.77	0.80
	Multiple	0.64	0.62	0.56	0.68	0.64	**0.71**

Table 6.2.: **Details of the found targets and first hops of the user study in comparison to the other data sets** The table displays the cosine similarity values of the user study and the ontologies. The most similar values to the user study are displayed in bold face.The information about found targets and first hops were viewed as a vector of values, for which the angle to the vector containing the information for the user study (i.e., the cosine similarity) was calculated. For the random walk, the average over 1000 random walks for each portal-target pair was used. The randomly generated ontology column was computed using an average over the three randomly generated ontologies considered. The results confirm that what has appeared somewhat apparent from the success ratios and the stretch, i.e., that ICD-10 and MeSH displayed the most similar behavior to the user study.

		User Study	**ICD-10**	**MeSH**	**SNOMED CT**	**Optimal**	**Randomly Gen. Ont.**	**Rand. Walk**
Back	Single	0.09	0.13	**0.11**	0.13	0.00	0.26	0.07
	Multiple	0.27	0.17	**0.18**	**0.18**	0.01	0.21	0.09
Home	Single	0.02	0.00	0.00	0.00	0.00	0.01	0.00
	Multiple	0.01	0.03	**0.02**	0.03	0.00	0.05	0.00

Table 6.3.: **Details of the back and home button uses of the user study in comparison to the other data sets** The table displays the averaged per-step clicks of these buttons in the user study and the ontologies. The most similar values to the user study are displayed in bold face. For the random walk, the average over 1000 random walks for each portal-target pair was used. The randomly generated ontology column was computed using an average over the three randomly generated ontologies considered. In summary, the results confirm that what has appeared somewhat apparent from the success ratios and the stretch, i.e., that ICD-10 and MeSH displayed the most similar behavior to the user study.

7. Discussion

This thesis studied simulated user navigation behavior via decentralized search. As one of the contributions, this thesis presented ontology-based decentralized search (OBDS), a novel navigation simulation method based on decentralized search and using ontologies as background knowledge. This thesis showed that the method can be successfully applied to navigation in information networks, and demonstrated that it can be applied to the information network of Wikipedia and ontologies from the biomedical domain.

7.1. Research Questions

This thesis addressed the research question presented in the introduction (see Section 1.2) as follows.

Research Question 1 **Can ontologies contribute useful information to navigation in information networks? How is their performance in comparison to randomly generated ontologies and random walks?**

This thesis found that ontologies can indeed inform navigation in information networks. The specific performance depends on the given domain and the quality of the mappings to the information network. In the evaluations, OBDS worked well for the ICD-10, MeSH and SNOMED CT data set and less well for the GeneOntology data set. In comparison to lower bounds, OBDS always performed better than random walks and, depending on the domain, for some data sets also better than randomly generated ontologies. For the ICD-10, MeSH and SNOMED CT data set, OBDS performed substantially better than randomly generated ontologies.

Research Question 2 **Does ontology-based decentralized search (OBDS) produce valid results, i.e., are the simulated navigation paths similar to those produced by human navigation?**

This thesis addressed this question by comparing the resulting navigation paths of the simulations to a user study. The click paths produce by OBDS performed well above pure random walks. The results from the Kullback-Leibler divergence, an information-theoretic measure of distance between probability distributions, showed that the resulting path lengths from OBDS were best suited to model human behavior for the single-target search scenario.

Research Question 3 **When using OBDS, what ontology is besteds suited to produce human-like navigation results?**

From the results, it shows that ICD-10 and MeSH are best suited to be used as a replacement for human behavior when navigating in an information network with the specific settings and the algorithmic navigation behavior presented in the setup section of this thesis. However, the overall differences between the ontologies were not very strong, and it is subject of ongoing research to further identify differences in the performances of different ontologies. SNOMED CT as well as all subontologies of the Gene Ontology turned out to be less suited to represent human navigation behavior.

7.2. Further Comments

Gene Ontology The significantly lower performance of the GeneOntology data set was very noticeable in the results. By closely inspecting the data, it turned out that the Wikipedia articles annotated by the GeneOntology were characterized by different properties than the other data sets: They contained a large number of stubs (very short articles) and orphans (articles not linked to by any other article). In addition, the Wikipedia article network was sparser than the other data sets, i.e., it contained fewer links (the density was 0.0013, compared to 0.0057 for the ICD-10/MeSH/SNOMED CT data set). Furthermore, each Wikipedia article referenced a greater number (up to fifty, in comparison to usually one or two for the other ontologies) of ontology concepts. It was hence significantly harder to discover a correct link by making an educated guess. This made navigating the graph more difficult.

User Study In comparison to the ontologies' performance, participants in the user study performed better for single-target search and worse for multiple-target search. This is also influenced by the fact that users aborted 30% of their multiple-target navigation tasks before having found all of the targets, while the simulations ran for whole number of possible steps (40).

Building User Models Using different ontologies as background knowledge, the results presented in this thesis could help researchers and engineers build and evaluate user interfaces with different user types. The ontologies compared in the results were rather similar and mostly shared a domain. In future work, it could be possible to use ontologies that do not cover the entire domain, modeling specialist users, or combining ontologies to form a complete coverage of the domain. Another idea might be to prune the ontologies at a certain depth, modeling broad generalist knowledge that does not extend beyond a certain depth.

Action Selection The simulations in the form presented in this thesis followed a deterministic greedy action selection model, in that it always selected the most profitable link according to the background knowledge. Related research has shown that users might rather use epsilon-greedy action selection mechanisms with dynamically changing epsilon [HSGS13]. In follow-up research, this thesis could be extended with stochastic action selection mechanism such as epsilon-greedy. This would also lead to another potentially crucial aspect of the present simulations, namely the need to evaluate games multiple times with potentially varying results. One could expect that these adaptations would bring the simulations even closer to modeling human behavior in information networks. However, the task of the concrete adaptation of these changes is left to future research.

Influence of ICD-10 The International Classification of Diseases (ICD-10) has found widespread use and has, without doubt, also influenced and inspired Wikipedia editors. On Wikipedia, disease articles are almost in all cases indexed by ICD-10 as the first entry in the articles infobox. Furthermore, the category system for the disease articles of the English Wikipedia is organized according to ICD-10. These two facts and the wide use of ICD-10 have possibly also influenced the link creation behavior on the encyclopedia as well as the general knowledge of the test subjects. This might be an explanation of why ICD-10 seems to be best suited to model human navigation behavior in the case study presented in this thesis.

8. Conclusions and Outlook

8.1. Conclusions

This work presented a novel, ontology-based method (Ontology-Based Decentralized Search) for simulating human behavior in information networks such as Wikipedia. The results provide technical answers to several questions regarding the use of ontologies in decentralized search: This thesis has not only presented a method to integrate ontological background knowledge into decentralized search, but also found that ontologies can serve as an *efficient* background knowledge to support navigation. With appropriate ontologies and Wikipedia link networks, the simulations using OBDS i) found targets more efficiently than two baseline approaches (random walks or randomly generated ontologies) and ii) produced navigational paths that are more similar to actual human navigational paths than to the baseline approaches.

While the human subject study was limited in terms of size, the results reported in this thesis are encouraging in several ways. First, the method opens up ways to explore the effects of assuming different kinds of background knowledge of users in a navigation task. For example, swapping different kinds of ontologies in future work could allow to explore their impact on the efficiency of decentralized search in information networks. Second, the results can be seen as additional corroboration that ontologies indeed capture useful knowledge about a domain. In some of the experiments, the investigated medical ontologies were able to outperform baseline approaches significantly.

Summarizing, the findings are relevant for researchers interested in new applications for ontologies or interested in modeling navigation in information networks using ontologies as background knowledge.

8.2. Future Work

The user study presented in this thesis was limited in that it was restricted to a subset of target nodes because of the requirement to be familiar to test subjects without a medical background. Since the simulation behavior for these targets was very close to the test subjects, it can be hypothesized that the user behavior for the whole set of targets is likewise similar. It is up to to future research to show more details of the comparison of human users and decentralized search.

The chosen portals (based on WebMD.com and the Wikipedia GeneWiki portal) undoubtedly influenced the navigation results. It is up to future work to compare different portals and shine a light on possible differences.

Another important question are the potential differences when applying Ontology-based Decentralized Search to information networks outside the biomedical domain. In the biomedical domain, ontologies have been adapted more frequently than in other disciplines [NT08] and constitute an important factor in research [B$^+$08]. Other domains have not seen such eager adoption rates. In related research by Helic, Strohmaier et al. (e.g., [HSGS13]), Hierarchical Decentralized Search was applied with hierarchies generated from network features (such as node in- and outdegree), a process that has led to promising results and might be interesting to further pursue in following work.

The idea to navigate to one single predefined target might seem somewhat artificial in the case of user behavior concerning explorative tasks. However, one idea to improve on this might be calculate the TF-IDF features of the target node beforehand and subsequently navigate until a page (or a number of pages) similar enough to the TF-IDF features has been found (which does not need to be the predefined target page). This could model the case of users exploring areas of the network.

Other potential research questions might include the limitation of visible links to links in the upper part of Wikipedia articles, comparing the results on non-English editions of the encyclopedia and the study of different methods of extracting background knowledge from the actual network used for navigation.

Appendix

Appendix A.

Wikipedia templates used for extraction

Infobox Symptom
Infobox Disease
Signsymptom Infobox

Table A.1.: Infobox templates used for the extraction of ICD-10 and MeSH Wikipedia articles

PBB

Table A.2.: Templates used for the extraction of GO Wikipedia articles

Appendix B.

User Study Briefing

Imagine you wake up one day and discover an itchy, bright red rash on your forearms. What do you do? You fire up Wikipedia, look at the article of itch or rash, and click around a couple of articles to see what it could be.

In this task we're trying to evaluate this scenario. Starting from an entry portal, your task is to find one or several target articles by only following links in the article texts. In order to define a target, we explicitly tell you what article to look for - but we pretend you don't know that beforehand and so you can't use the search function to directly jump to it.

You may:

- jump back to the portal at any time by clicking the link or the Wikipedia logo in the upper left corner
- use the in-page search (CTRL + F) (occurrences of target article names are displayed in parenthesis)
- use the back and forward button of the browser

But please make sure to observe the following:

- Please do not modify the address bar.
- Please do not visit any external web sites.
- Do not jump several steps forward or backward at once

You will be given 15 tasks, consisting either of finding a single target article (e.g., ”Excessive daytime sleepiness”), or a set of semantically related targets articles (e.g., ”Excessive daytime sleepiness, Non-24-hour sleep-wake syndrome, Irregular sleep–wake rhythm”). Please note that links might not necessarily have the same title as the target page (e.g., a link entitled “Autistic disorder” can link to the “Autism” page). If you're desperate and unable to find the target page, you may click an abort link (that appears after you've tried for a while).

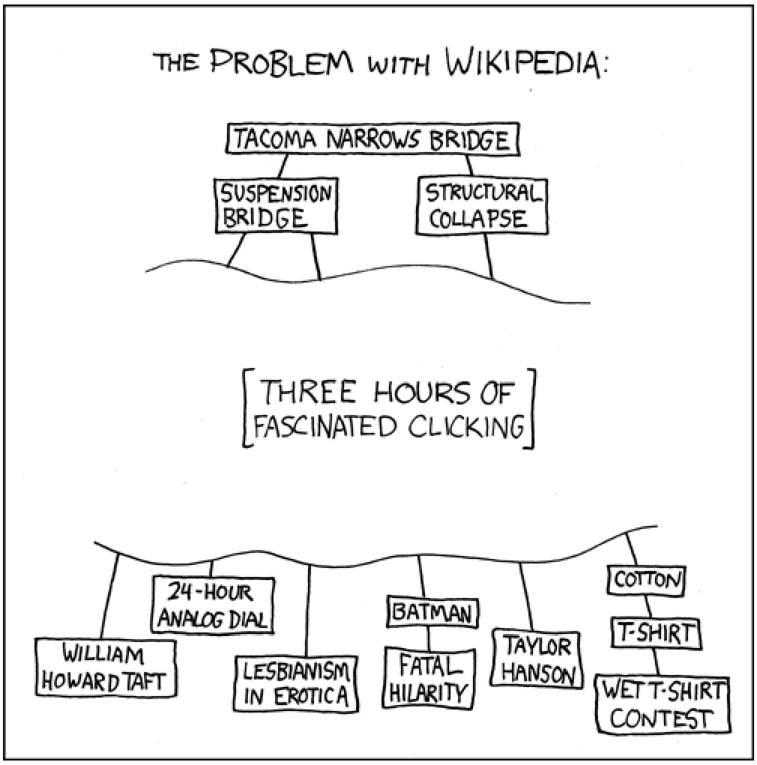


Figure B.1.: **The problem with Wikipedia** (CC BY-NC 2.5 from `http://xkcd.com/214/`) This figure was included on the sheet of instructions presented to participants as a motivation for the study.

Appendix C.

Clusters and targets used in the user study

Allergy	Asthma	Food allergy	Peanut allergy
Attention deficit hyperactivity disorder	Panic attack	Bipolar disorder	Schizophrenia
Obsessive–compulsive disorder	Panic disorder	Social anxiety disorder	Separation anxiety disorder
Arthritis	Rheumatoid arthritis	Osteoarthritis	Gout
Back Pain	Low back pain	Osteoporosis	Whiplash (medicine)
Cancer	Myeloid leukemia	Leukemia	Uterine cancer
Ovarian cancer	Thyroid cancer	Stomach cancer	Testicular cancer
Common Cold	Influenza	Pneumonia	Viral pneumonia
Chronic obstructive pulmonary disease	Chronic bronchitis	Asthma	Bronchitis
Common Cold	Influenza	Tuberculosis	Sinusitis
Diabetes mellitus	Prediabetes	Diabetes mellitus type 2	Hypoglycemia
Conjunctivitis	Glaucoma	Myopia	Retinal detachment
Fibromyalgia	Chronic fatigue syndrome	Irritable bowel syndrome	Sleep disorder
Hypertension	Hypotension	Hypercholesterolemia	Cardiovascular disease
Urinary incontinence	Urinary tract infection	Urinary bladder disease	Urinary retention
Migraine	Cluster headache	Tension headache	Vascular headache
Sexually transmitted disease	Chlamydia infection	Erectile dysfunction	Herpes genitalis
Hepatitis	Hepatitis A	Hepatitis B	Hepatitis C
Itch	Insect bites and stings	Sunburn	Eczema
Sleep disorder	Circadian rhythm sleep disorder	Insomnia	Non-24-hour sleep-wake syndrome

Table C.1.: **Target clusters used in the user study**

Bronchitis	Pneumonia
Bipolar disorder	Eczema
Myeloid leukemia	Panic attack
Thyroid cancer	Osteoarthritis
Cancer	Circadian rhythm sleep disorder
Schizophrenia	Boil
Prediabetes	Sinusitis
Stomach cancer	Mood disorder
Testicular cancer	Whiplash (medicine)
Erectile dysfunction	Angina pectoris
Posttraumatic stress disorder	Diabetes mellitus
Major depressive disorder	Syphilis
Retinal detachment	Common Cold
Peanut allergy	Obsessive–compulsive disorder
Glaucoma	Urinary tract infection
AIDS dementia complex	Hypoglycemia
Bone disease	Bladder cancer
AIDS	Insect bites and stings
Tuberculosis	Myopia
Rheumatoid arthritis	Leukemia
Gastroesophageal reflux disease	Osteoporosis
Ovarian cancer	Urinary incontinence
Diabetes mellitus type 2	Allergy
Back Pain	Delayed sleep phase syndrome
Hepatitis B	Chronic obstructive pulmonary disease
Chronic bronchitis	Hypotension
Gout	Stroke
Urinary bladder disease	Hypertension
Urinary retention	Migraine
Measles	Asthma
Sleep disorder	Heart failure
Hepatitis A	Vascular headache
Food allergy	Non-24-hour sleep-wake syndrome
Cluster headache	Sexually transmitted disease
Chlamydia infection	Schizoid personality disorder
Tension headache	Influenza
Low back pain	Conjunctivitis
Social anxiety disorder	Hepatitis
Esophageal cancer	Chronic fatigue syndrome

Fibromyalgia	Arthritis
Hepatitis C	Sunburn
Alcoholism	Herpes genitalis
Separation anxiety disorder	Genital wart
Insomnia	Panic disorder
Irritable bowel syndrome	Cardiac arrest
Viral pneumonia	Cardiovascular disease
Multiple sclerosis	Uterine cancer
Obsessive–compulsive personality disorder	Hypercholesterolemia
Itch	Hypersomnia
Attention deficit hyperactivity disorder	Lung cancer

Table C.2.: **Single targets used in the user study**

Bibliography

[AA04] Lada A. Adamic and Eytan Adar, *How to search a social network*, November 2004.

[AH09] Grigoris Antoniou and Frank van Harmelen, *Web ontology language: Owl*, Handbook on ontologies (2009), 91–110.

[Ash00] M. Ashburner, *Gene ontology: Tool for the unification of biology*, Nature Genetics **25** (2000), 25–29.

[B+08] Olivier Bodenreider et al., *Biomedical ontologies in action: role in knowledge management, data integration and decision support*, Yearb Med Inform **67** (2008), 79.

[BBG12] Djallel Bouneffouf, Amel Bouzeghoub, and Alda Lopes Gançarski, *A contextual-bandit algorithm for mobile context-aware recommender system*, Neural Information Processing, Springer, 2012, pp. 324–331.

[BBR+11] L. Backstrom, P. Boldi, M. Rosa, J. Ugander, and S. Vigna, *Four degrees of separation*, arXiv preprint arXiv:1111.4570 (2011).

[BGM] Janez Brank, Marko Grobelnik, and Dunja Mladenić, *A survey of ontology evaluation techniques*, Proc. of 8th Int. multi-conf. Information Society.

[BLHL+01] Tim Berners-Lee, James Hendler, Ora Lassila, et al., *The semantic web*, Scientific american **284** (2001), no. 5, 28–37.

[CT06] Thomas M Cover and Joy A Thomas, *Elements of information theory*, Wiley-interscience, 2006.

[DFVH03] John Davies, Dieter Fensel, and Frank Van Harmelen, *Towards the semantic web*, Wiley Online Library, 2003.

[DMW03] Peter Sheridan Dodds, Roby Muhamad, and Duncan J Watts, *An experimental study of search in global social networks.*, Science **301** (2003), no. 5634, 827–829.

[Eri79] Bonnie H Erickson, *Some problems of inference from chain data*, Sociological methodology **10** (1979), no. 1, 276–302.

[FE07] Jennifer Fang and Joerg Evermann, *Evaluating ontologies: Towards a cognitive measure of quality.*, EDOCW, IEEE Computer Society, 2007, pp. 109–116.

[FVHH$^+$01] Dieter Fensel, Frank Van Harmelen, Ian Horrocks, Deborah L McGuinness, and Peter F Patel-Schneider, *Oil: An ontology infrastructure for the semantic web*, Intelligent Systems, IEEE **16** (2001), no. 2, 38–45.

[GCFG10] David F. Gleich, Paul G. Constantine, Abraham D. Flaxman, and Asela Gunawardana, *Tracking the random surfer: empirically measured teleportation parameters in pagerank*, Proceedings of the 19th international conference on World wide web (New York, NY, USA), WWW '10, ACM, 2010, pp. 381–390.

[GN87] M.R. Genesereth and N.J. Nilsson, *Logical foundations of artificial intelligence*, vol. 9, Morgan Kaufmann Los Altos, CA, 1987.

[HKG$^+$12] Denis Helic, Christian Körner, Michael Granitzer, Markus Strohmaier, and Christoph Trattner, *Navigational efficiency of broad vs. narrow folksonomies*, Proceedings of the 23nd ACM conference on Hypertext and hypermedia, 2012.

[HS11] D. Helic and M. Strohmaier, *Building directories for social tagging systems*, 20th ACM Conference on Information and Knowledge Management (CIKM 2011), Glasgow, UK, 2011.

[HSGS13] Denis Helic, Markus Strohmaier, Michael Granitzer, and Reinhold Scherer, *Models of human navigation in information networks based on decentralized search*, Proceedings of the 24th ACM conference on Hypertext and social media, 2013.

[ICD12] *International classification of diseases, revision 10*, http://www.who.int/classifications/icd/en, 2012.

[Kle00] Jon Kleinberg, *The small-world phenomenon: an algorithm perspective*, Proceedings of the thirty-second annual ACM symposium on Theory of computing (New York, NY, USA), STOC '00, ACM, 2000, pp. 163–170.

[Kle01] Jon M. Kleinberg, *Small-world phenomena and the dynamics of information.*, NIPS, 2001, pp. 431–438.

[KPK$^+$10] D. Krioukov, F. Papadopoulos, M. Kitsak, A. Vahdat, and M. Boguñá, *Hyperbolic Geometry of Complex Networks*, Physical Review E **82** (2010), no. 036106.

[LTSM03] Baker L, Wagner TH, Singer S, and Bundorf M, *Use of the internet and e-mail for health care information: Results from a national survey*, JAMA **289** (2003), no. 18, 2400–2406.

[LZHH03] Zupeng Li, Xiubin Zhao, Daoyin Huang, and Jianhua Huang, *An improved network broadcasting method based on gnutella network.*, GCC (2) (Minglu Li, Xian-He Sun, Qianni Deng, and Jun Ni, eds.), Lecture Notes in Computer Science, vol. 3033, Springer, 2003, pp. 404–407.

[Mar06] Gary Marchionini, *Exploratory search: from finding to understanding*, Commun. ACM **49** (2006), no. 4, 41–46.

[MCS$^+$11] V Mohanraj, M Chandrasekaran, J Senthilkumar, S Arumugam, and Y Suresh, *Ontology driven bee's foraging approach based self adaptive online recommendation system*, Journal of Systems and Software (2011).

[MeS12] *Medical subject headings*, http://www.nlm.nih.gov/mesh/, 2012.

[Mil67] Stanley Milgram, *The small world problem*, Psychology Today **1** (1967), no. 1, 61–67.

[MTC$^+$12] Gengxin Miao, Shu Tao, Winnie Cheng, Randy Moulic, Louise E. Moser, David Lo, and Xifeng Yan, *Understanding task-driven information flow in collaborative networks*, Proceedings of the 21st international conference on World Wide Web (New York, NY, USA), WWW '12, ACM, 2012, pp. 849–858.

[New03] M. E. J. Newman, *The structure and function of complex networks*, SIAM Review **45** (2003), no. 2, 167–256.

[NM$^+$01] N.F. Noy, D.L. McGuinness, et al., *Ontology development 101: A guide to creating your first ontology*, 2001.

[NT08] Natalya F Noy and Tania Tudorache, *Collaborative ontology development on the (semantic) web*, AAAI Spring Symposium on Semantic Web and Knowledge Engineering (SWKE), Stanford, CA, 2008.

[PBMW98] L. Page, S. Brin, R. Motwani, and T. Winograd, *The pagerank citation ranking: Bringing order to the web*, Proceedings of the 7th International World Wide Web Conference (Brisbane, Australia), 1998, pp. 161–172.

[PH99] Mike Papazoglou and Jeroen Hoppenbrouwers, *Knowledge navigation in networked digital libraries*, Knowledge Acquisition, Modeling and Management (1999), 13–32.

[Pid03] W. Pidcock, *What are the differences between a vocabulary, a taxonomy, a thesaurus, an ontology, and a meta-model?*

[Pir07] Peter Pirolli, *Information foraging theory: Adaptive interaction with information*, Oxford University Press, 2007.

[PS00] C. Price and K. Spackman, *Snomed clinical terms*, BJHC&IM-British Journal of Healthcare Computing & Information Management **17** (2000), no. 3, 27–31.

[PVG$^+$11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *Scikit-learn: Machine Learning in Python* , Journal of Machine Learning Research **12** (2011), 2825–2830.

[RKA$^+$08] Menaka Rajapakse, Rajaraman Kanagasabai, Wee Tiong Ang, Anitha Veeramani, Mark J Schreiber, and Christopher JO Baker, *Ontology-centric integration and navigation of the dengue literature*, Journal of biomedical informatics **41** (2008), no. 5, 806–815.

[RSN08] D.L. Rubin, N.H. Shah, and N.F. Noy, *Biomedical ontologies: a functional perspective*, Briefings in bioinformatics **9** (2008), no. 1, 75–90.

[SBF98] Rudi Studer, V. Richard Benjamins, and Dieter Fensel, *Knowledge engineering: Principles and methods*, Data and Knowledge Engineering **25** (1998), no. 1-2, 161–197.

[Sch09] Sebastian Schnettler, *A small world on feet of clay? a comparison of empirical small-world studies against best-practice criteria*, Social Networks **31** (2009), no. 3, 179–189.

[SHB$^+$12] Markus Strohmaier, Denis Helic, Dominik Benz, Christian Körner, and Roman Kern, *Evaluation of folksonomy induction algorithms*, ACM Trans. Intell. Syst. Technol. **3** (2012), no. 4, 74:1–74:22.

[Smi08] B. Smith, *Ontology*, The Blackwell guide to the philosophy of computing and information (2008), 153–166.

[sna12] *Stanford network analysis project*, http://snap.stanford.edu, 2012.

[Stu09] Heiner Stuckenschmidt, *Ontologien: Konzepte, technologien und anwendungen*, Springer, 2009.

[TMTM69] Jeffrey Travers, Stanley Milgram, Jeffrey Travers, and Stanley Milgram, *An experimental study of the small world problem*, Sociometry **32** (1969), 425–443.

[TSHS12] Christoph Trattner, Philipp Singer, Denis Helic, and Markus Strohmaier, *Exploring the differences and similarities between hierarchical decentralized search and human navigation in information networks*, I-KNOW, 2012, p. 14.

[UG+96] Mike Uschold, Michael Gruninger, et al., *Ontologies: Principles, methods and applications*, Knowledge engineering review **11** (1996), no. 2, 93–136.

[VDMF10] Jose Renato Villela Dantas and Pedro Porfirio Muniz Farias, *Conceptual navigation in knowledge management environments using navcon*, Information processing & management **46** (2010), no. 4, 413–425.

[w3c04] *Owl web ontology language overview*, http://www.w3.org/TR/owl-features/, 2004.

[WL12a] Robert West and Jure Leskovec, *Automatic versus human navigation in information networks*, ICWSM, 2012.

[WL12b] Robert West and Jure Leskovec, *Human wayfinding in information networks*, Proceedings of the 21st international conference on World Wide Web, ACM, 2012, pp. 619–628.

[WNS+11] Patricia L. Whetzel, Natalya Fridman Noy, Nigam H. Shah, Paul R. Alexander, Csongor Nyulas, Tania Tudorache, and Mark A. Musen, *Bioportal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications*, Nucleic Acids Research **39** (2011), no. Web-Server-Issue, 541–545.

[WPP09] R. West, J. Pineau, and D. Precup, *Wikispeedia: An online game for inferring semantic distances between concepts*, Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI), 2009.

[WS98] Duncan J. Watts and Steven H. Strogatz, *Collective dynamics of small-world networks*, Nature **393** (1998), no. 6684, 440–442.

[Zäc03] S. Zächerl, *Semantic web - rdf daml + oil*, GRIN Verlag, 2003.