Patrick Kasper, BSc

New Features for a Better Understanding of Human Navigation on the Internet

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Software Development and Business Management

submitted to

Graz University of Technology

Knowledge Technologies Institute
Head: Assoc.Prof. Dipl.-Ing. Dr.techn. Denis Helic

Supervisor: Assoc.Prof. Dipl.-Ing. Dr.techn. Denis Helic

Graz, March 2015

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

If we have data, let’s look at data.
If all we have are opinions, let’s go with mine.

Jim Barksdale

Abstract

Understanding human navigational behaviour on the Internet is a key to further improve the Web. Attempting to find a specific piece of information involves visiting multiple pages and clicking on many links. Depending on the underlying network and the structure of the pages this can be a complex and often frustrating task. A usual analysis is based upon click-trails, a list of which pages where visited by the users and when, but data on how a user navigated within a single page is lacking. I propose including data on active human input, such as the positioning and movement of the mouse cursor and information about the browser window size and scroll offset to these kind of user studies. In this thesis I present two contributions. First, a tracking system capable of extracting and recording these new analytical features was developed and implemented and second the value of the new Features is evaluated. Using this system a study was performed where users had to solve missions similar to *Wikispeedia* or *The Wiki Game*. These missions represent a human computation game based on Wikipedia. The data is recorded as users try to navigate from a given start to a clearly defined target page only using hyperlinks. We find that users preferably click on links immediately present when a page is loaded but reading more content of a page results in a higher probability of a successful click (bringing the user closer to the target). Staying on a page for over two minutes increases the chance for unsuccessful clicks. During the navigation *hub-highways*, high degree nodes linking to other high degree nodes, are being used, confirming with previous research. These findings outline the capabilities of adding input-data to recorded click-trails and serve as foundation for further research on the subject and help to build a more complete model of human navigational behaviour.

Das Verstehen des menschlichen Verhaltens im Internet ist unabdinglich für eine Weiterentwicklung des World Wide Web. Die Suche nach einer konkreten Information impliziert den Besuch von mehreren Seiten und das Klicken auf eine Vielzahl von Links. Abhängig vom darunterliegenden Netzwerk kann dies eine große Herausforderung und oft auch frustrierend sein. Übliche Analysen von Klick-Pfaden unterschiedlicher Benutzer/-Innen listen wann welche Seite besucht wurde. Informationen hingegen wie auf der einzelnen Seite navigiert wurde fehlt jedoch. Ich schlage daher vor, aktive Benutzereingaben, wie die Mausposition und den Status des Browserfensters, für Analysen in die klassischen Klick-Pfade zu inkludieren. In dieser Masterarbeit präsentiere ich zweierlei Kontributionen. Als Erstes ein Tracking-System implementiert welches in der Lage ist, diese neuen Merkmale zu extrahieren und zu speichern, und zweitens werden diese neu erhaltenen Daten auf ihre Wertigkeit überprüft. Mit dem Tracking-System wurde eine Studie durchgeführt, bei der Benutzer vor die Aufgabe gestellt wurden, unterschiedliche Missionen zu absolvieren. Diese einzelnen Missionen sind ähnlich bekannter Spiele wie *Wikispeedia* oder *The Wiki Game*, bei denen Spieler/-Innen von einer zufälligen Startseite zu einem ebenfalls zufälligen Ziel navigieren müssen. Zum Navigieren selbst sind nur Links erlaubt. Es stellt sich heraus, dass durchschnittliche Benutzer vorwiegend auf Links klicken, welche direkt beim Laden der Seite sichtbar waren. Ebenso war jedoch erkennbar, dass das Lesen größerer Teile der Seite sich positiv auswirkt und die Wahrscheinlichkeit für einen guten Klick erhöht, der die Distanz zum Ziel verringert. Ein Verweilen von über zwei Minuten hatte jedoch das gegensätzliche Resultat. Im Rahmen der Missionen navigieren die Benutzer/-Innen gerne über jene Knoten, die viele Verbindungen zu anderen Knoten haben (sogenannte Hubs), welches sich mit vorhergehender Forschung deckt. Die erbrachten Erkenntnisse zeigen das Potenzial dieser neuen Daten und bilden das Fundament für weitere Forschung und Helfen ein genaueres Modell für das menschliche Verhalten zu erstellen.

Contents

List of Figures

List of Tables

1 Introduction

The Internet has become an ever-present companion to most humans in their everyday life and using the Internet for information retrieval is one of the most common tasks. Hence, understanding how people navigate the Internet has become important for many entities. Science wants to better understand human behaviour, website owners want to provide the best possible content and advertisers want to tailor their adverts for the individual users.

Tracking the movements of Internet users is nothing new but up to now was always limited by technology. Passively tracking the visited websites does provide insight into a user's interests and reveals popular sites in general but is unable to reconstruct the causes for each individual click. A different approach, Remote User Testing, tries to tackle these issues by recording inputs such as mouse movements but relies on specific software to be installed on a user's device and focuses on improving the tested site instead of understanding the user.

In order to find answers for actions performed by humans browsing the Internet these two aspects have to be combined and their flaws removed. The passive tracking that records visited sites and timestamps for these visits needs to be enriched with active data. Data about the direct inputs made by the user. Where do people click? How much do they read? What can they actually see? All these questions can be answered by extracting new features directly inside the user's web browser since they are powerful enough to perform the tracking in JavaScript without having a negative impact on the overall browsing experience.

This thesis tackles these problems and tries to answer the research questions in section 1.1. It can be broken down into two sections. Firstly, the tracking software (chapter 2) and the newly introduced features (chapter 3) will be explained. As a second part a user study (chapter 5) was performed and the results analysed (chapter 6). Finally a conclusion is drawn and the final chapter ()7) outlines future work based on the results and any issues occurring during the research.

My work resulted in two contributions. Firstly, it is shown that information about the mouse cursor and the area visible to the user can estimate the performance whilst navigating a Wikipedia network and secondly I provide a tracking software capable of recording these features.

1.1 Research Questions

To address the aforementioned problems and improve the understanding of human navigation behaviour on the Internet this thesis aims to answer the following research questions:

RQ 1: Better Features
How can we obtain additional, better navigation features that provide more insight into a user's browsing behaviour?
Research on human navigation on the Internet relies on data tracked by various sources. But click traces only provide so much information. A user's path through the network can be recreated easily but how a user navigated on individual sites is unknown. New features should thus focus on obtaining data about direct human input, such as mouse movement and scrolling behaviour.

RQ 2: Tracker Architecture
How would such a tracker work?
Traditional tracking software only records the data the browsers sends to the server during an HTTP request. As this data does not include information about the user's input a new system to track users has to be developed. Tracing input, such as the mouse cursor, requires either tracking inside the user's web browser using JavaScript or secondary software to be installed. Developing new tracking software capable of

recording human input thus implies building a Server/Client architecture.

RQ 3: Feature Usage

Can these new features provide explanations for lucrative and non-lucrative clicks in a Wikipedia-game scenario?
In order to be able to reason about the viability of human input as navigation feature an analysis has to be conducted. The recorded data has to be compared with results from previous studies. Mission based games, such as Wikispeedia and The Wiki Game, were considered a golden standard and thus, a similar study was performed. If human input can provide possible explanations as to whether a click was lucrative then the data can be used for further research.

1.2 Related Work

Related work can be separated into two areas. Firstly, analysing and understanding human navigation on the Internet and secondly, how user input such as mouse movement can help be used to understand a humans interest and intentions and how they can approximate the results of much more complex eye-tracking hardware.

The initial inspiration for this thesis was the research by West, Pineau, and Precup. The website *Wikispeedia*, built by West, serves as a scientific game platform. Visitors get tasked with completing specific navigation missions. Starting at a random page *A* the user then has to navigate to another random page *B* using only the links provided on the individual pages. The underlying network on which the navigation occurs is a limited dump of the English *Wikipedia*, *Wikipedia for Schools 2007*. The network follows the rules of the *Small World* (Milgram, 1967) allowing for even random missions to be completed within a limited amount of hops. The resulting data was used for human computation comparable to the *Amazon Mechanical Turk* (Amazon, 2015) or *Verbosity* (Von Ahn, Kedia, and Blum, 2006). Primary goal of the initial research was to calculate the semantic distance between two concepts (the start and target of a mission) using those human traces and combining them with posterior knowledge, such as the PageRank (Brin and Page, 1998), resulting in significantly outperforming purely mathematical approaches such as *Latent Semantic Analysis*. The recorded click traces revealed a broad pattern of how the players navigated the network coining the terms *getting-away*- and *homing-in*-phase. (West, Pineau, and Precup, 2009)

Singer et al. used data provided by *The Wiki Game*, a platform producing similar data to *Wikispeedia* but using the live *Wikipedia*. Relatedness was captured using *sliding windows* and then compared to a golden standard dataset, the *WordSimilarity-353* (Finkelstein et al., 2001). The research iterated upon West, Pineau, and Precup showing further hints that a path that a human used for navigation can be used as source for calculating the semantic related of two concepts. (Singer et al., 2013)

The data collected by *Wikispeedia* has since been used for a multitude of projects. West and Leskovec further shift the focus from using the human input for calculation to understanding the human navigational patterns. Analysing over 30000 Wikispeedia-Missions revealed that human wayfinding is not only usually very efficient but follows a set of characteristics that differ from the ideal shortest path. Humans try to navigate using hubs. Thus, the first step in the *getting-away phase* is to find a large node connected to the start. This almost always reduces the distance to the target. During the second phase (*homing-in phase*) the player jumps from hub to hub trying to get closer to the target. During this phase the distance stays relatively the same until a hub in the target area is found. Whilst performing the final steps the user spears down towards the target reducing the distance with almost every click. Hence, the first and the last few clicks are usually very efficient whilst the hops in between are not. The players displayed varying selection patterns during the course of a mission. The first hops are based on degree of the pages whilst for the last steps the users looks for similarity to decide which link to follow. With these characteristics in mind a predictive model was built reflecting human wayfinding. (West and Leskovec, 2012)

One characteristic of human wayfinding is that the search for a target is not always successful. Scaria et al. were interested those missions where the players had given up trying to find the target page. Backtracking, as in returning to a previously visited page, often caused by the player realising that an incorrect path was chosen. In order to explore the network users tend to backtrack to hubs, but a high number of backtracks during a mission causes the player to get more and more frustrated eventually abandoning a mission. To measure the backtracking rate of a path the *Star score* was introduced. During the early stages of a mission the overall backtracking rate is low. But when the player has to home in onto the single target page backtracking increases. Thus, when a mission is abandoned the player was typically only 1 or 2 hops away from the target. (Scaria et al., 2014)

The second large source of tracking data comes from search engines. Improving the quality of search results and understanding how humans interact with search interfaces is in the best interest of websites like *Google*. Already back in 1989 the concept of *berrypicking* was introduced to describe human interaction with search interfaces. (Bates, 1989) A user does typically not know how to precisely find a specific piece of information like described in the classic model of information retrieval. (Robertson, 1977) Mostly because the *Universe of Interest* is much smaller than the *Universe of knowledge*. Users type in their initial query, look at the results and then update the query if the target information is not present. (Bates, 1989) Thus, a good ranking of the retrieved search results is very important.

In an attempt to improve the result lists Agichtein, Brill, and Dumais analysed 3000 sample queries from logs created by search engines and derived a ranking for the search results based on implicit user feedback. Some of the features used were the position of the search result, how often the link was clicked as well as checking how much the title of the search result overlaps with the initial query. This implicit feedback can be exploited to improve the ranked result lists returned by search engines. (Agichtein, Brill, and Dumais, 2006)

Iterating upon this research Bilenko and White analysed the user behaviour after a click on a result from a search engine. Again using this implicit user feedback, such as time stayed on a page or deeper links followed, algorithms were created combining both the search history and browsing behaviour of users to estimate a topical relevance of a given webpage. (Bilenko and White, 2008)

Tracking actual human input is the biggest improvement of the tracking system presented in this thesis but research on understanding the implications of mouse movements and clicks is nothing new.

Rayner performed extensive research on eye-tracking. By analysing saccade and gaze durations of humans during different tasks it is shown that the tracked data is more descriptive during problem solving than general reading. Hence, in his review "Eye movements in reading and information processing: 20 years of research." he came to the conclusion that data generated by tracking the human eye depends on the task a user was given and cannot be generalised. (Rayner, 1998)

Because of this dependency on the individual task Granka, Joachims, and Gay performed a study on user behaviour during internet searches using eye-tracking. Over the course of 397 search queries data was recorded. It was revealed that humans tend to scan pages from top to bottom and those users who clicked on links leading to more specific pages (having a lower rank) ingested proportionally more information on the current page than those who did not. (Granka, Joachims, and Gay, 2004)

Eye-tracking is not possible when recording users at remote locations (because the required hardware is not common). Hence, researchers tried to find a way to estimate the data eye-tracking would provide using other means of human input discovering a correlation between the position of the mouse cursor and the user's gaze. Chen, Anderson, and Sohn performed experiments using eye-tracking hardware and recorded humans during their Internet browsing activities using a modified web browser. Analysing mouse movement and saccades it was discovered that in over 75% of the recorded occurrences the mouse stopped on a meaningful region. And when this happens it can be assumed that the user's gaze is very close to the position of the mouse cursor on the screen. Thus, mouse movement can be used to estimate the area on a website the user is looking at. Complex eye-tracking hardware is no longer required to obtain a broad idea about zones of interest on a website. (Chen, Anderson, and Sohn, 2001)

MouseTrack implements a first approach of tracking mouse cursors on websites not only when a user clicks but during the entire page visit. Drawbacks of the system described are that JavaScript reliant websites and sites that do not strictly follow the W3C standards may interfere with the tracking system. Nevertheless, the MouseTrack system is capable of tracking and visualising human behaviour on a website. (Arroyo, Selker, and Wei, 2006)

A first attempt at a user tracking tool for *Remote User Testing* was already built in 2001. The software called *WebQuilt* uses a proxy to intercept and record a user's input which is, at its core, a similar approach as it was used in the tracking software presented in this thesis. But, limited by technology the capabilities do not go far beyond tracking the websites a user visited. Whilst this provides no benefit if the study is limited on websites on a controlled server the WebQuilt proxy is capable of recording the traces anywhere throughout the Internet. Unlike modern systems the visited websites are modified by the system. All *href* elements (Links) are changed to again route through the proxy. This made it almost impossible to escape the tracking system. (Hong and Landay, 2001) It has to be noted that dynamic websites, relying heavily on JavaScript, were not present at the time of this research and now, the system would be incapable of tracking many websites available on the Internet.

This study is unique as it aims to merge research on human wayfinding with human input analysis. The study process is well known and the recorded data allows for an analysis much more detailed than previously possible. The focus of this thesis lies on explaining how these new features were obtained with the newly developed tracking system and on illustrating the new possibilities.

2 The Tracker

Tracking users on the Web is a common task in the aspect of science as well as for commercial reasons, most notably Google with its product *Google Analytics*. Yet, the goal of these two groups varies greatly. With the example of Google Analytics the core aim of the system is to provide demographical data information about each individual page to the site-owner and provide Google with information about the specific user to improve its advertising. The approaches having a scientific background, for example *Wikispeedia*, focus on analysing a general user's path through the network.

For my research the more general approach presented itself as the ideal choice since the goal is to get information about the average user and compare the performance and information value of the newly acquired features to existing studies.

This section will firstly outline the idea behind the tracking system and show the differences to existing systems, followed by an explanation of the technical implementation and a discussion on my design decisions.

2.1 Core Concept

Due to its common use in existing research *Wikispeedia* was considered the base model for the analysis and served as reference when designing the new tracker. I particularly focused on its capabilities to detect functional weaknesses. Based on this idea it quickly became apparent that the best approach would be to grab data that is not transmitted when a HTTP request is made. Primarily this meant mouse-cursor information. In addition to that I wanted to split the tracker as far away as possible from any domain specific knowledge. The final consideration was on the look and feel. To me it was very important to minimise the influence on the user. This meant keeping the tracker hidden where possible and not derogate the browsing experience in any way. Hence, performance was an important factor as well. Finally the decision was made to split the system into two parts: The server which contains domain specific knowledge and knows about the network, users and missions, and the tracker itself which should run inside the users browser in the background. For the end-user no installation of any kind should be required.

A second engineering goal was to develop a system which is easy to extend. New tracked data should, ideally, only require new code where it is actually created and recorded. For example during the early stages the client did not track the maximum data size of the currently viewed page. Adding this data to be tracked required telling the JavaScript to store this size when calculating the viewport. On the Server not a single line of code had to be changed.

2.1.1 Difference to Existing Systems

Thanks to the consistent increase in Web-Browser performance and the broadly implemented features from the upcoming *HTML5* standard I was able to implement the tracking software in JavaScript. *Wikispeedia*, for instance, relies on data from HTTP Requests as sources for analysis. Hence, the majority of actual input done by the user never reaches the server, resulting in an inability to record possibly interesting data. Without escaping a web browser's sandbox the approach providing the largest amounts of data is to grab JavaScript events directly. Normally, these events, including mouse movements and actions, or information about the viewport, are being used to provide better and sometimes interactive experiences on various websites. Attaching the tracker to this interface allows for the tracking of new features.

Disadvantages

JavaScript is an interpreted language and this interpretation is done in the users's browser. Hence, the source-code is visible to the user. Whilst it is possible to counteract this flaw with obfuscation any user with a reasonable understanding in JavaScript can decode the system and flood the server with useless noise. To prevent this from happening various security mechanisms had to be implemented which would otherwise not be required.

2.2 Structure and Architecture

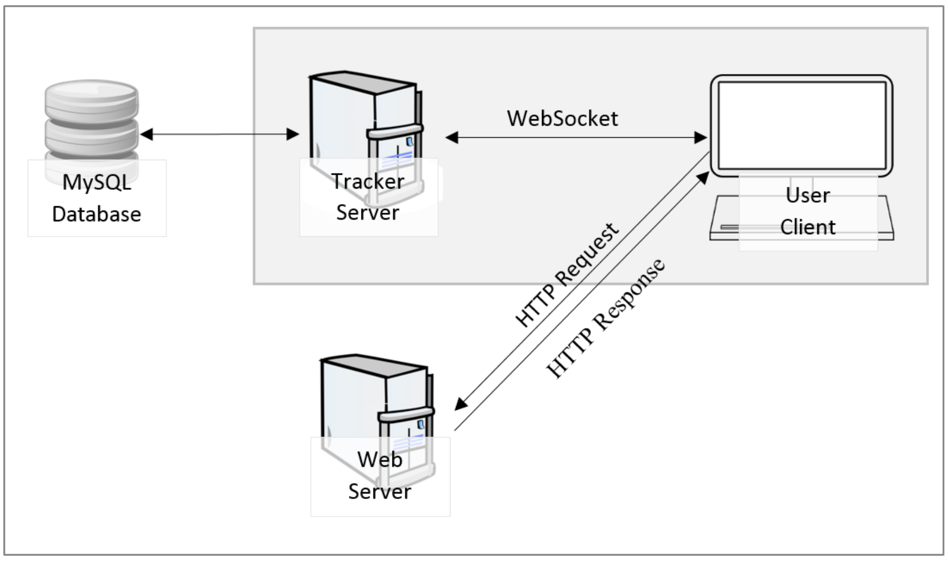


Figure 2.1: **System Structure** - An overview illustrating the individual components of the tracking system and the protocols they use to communicate with each other. The two components manually implemented are the tacker server and the user client who communicate via a WebSocket. The user client tracks JavaScript events and forwards them to the tracker server who then creates the log files and manages the study missions. Thus, the client is able to function without any domain specific knowledge. For the web server and the database the common applications MySQL and Apache were used and are accessed by typical means.

As seen in figure 2.1 the system is split into several components. On the user's side there are the tracker and the actual Wikipedia Game whilst on the remote end on the server both the tracker-server and the HTTP-server are running. This separation on both ends increases the portability of the system as a whole. This section aims to shed light on the client as well as the server and illustrate how they communicate with each other, then goes over the technologies used for each module and finally discusses the requirements and portability.

2.2.1 Server-Side

Helper Servers

Helper servers are modules required by the tracking system. To minimise the impact on the server machine very common and widely distributed packages are used.

HTTP Server

A HTTP server is required if the user has to visit sites which are stored on the server machine. For this tracker the *Apache HTTP Server Project* package has been used. Alternatively, any webserver can be tasked with hosting the website. Understandably, when tracking on external sites this module is not required.

SQL Server

In its current implementation all the required information about the underlying network is stored within a Database. With small changes to the Tracker Server, introduced in this chapter the entire SQL Server can be omitted. Doing so requires to store the entire network inside the Tracker Server. For small networks this does not cause problems even for weaker machines, but as the network size and the size of its data increase the high memory requirement is likely to cause problems. Hence, the SQL Server was attached to the project for its capabilities to quickly store and access large amounts of data. The current implementation works with MySQL

Table	Description
categories	Maps the category-hierarchy
category_pages	Maps individual sites to their categories
missionlists	Sets of references to individual missions.
missions	Start and target pages for each mission
sessions	The states for each active or finished user session.
hub_mapping	Recommended pages for each page. (Future Work)
links	Edge mapping for the network.
log	Database copy of the logfiles created by the tracker
node_values	Network metrics (degree, PageRank) for each page
pages	Names and URLs for each page
path_lengths	Shortest path lengths for each individual pair of pages
users	The data a user entered at the beginning of the session

Table 2.1: **Database Structure** - The table structure of the database as required by the tracking system.

and table 2.1 outlines the structure of the database as used by the tracking system.

Tracker Server

A server for the tracker has to run on the remote machine and is currently implemented in python. The Tracker Server is the sole module requiring limited domain knowledge which is stored in the SQL Database.

One of the challenges when designing a system like this is to prepare for multiple concurrent users and keeping the system load low. To do so the amount of per-user-modules was minimised. Each new connection starts a new instance of a tracker class. This class contains the methods to store the log data and to start, stop, or restore a session, and contains a user instance as well. To communicate with the database or fetch domain specific data a global set of functions is available for all tracker instances. Making these global allows the system to improve the memory consumption without any drawback.

Aquiring User Data

Personal information is helpful when trying to interpret results. But since the specifics about which data is needed varies greatly from project to project a very simply system was devised. For each individual piece of information only the name and the data-type, for instance string or integer, are required. Until satisfied the server will ask the client to prompt for this information. In order to not ask the same question twice after an exception (for instance a crash) the client can locally store the user input and then reply instantly to the server's inquiry without prompting the user. This storage technique is illustrated in section 2.2.2.

Logs

Field Name	Datatype
Server Timestamp	integer
Client Timestamp	integer
Type	string
Payload	string

Table 2.2: **Log Entry Structure** - Each entry in a log file contains these 4 fields. Since the final storage is text-based the entry is transformed into a string but when parsing the log and splitting it back up the listed datatypes can be assumed for the respective fields.

Log files represent the data collected by the tracker. For each task a separate file is being created. All broadcasted events the server receives from the client are being stored here and each event is saved as a line. As seen in table 2.2 each entry has two different timestamps. The Server Timestamp marks the time, as UNIX epoch timestamp, when the event has arrived at the server. Client Timestamps on the other hand are created on the user's machine. It starts at 0 for each task and increases in milliseconds as the user navigates the network. For any analysis the Client Timestamp is to be used, because it does not care about internet lag and random short disconnects and is therefore a much more precise. After the timestamps the types of messages is stored. There are 4 base types of log messages:

- Mission Status

- Screenshot
- Link Data
- (JavaScript) Event

Each type has a different payload structure. Mission Status marks when a task is being started, completed, or aborted. Hence, the first and the last line of each log file will always be of this type. In a normal scenario there is no additional line of this type in the body. Only if the connection is interrupted for any reason and the session cannot be fully restored then a new start event will be logged. If the analyst has no interest in the preceding data form the interrupted session the block above this line can be removed. But in many cases the information value is the same. The payload of Mission Status entries is the name of current mission.

Screenshot entries are being produced when, as the name suggests, a screenshot was sent to the server. The payload simply shows the file's path and name.

Link Data entries hold information about any link clicked. Its payload provides information about the actual HTML link object inside the pages DOM tree. It contains the following data:

- Inner Text
- Link Target
- Offset
  - Link Index
  - Thumbnail-Info
  - Word Offset

The Inner Text of the link shows the actual text of the link, the usually underlined text the user can click. The target shows where the link leads to and the offset marks some metrics about its position. The link index tells the analyst the index of the link within the DOM tree so the page can be parsed at a later date and the exact link which has been clicked can be identified. The other two values are meant to allow analysis without later parsing. Thumbnail link is a Boolean value expressing whether the link was inside an info box. This is Wikipedia specific and needs to be changed or ignored for other networks. The word offset counts the words that appear before the link, ignoring all HTML tags. Using this data anyone can calculate positioning of the link with a good precision.

JavaScript Events contain the main bulk of the tracked information. The server does not care about the type of the event as this type is passed within the message block. Hence, the text in the message field is the actual type of the event. This has been done to easily add new tracked features. Each of these entries comes along with a large serialised object describing the user's mission and browser states. It contains the following fields:

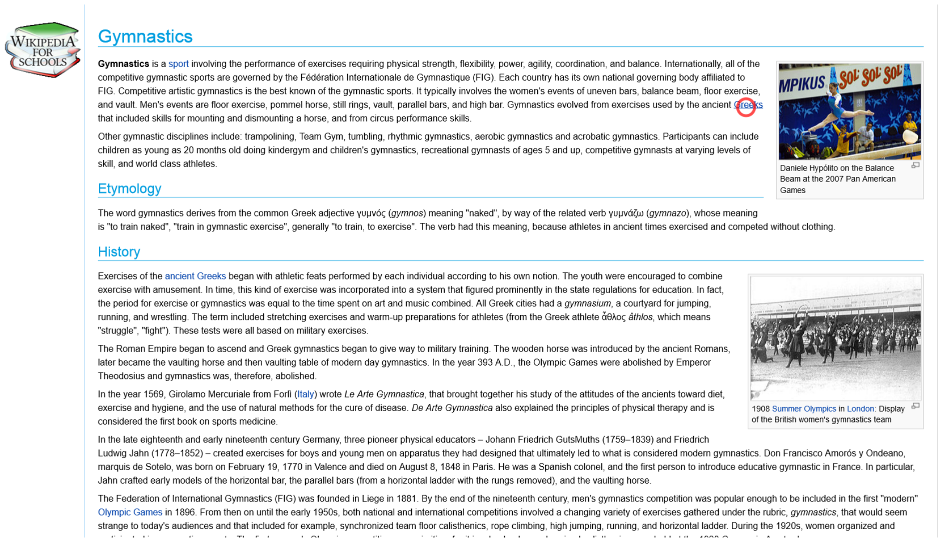
- Status
  - Mission ID
  - Current Page
- Timestamp
- Mouse
  - Position
  - Pressed Buttons
- • Viewport
  - Size
  - Scroll Offset
  - Scroll Size

The Mission ID and Current Page are needed to uniquely identify where the user is positioned in the network and what the current task/mission is. Naturally, the Mission ID is identical with mission status log entries and the timestamp is identical with the Client Timestamp field.

The Mouse field shows where the user"s mouse cursor was when the event fired and the viewport provides information about what the user actually saw. Note, the mouse position ignores scroll status. The top left corner will always be $[0, 0]$ regardless how far has been scrolled. An in-depth clarification of these fields can be found in chapter 3.

Screenshots

To better visually reconstruct what the user actually sees at any point a screenshot functionality was added to the tracker. At its core this module uses the *html2canvas* library. To produce its image the library parses the html tree and the associated stylesheets and builds a representation of the data inside an HTML canvas object. (von Hertzen, 2015) Before being sent to the server a small indicator is placed on the spot where the mouse cursor was positioned when the event triggered. Figure 2.2 shows the result of this process on a sample page.



Gymnastics

Gymnastics is a sport involving the performance of exercises requiring physical strength, flexibility, power, agility, coordination, and balance. Internationally, all of the competitive gymnastic sports are governed by the Fédération Internationale de Gymnastique (FIG). Each country has its own national governing body affiliated to FIG. Competitive artistic gymnastics is the best known of the gymnastic sports. It typically involves the women's events of uneven bars, balance beam, floor exercise, and vault. Men's events are floor exercise, pommel horse, still rings, vault, parallel bars, and high bar. Gymnastics evolved from exercises used by the ancient Greeks that included skills for mounting and dismounting a horse, and from circus performance skills.

Other gymnastic disciplines include: trampolining, Team Gym, tumbling, rhythmic gymnastics, aerobic gymnastics and acrobatic gymnastics. Participants can include children as young as 20 months old doing kindergym and children's gymnastics, recreational gymnasts of ages 5 and up, competitive gymnasts at varying levels of skill, and world class athletes.

Daniele Hypólito on the Balance Beam at the 2007 Pan American Games

Etymology

The word gymnastics derives from the common Greek adjective γυμνός (*gymnos*) meaning "naked", by way of the related verb γυμνάζω (*gymnazo*), whose meaning is "to train naked", "train in gymnastic exercise", generally "to train, to exercise". The verb had this meaning, because athletes in ancient times exercised and competed without clothing.

History

Exercises of the ancient Greeks began with athletic feats performed by each individual according to his own notion. The youth were encouraged to combine exercise with amusement. In time, this kind of exercise was incorporated into a system that figured prominently in the state regulations for education. In fact, the period for exercise or gymnastics was equal to the time spent on art and music combined. All Greek cities had a *gymnasium*, a courtyard for jumping, running, and wrestling. The term included stretching exercises and warm-up preparations for athletes (from the Greek athlete ἆθλος *âthlos*, which means "struggle", "fight"). These tests were all based on military exercises.

The Roman Empire began to ascend and Greek gymnastics began to give way to military training. The wooden horse was introduced by the ancient Romans, later became the vaulting horse and then vaulting table of modern day gymnastics. In the year 393 A.D., the Olympic Games were abolished by Emperor Theodosius and gymnastics was, therefore, abolished.

In the year 1569, Girolamo Mercuriale from Forlì (Italy) wrote *Le Arte Gymnastica*, that brought together his study of the attitudes of the ancients toward diet, exercise and hygiene, and the use of natural methods for the cure of disease. *De Arte Gymnastica* also explained the principles of physical therapy and is considered the first book on sports medicine.

1908 Summer Olympics in London: Display of the British women's gymnastics team

In the late eighteenth and early nineteenth century Germany, three pioneer physical educators – Johann Friedrich GutsMuths (1759–1839) and Friedrich Ludwig Jahn (1778–1852) – created exercises for boys and young men on apparatus they had designed that ultimately led to what is considered modern gymnastics. Don Francisco Amorós y Ondeano, marquis de Sotelo, was born on February 19, 1770 in Valence and died on August 8, 1848 in Paris. He was a Spanish colonel, and the first person to introduce educative gymnastic in France. In particular, Jahn crafted early models of the horizontal bar, the parallel bars (from a horizontal ladder with the rungs removed), and the vaulting horse.

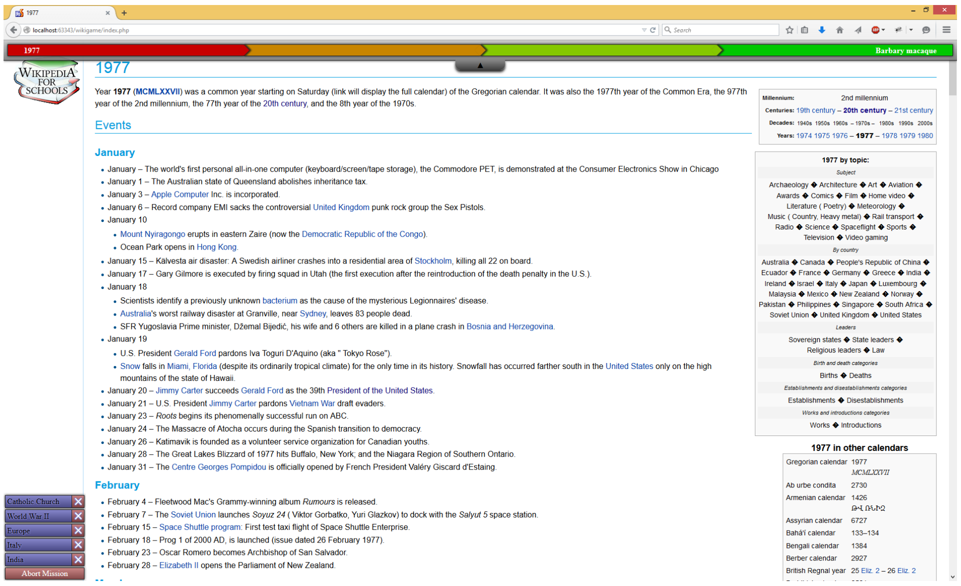
The Federation of International Gymnastics (FIG) was founded in Liege in 1881. By the end of the nineteenth century, men's gymnastics competition was popular enough to be included in the first "modern" Olympic Games in 1896. From then on until the early 1950s, both national and international competitions involved a changing variety of exercises gathered under the rubric, *gymnastics*, that would seem strange to today's audiences and that included for example, synchronized team floor calisthenics, rope climbing, high jumping, running, and horizontal ladder. During the 1920s, women organized and

Figure 2.2: **Automatic Screenshot** - An example of an automatically generated screenshot. The JavaScript reconstructs the actual view of the user and stores the resulting image into a canvas. Before being serialised and sent to the server a small indicator is placed on the position the user's mouse was at when the screenshot was triggered.

Once constructed the serialised image is sent over the socket to the server to be stored. This is the most accurate method for generating website screenshots of remote machines without violating the rule that no software is to be installed.

This way of creating images comes with a downside. The required computational power is high and since the tree must not be manipulated during the parsing process a weaker machine or a large screen may cause stuttering and impedes the browsing experience. Hence, this module is currently only in use for problem reporting.

2.2.2 Client-Side



1977

Year **1977** (MCMLXXVII) was a common year starting on Saturday (link will display the full calendar) of the Gregorian calendar. It was also the 1977th year of the Common Era, the 977th year of the 2nd millennium, the 77th year of the 20th century, and the 8th year of the 1970s.

Events

January

- January – The world's first personal all-in-one computer (keyboard/screen/tape storage), the Commodore PET, is demonstrated at the Consumer Electronics Show in Chicago
- January 1 – The Australian state of Queensland abolishes inheritance tax.
- January 3 – Apple Computer Inc. is incorporated.
- January 6 – Record company EMI sacks the controversial United Kingdom punk rock group the Sex Pistols.
- January 10
  - Mount Nyiragongo erupts in eastern Zaire (now the Democratic Republic of the Congo).
  - Ocean Park opens in Hong Kong.
- January 15 – Kälvesta air disaster: A Swedish airliner crashes into a residential area of Stockholm, killing all 22 on board.
- January 17 – Gary Gilmore is executed by firing squad in Utah (the first execution after the reintroduction of the death penalty in the U.S.).
- January 18
  - Scientists identify a previously unknown bacterium as the cause of the mysterious Legionnaires' disease.
  - Australia's worst railway disaster at Granville, near Sydney, leaves 83 people dead.
  - SFR Yugoslavia Prime minister, Džemal Bijedić, his wife and 6 others are killed in a plane crash in Bosnia and Herzegovina.
- January 19
  - U.S. President Gerald Ford pardons Iva Toguri D'Aquino (aka " Tokyo Rose").
  - Snow falls in Miami, Florida (despite its ordinarily tropical climate) for the only time in its history. Snowfall has occurred farther south in the United States only on the high mountains of the state of Hawaii.
- January 20 – Jimmy Carter succeeds Gerald Ford as the 39th President of the United States.
- January 21 – U.S. President Jimmy Carter pardons Vietnam War draft evaders.
- January 23 – *Roots* begins its phenomenally successful run on ABC.
- January 24 – The Massacre of Atocha occurs during the Spanish transition to democracy.
- January 26 – Katimavik is founded as a volunteer service organization for Canadian youths.
- January 28 – The Great Lakes Blizzard of 1977 hits Buffalo, New York; and the Niagara Region of Southern Ontario.
- January 31 – The Centre Georges Pompidou is officially opened by French President Valéry Giscard d'Estaing.

February

- February 4 – Fleetwood Mac's Grammy-winning album *Rumours* is released.
- February 7 – The Soviet Union launches *Soyuz 24* (Viktor Gorbatko, Yuri Glazkov) to dock with the *Salyut 5* space station.
- February 15 – Space Shuttle program: First test taxi flight of Space Shuttle Enterprise.
- February 18 – Prog 1 of 2000 AD, is launched (issue dated 26 February 1977).
- February 23 – Oscar Romero becomes Archbishop of San Salvador.
- February 28 – Elizabeth II opens the Parliament of New Zealand.

Millennium:	2nd millennium
Centuries:	19th century – **20th century** – 21st century
Decades:	1940s 1950s 1960s – 1970s – 1980s 1990s 2000s
Years:	1974 1975 1976 – **1977** – 1978 1979 1980

1977 by topic:

Subject

Archaeology � Architecture � Art � Aviation � Awards � Comics � Film � Home video � Literature (Poetry) � Meteorology � Music (Country, Heavy metal) � Rail transport � Radio � Science � Spaceflight � Sports � Television � Video gaming

By country

Australia � Canada � People's Republic of China � Ecuador � France � Germany � Greece � India � Ireland � Israel � Italy � Japan � Luxembourg � Malaysia � Mexico � New Zealand � Norway � Pakistan � Philippines � Singapore � South Africa � Soviet Union � United Kingdom � United States

Leaders

Sovereign states � State leaders � Religious leaders � Law

Birth and death categories

Births � Deaths

Establishments and disestablishments categories

Establishments � Disestablishments

Works and introductions categories

Works � Introductions

1977 in other calendars

Gregorian calendar	1977 *MCMLXXVII*
Ab urbe condita	2730
Armenian calendar	1426 ԹՎ ՌՆԻԶ
Assyrian calendar	6727
Bahá'í calendar	133–134
Bengali calendar	1384
Berber calendar	2927
British Regnal year	25 Eliz. 2 – 26 Eliz. 2

Figure 2.3: **Tracker Screenshot** - The user's view of the tracking system with all modules activated during a recording session. In the bottom left corner hints and an abort button can be found. On the top of the page the user sees information about the current mission and in the back tracking system displays the iframe containing the *Wikipedia for Schools 2013* pages. Note, the character-set errors displayed on the right-hand side are due to errors during the crawling process of the *Wikipedia* and are not caused by the tracking system.

As mentioned one of the innovations of this tracking system is that the tracking is performed at the user's end. This idea is nothing new in remote user testing, but until now a specific piece of software, or browser extension, was required in order for trackers to work. The tracker built for this thesis works on all modern, JavaScript enabled browsers relying solely on already built in features. This section aims to first explain how the JavaScript package works, what features it provides and will list everything currently recorded by the tracking system. Subsequently an illustration of how the trackers client and server modules communicate will be given. Note, only modules relevant for general understanding of the functionality, the research and the user study will be explained here.

Functionality

The client is built out of several independent modules. Each of these provides a set of public and private methods for all other modules to use. When the user's browser loads and interprets the code the first thing after initialisation of the modules is the creation of an HTML iframe. The tracker resides inside the main window whilst the user will actually navigate within this iframe. To minimize the impact on the user experience this iframe takes up all available space whilst still providing the tracker the option to create various overlays. Once the iframe and the overlays are created the initial page is loaded into it so that the user does not look at a white screen whilst the remaining modules are starting up.

As next step the system tries to connect to the server via the use of a *WebSocket*. Once this connection is established and the handshake done the actual tracking module attaches itself to the iframe. It listens to the events listed in table 2.3 and, because a *documentWindow* clears its state at any reload, reattaches the event listeners whenever a new page is loaded inside the iframe. Naturally, because if this architecture the user is not allowed to escape from the iframe because new windows or new tabs cannot be tracked that way. This can be avoided by either not allowing new tabs and windows or adding a small line of code to each visit-able page which reloads the tracker. For this work the first option was chosen because I wanted to limit the manipulation of the natural dataset as little as possible. Injecting JavaScript is not advised when tracking external sites since intrusion detection software can consider this action as an attack.

Whenever a tracked event is fired its data is passed to the event handler module, a long *switch/case* statement decides what to do with each of them. Every event is assigned a broadcast value, a Boolean deciding whether a package should be sent over the socket or not. This allows for a high level of customisation. Since events can be fired very rapidly (for instance mouse move) a spam interval variable was introduced. This allows for throttling of the transmitted data because a package is only sent for an event when the given time has passed since its last occurrence.

Some events shown in table 2.3 trigger an update to the stored viewport information. These are caused by changes to the area the user sees at any given time. Note, the load event is added here because this is the first event ever fired and therefore used as trigger to calculate the initial values.

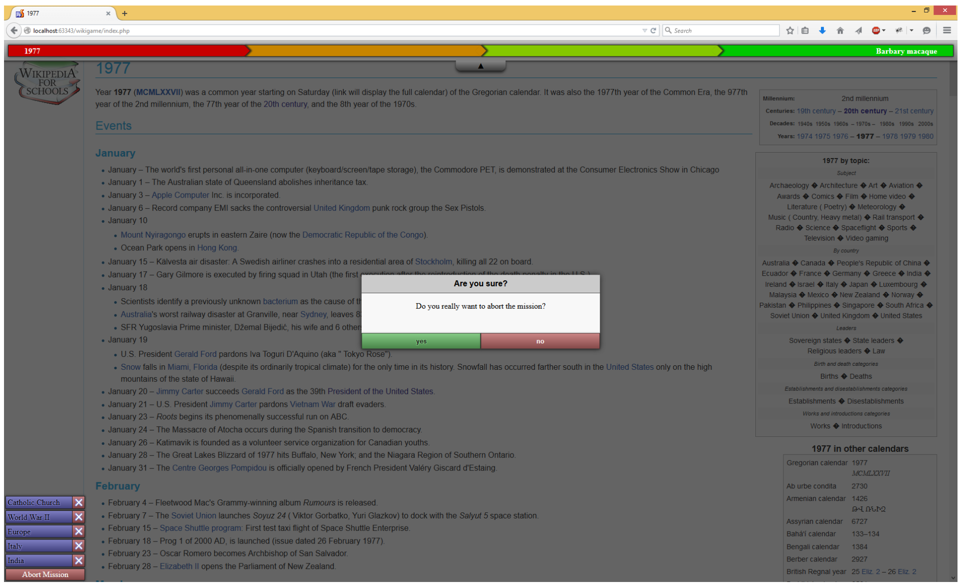
When an event is to be broadcasted the network controller is notified. It creates a package containing the event type, its information, the current page and the mouse and viewport states and then passes this on to a message queue. Normally this queue is being flushed immediately and the information is sent to the server. But, if for any reason the connection is interrupted the queue increases until is flushed as soon as the connection is restored. Again, this is why for the data analysis the client timestamp should be used because it marks the time when the event was fired, not when it was finally sent to the server.

Whenever the server intends to send data to the client the same WebSocket is being used. In a similar fashion to the event handler any received message will first be checked for integrity and then passed to the message handler. The main use for this functionality is to inform the client about new missions, to provide the user with information and to prompt for input (for instance to fetch the user data at the start of a session). Additionally it can be used to show or hide the navigation menu and server as an interface for future tracker extensions.

Once the initial setup is completed and the server has all the required information, a mission is sent to the client. The data required for the client are name and URL of both the start and the goal pages. Additionally a value indicating the shortest distance can be sent to provide visual feedback. The client will then load the start page and display the target to the user. Finally an overlay providing possibly helpful links, if they provided by the server, and an abort button are added to the user's view.

From here on in the tracker records all events and continuously forwards the data to the server. The client itself does not care what site the user is currently on. Even if the goal page is reached it will continue to track. The server has to perform these checks and then inform the client that an individual mission or the entire session is completed. Once the session is finished the client disconnects from the server and deletes all locally stored data in order to not leave any unnecessary ballast and prepare for the next potential user.

Overlays



1977

Year **1977** (MCMLXXVII) was a common year starting on Saturday (link will display the full calendar) of the Gregorian calendar. It was also the 1977th year of the Common Era, the 977th year of the 2nd millennium, the 77th year of the 20th century, and the 8th year of the 1970s.

Events

January

- January – The world's first personal all-in-one computer (keyboard/screen/tape storage), the Commodore PET, is demonstrated at the Consumer Electronics Show in Chicago
- January 1 – The Australian state of Queensland abolishes inheritance tax.
- January 3 – Apple Computer Inc. is incorporated.
- January 6 – Record company EMI sacks the controversial United Kingdom punk rock group the Sex Pistols.
- January 10
  - Mount Nyiragongo erupts in eastern Zaire (now the Democratic Republic of the Congo).
  - Ocean Park opens in Hong Kong.
- January 15 – Kälvesta air disaster: A Swedish airliner crashes into a residential area of Stockholm, killing all 22 on board.
- January 17 – Gary Gilmore is executed by firing squad in Utah (the first execution after the reintroduction of the death penalty in the U.S.)
- January 18
  - Scientists identify a previously unknown bacterium as the cause of th
  - Australia's worst railway disaster at Granville, near Sydney, leaves 8
  - SFR Yugoslavia Prime minister, Džemal Bijedić, his wife and 6 other
- January 19
  - U.S. President Gerald Ford pardons Iva Toguri D'Aquino (aka " Tokyo Rose").
  - Snow falls in Miami, Florida (despite its ordinarily tropical climate) for the only time in its history. Snowfall has occurred farther south in the United States only on the high mountains of the state of Hawaii.
- January 20 – Jimmy Carter succeeds Gerald Ford as the 39th President of the United States.
- January 21 – U.S. President Jimmy Carter pardons Vietnam War draft evaders.
- January 23 – *Roots* begins its phenomenally successful run on ABC.
- January 24 – The Massacre of Atocha occurs during the Spanish transition to democracy.
- January 26 – Katimavik is founded as a volunteer service organization for Canadian youths.
- January 28 – The Great Lakes Blizzard of 1977 hits Buffalo, New York; and the Niagara Region of Southern Ontario.
- January 31 – The Centre Georges Pompidou is officially opened by French President Valéry Giscard d'Estaing.

February

- February 4 – Fleetwood Mac's Grammy-winning album *Rumours* is released.
- February 7 – The Soviet Union launches *Soyuz 24* (Viktor Gorbatko, Yuri Glazkov) to dock with the *Salyut 5* space station.
- February 15 – Space Shuttle program: First test taxi flight of Space Shuttle Enterprise.
- February 18 – Prog 1 of 2000 AD, is launched (issue dated 26 February 1977).
- February 23 – Oscar Romero becomes Archbishop of San Salvador.
- February 28 – Elizabeth II opens the Parliament of New Zealand.

Do you really want to abort the mission?

Millennium:	2nd millennium
Centuries:	19th century – **20th century** – 21st century
Decades:	1940s 1950s 1960s – 1970s – 1980s 1990s 2000s
Years:	1974 1975 1976 – **1977** – 1978 1979 1980

1977 by topic:

Subject

Archaeology � Architecture � Art � Aviation � Awards � Comics � Film � Home video � Literature (Poetry) � Meteorology � Music (Country, Heavy metal) � Rail transport � Radio � Science � Spaceflight � Sports � Television � Video gaming

By country

Australia � Canada � People's Republic of China � Ecuador � France � Germany � Greece � India � Ireland � Israel � Italy � Japan � Luxembourg � Malaysia � Mexico � New Zealand � Norway � Pakistan � Philippines � Singapore � South Africa � Soviet Union � United Kingdom � United States

Leaders

Sovereign states � State leaders � Religious leaders � Law

Birth and death categories

Births � Deaths

Establishments and disestablishments categories

Establishments � Disestablishments

Works and introductions categories

Works � Introductions

1977 in other calendars

Gregorian calendar	1977 *MCMLXXVII*
Ab urbe condita	2730
Armenian calendar	1426 ԹՎ ՌՆԻԶ
Assyrian calendar	6727
Bahá'í calendar	133–134
Bengali calendar	1384
Berber calendar	2927
British Regnal year	25 Eliz. 2 – 26 Eliz. 2

Figure 2.4: **Tracker Overlays** - To communicate to the user the tracking systems provides an array of usable overlays. This figure shows the 3 different types of overlay highlighted. At the top domain information sent by the tracker server is displayed showing the start and the target page of the current mission. In the center an overlay used for user briefings or prompts is visible and in the bottom left the hint module can display various links to possibly helpful pages and shows and abort button to cancel the current mission.

As mentioned in the previous subsection the client provides a range of different overlays to visualise mission specific information or prompt the user for data. As seen in figure 2.4 there are 3 types of overlay.

At the top users find a butt strap providing information about the current mission. Start and target page are shown here and if provided and indicator how far the two are apart from each other. An additional highlight can be added by informing the client how far the current page is away from the goal to help the user identifying whether his last click was correct or not. This butt strap can be minimised to only show a small handle at the very top of the page. Opening and closing can be performed by the user as well as by the server. The controller for this overlay allows any information to be displayed allowing the tracker to be used for various tasks.

Secondly on the bottom left-hand side there is an overlay that allows the server to provide hints for the user and it shows an abort button to cancel a mission. Abort is used to inform the server that the current mission will not be completed. Hints are a feature mainly aimed at future work. They provide a possibility to attach a recommender system to the server. The system can provide possible interesting and useful links to the user using this overlay.

Finally there is an overlay required for information input. It can be used to show simple information via an alert box, prompt for yes or no questions and allows for more specific input as required by the server to obtain user information.

All these overlays are contained within the original document. Hence they are only controlled by the tracker client and can ignore in whatever state the tracked iframe is. By using this method information can be displayed at any time without getting lost when the user loads a new page within the tracked iframe.

Tracked Events

Table 2.3 lists every event the tracking client is recoding. The broadcasted value indicates its default state. Naturally, this can be changed at will at any time. Note, whilst click is broadcasted, mouse up and mouse down are not, because each mouse down followed by a mouse up sequence trigger a click event. For this research sending a package to the server was not necessary. These events were used to update the stored mouse information.

Communication and Data Transfer

One of this trackers greatest advancements over comparable pieces of software is its rapid two-way communication. As previously mentioned, traditional trackers track HTTP-Requests sent by the client. This comes with a multitude of downsides. First, it is a one sided system where the server cannot contact a client at will. This can be fixed by the use of a polling method, but scales badly when the concurrent user count increases. Secondly, the

Event Type	Description	Broad-cast	Viewport Update
beforeunload	Before the page is left	No	No
click	After a click is completed	Yes	No
dblclick	After a 2 clicks in rapid succession	Yes	No
load	After a page finished loading	Yes	Yes
mousedown	A button on the mouse is pushed	No	No
mousemove	When the cursor position changes	Yes	No
mouseup	A button on the mouse is released	No	No
resize	When the window browser is resized	Yes	Yes
scroll	Whenever the user scrolls on the site	Yes	Yes

Table 2.3: **Tracked Events** - The JavaScript events currently tracked by the system. The *broadcast* value indicates whether the event gets sent to the server and the *viewport update* field shows if the event causes the viewport information to be recalculated. Viewport updates are neccessairy when the user changes the dimensions of the browser window. The *load* event causes the recalculation of the dimension to avoid empty intial values.

server has to rely on cookies and data passed back and forth in order to associate each package to an individual client and finally, data is transferred in high bursts instead of a lower consecutive transmission. Naturally, a fall-back using these traditional technologies can be implemented to increase stability.

This tracker relies on the WebSocket technology for its data transfer. Hence, very old browsers are not supported. But, since most widespread browsers update themselves automatically the user instalment of browsers supporting this technology is very high. During the user study conducted in chapter 5 not a single non-supported browser was detected. Old browsers cause more problems in general because the high frequency tracking relies on a certain base level of performance in order to not influence the user's browsing experience. Websockets are full sockets that provide a fast full-duplex communication channel over a single connection. The initial handshake is performed by the browser and the python package *Tornado Web Server*. To allow session restores an additional, a manual handshake is performed afterwards.

Handshake

One of the largest problems with continuous connections is the risk of any kind of interruption. These can happen for numerous reasons. The browser can crash, the internet connection can fail or the user accidentally closes the browser. To counteract this risk a method to restore sessions is introduced. Since even IP addresses can change during disconnects and unique browser identifiers are not that widespread, the browser actually has to know who it is. Whilst this is usually done with the help of cookies, our tracking system uses the *Web Storage* interface as described in the *HTML5 draft*. (W3C, 2015a)

A typical handshake begins with the client sending a Hello-message. But as the connection is already established by the underlying software this manual two-way handshake request is sent out by the server. It sends a package of type handshake to the client with a proposed UUID identifier. The client will then check if whether it has already a session identifier stored. If that is the case the client will send a package back to the server with the old identifier. Otherwise the proposed identifier is accepted, stored and finally sent back to the server. Once the server receives the response package it always knows the identifier of the client and can restore the session if the received identifier was not identical with the proposed one.

Message Packages

The base model for communication is a *JSON* string. (ECMA-International, 2015) This was chosen because JSON encoding and decoding is widely supported and it is easily readable for the human eye. As seen in the sample message (figure 2.5) each message generally consists of 4 blocks with the last one, mission_features being optional and only used for tracking data. *type* always describes the message type and the *timestamp* field marks the epoch timestamp when the message was sent. This is used to check the message-flow for integrity. When a message with an old timestamp pops up a warning can be raised. Finally the *message* field is used as payload for the package. In most cases it provides the data required to complete the action requested by the event type. The grey highlighted area in the sample encloses the final, optional message block, the *mission_status*. It passes all the tracked features explained in chapter 3 to the recipient. Obviously, this block is, as previously stated, only attached to messages sent from the client to the server and only when the message package is about a tracked event.

```
{
    "type": "event",
    "timestamp": 1420124618293,
    "message": "mousemove",
        "mission_features":
        {
            "timestamp":1237,
            "mission_status":
            {
                "mission_id": "SET_1_dec91215-4faa-4db7-aae1-f8a0db24991e",
                "current_page": "http://localhost/wikigame/wiki-schools/wp/1/1977.htm"
            },
            "mouse":
            {
                "position":
                {
                    "x": 175,
                    "y": 985
                }
                "buttons_pressed":
                {
                    "left": false,
                    "middle": false,
                    "right": false,
                }
            },
            "viewport":
            {
                "size":
                {
                    "x": 1920,
                    "y": 1089
                }
                "scroll":
                {
                    "x": 0,
                    "y": 0
                }
                "scrollsize":
                {
                    "x": 1903,
                    "y": 11630
                }
            },
        },
}
```

Figure 2.5: **Sample Message** - An example message sent by the client to the server. The values are randomly chosen but representative. Note that, unlike JavaScript, Python quotes the field names in JSON strings. The section highlighted in grey is optional and only sent when a tracked event is being broadcast.

Locally Stored Data

As aforementioned the tracking system makes use of the *Web Storage* interface provided by modern browsers. These JavaScript cookies can persist over individual sessions, meaning unless manually deleted they will remain even if the browser is closed. This feature is essential to cover more possible reasons for disconnects. Additionally this system is used to automate user responses. Whenever the server asks the client to perform a prompt it will first check whether a similar request has already been answered by the user, because any prompt answer is stored within the *localStorage*. If an identical question has already been found the answer will be sent back immediately without ever asking the user. The client deletes this locally stored data upon request in order to allow the tracking system to fully reset.

This storage is used firstly to perform the manual handshake and the session restoration, and secondly to automate the user input in the study (chaper 5) since each user was first given a training set after which the tracker was reset. In order to not force the user to input information about name or age a second time the client replied to the server automatically.

2.3 Technical Requirements

Both the tracker server and the user client have a set of requirements, shown in table 2.4, in order to be able to fulfil their function. Whilst the server requires specific python packages to be installed the client requires a set of built in browser features.

Python Packages (Server)	JavaScript (Client)	External
• tornado	• WebSocket	• Web Server
• unidecode	• localStorage	• MySQL server
• pymysql	• classList	
• binascii (screenshots only)		

Table 2.4: **Tracker Requirements** - The requirements of the full tracking system. For both python and external software commonly used packages were preferred. JavaScript requirements list a set of features the user's browser has to support.

3 New Navigational Features

As outlined in the introduction (chapter 1) the aim of my research is to better understand human navigation and improve user tracking with the help of new, more detailed features in addition to what existing systems can already provide. Chapter 2 gave an overview over the technical aspects, how the features are generated and how they transmitted and stored on the server. This chapter will explain every new feature by analysing how much data they provide and how accurate they are.

3.1 Mouse Actions

When using a PC to navigate the Internet the predominant input device is the mouse cursor. Clicks are used to follow links and to interact with the browser UI. This applies to mobile devices as well since a tap on a website is interpreted as a click. Though, mobile devices offer a wide range of possibly interesting gestures but phones and tablets are not part of this user study or this thesis but are considered an important element for future work (section 7.4). For this project users were asked to use long-established mouse-keyboard based setups in order to provide consistent data.

Mouse cursor actions can be grouped into two types: movement and button presses, commonly known as clicks. The extracted data provides clues to what the user actually does on the currently seen section of a page.

3.1.1 Clicks

Clicks are the JavaScript event that triggers whenever a mouse button is pressed and shortly after released. When such a click is performed on a link the web-browser tries to follow this link and load the next site. Logically, the user can click anywhere on the screen without any specific goal. But clicks can still provide information when performed on elements that provide no actual interaction. For instance, when multiple users click on the same interaction-less element that element should be analysed. The expectation is that users expect a link to be at this position, either because of a common standard on most websites or the information provided by the element relates to the users goal. Either option suggests that changes to the site should be made.

As for this tracker, a click on a link provides detailed information about the actual anchor tag on the site. This can be used in combination with the DOM tree to analyse correlations between certain elements and followed links and will be explained in section 3.4.

Most importantly on untampered sites clicks do not fire without the user performing the action, meaning whenever a click event is fired the user actually clicked with his mouse. The recorded data aims to shed light onto the cause for this click.

3.1.2 Position and Movement

The positioning and movement of the mouse cursor provides a detailed information about what the user is doing at any given moment. Chen, Anderson, and Sohn discovered in their research on eye and mouse movement that there is a strong correlation between the position of the cursor on the screen and where the user is actually looking, implying that a mouse tracker is a reasonable replacement for a much more complex eye tracker. (Chen, Anderson, and Sohn, 2001)

This tracker provides pixel-accurate positioning of the cursor on the screen. This allows for a probabilistic reconstruction of what the user was focussing on at any given time during the tracking session. Frequently hovered areas suggest that position containing important elements on the site which should be analysed. Whilst on an individual site the reasons for such a focus point can be plenty, when merged with the recordings from various sites with a similar structure information about the layout can be extracted. This data can then be used to improve the site in order to provide a better user-experience.

3.2 Viewport Information

When analysing a user's navigation path through a network information about what the user actually saw on its screen at any point can greatly benefit the understanding of this user's behaviour. Considering the *Wikispeedia*-mission scenario the analyst can see whether a successful link was visible when a user performed a click. In addition to that information about the user's viewport reveals how much space has been covered, allowing for determine if a user has even read the page or just skimmed through it. This subsection explains the viewport related track-data and their implications.

All values are provided in pixels. Different display devices provide different pixel densities, meaning huge dimension do not necessarily mean a large screen. But since the focus of this section is to show how much content the user can see pixels provide a better indicator than Euclidean distances as style definitions for text and images usually refer to pixel or point (*pixel * 1.3*) values.

3.2.1 Dimensions

The viewport dimension describes the width and height of the user's browser window. This information is vital to calculate what the user actually saw during navigation. The expectation is that the average user only clicks on links immediately visible after the page has finished loading. Hence, users with larger screens should take more of this initial information into account. Sudden changes of a user's viewport dimensions indicate a browser window resize. In some cases the appearance of a toolbar can affect the viewport size causing a small drop in the height value. Analysing these changes can be interesting as the user opening the browser-built-in search box commonly causes such a toolbar to appear.

3.2.2 Scroll Offset and Content Size

The scroll offset gives insight on how far the user has scrolled every time an event is triggered. As seen in 2.3 scrolling by itself is such an event implying the recorded data immediately shows when a user scrolls and by how much the offset changes. This reconstructs a user's navigation pattern within a single site. Since the event fires whenever the scroll offset changes it is possible to see whether a user scrolled using a browsers scrollbar or jumped around by different means such as the search box.

A pages content size records how far a user can theoretically scroll. Combining this information with scroll offset data it is possible to analyse how much of any visited site the user has actually seen before leaving. The expectation is that higher coverage leads to a higher rate of successful clicks. Please note that the content size depends on a user's viewport dimensions and various CSS settings. In the case of *Wikipedia for Schools 2013* there is always a bar on the left-hand side (as seen in figure 2.2) no matter how high the viewport is. Due to this property a device with a lower width results in a larger content size when content-width and content-height are multiplied. Hence, when analysing coverage viewport dimension changes must be taken into account.

3.2.3 Screenshots

As a visual aid this tracker allows the creation of screenshots. The intended use for this module is to verify the results of an analysis. Note that these images are reconstructions of the actual screen using JavaScript. Whilst some browsers provide built-in functions to provide screenshots these require the JavaScript to run in the privileged mode[1]. This execution mode is

[1]https://developer.mozilla.org/en-US/docs/Web/API/HTMLIFrameElement.getScreenshot

restricted for plugins and hence cannot be used without the installation of an extension for the browser. When used on complex and dynamic pages the used workaround will not be able to provide an accurate representation of the user's view but for simple sites and static sites such as Wikipedia or Wikipedia for Schools the created images is identical with the users view.

3.3 Timings

Providing precise timing information is a vital part of every tracking system. Timestamps are created when the user triggers the event instead of when its information reaches the server. The recorded timing data is given in milliseconds allowing for a very detailed analysis. A possible scenario is how soon a user clicks on a link after it has become visible on the screen. Doing so grants information about how visible the used link is.

Combined with the other extracted features these precise timestamps allow for a very accurate reconstruction of a user's actions and allow for an in-depth analysis of how a user navigates a single site instead of just recreating an overall navigational path through the whole network.

3.4 Link Information

Since following links is a very common, and in this research in fact the only way to navigate through a network, detailed information about any clicked link can provide very useful information. Blackmon et al. analyse the similarity of a links label and the title of the target. In order to perform a similar analysis and compare results the tracker stores the label information about any clicked link. (Blackmon et al., 2002) In addition to that the link followed is uniquely identified via its link index in the DOM-tree. This way multiple links with the same label on a site can be distinguished.

When combined with the mouse data information about the link can be used to for a location based analysis revealing which links are commonly followed and when combined with the DOM-tree a structural analysis is possible. The expectation here is that a user will more likely look for groups of links, for instance a list, as a cognitive connection between the links is made and the user can expect whether or not the goal link is in the group by just looking at its first few members.

3.5 Dataset Dependency

When attempting to reconstruct a user's path through any network it is essential that the network does not change during the recording process. Many websites use dynamic content generation with the help of JavaScript. Whilst this does not affect the tracker in general it does affect these extracted features. When a page shifts whilst the user is navigating it the content size commonly changes. This makes a coverage analysis difficult. For precise experiments it is therefore advantageous if a site remains static. The dataset used for the study in this master's thesis, explained in the next section (4), follows these characteristics in order to better visualise the capabilities of the tracker and to make the results better comparable with existing studies.

4 The Dataset

The user study in section 5 was performed on a known and controlled network. Like most other studies in the field this study was performed on a limited dump of *Wikipedia*, *Wikipedia for Schools 2013*. Wikipedia for Schools is a static and moderated, and very limited dump aimed for the use in schools in the United Kingdom. All information provided on its pages have been checked by experts assuring the facts provided are correct. The removal of any non-child-friendly content is one other benefit allowing users of all ages to participate in the study. This section serves as an overview over the dataset and how the underlying network was extracted from the original data-dump.

For a better understanding the terminology for this chapter is as follows:

Network The combination of all visitable pages and their directed links.

Node An individual visitable page. In this case these are all the article pages on the Wikipedia selection.

Edge An existing hyperlink (<A>-tag) from any page A to any page B. Only full anchor tag links were considered. Various images contain a map which is capable of linking to other pages. But for the analysis and the study these links were disabled.

4.1 Removed Nodes

Only article nodes were considered for this network. Hence, all other possible nodes, like for instance images were removed. These sites are dead ends for navigating as they lead nowhere else and because of this serve no purpose in the user study.

The second group of removed nodes is a group of SOS-Schools pages. *Wikipedia for Schools 2013* is compiled by SOS Children and aims to raise awareness for the poor conditions in many nations around the world. Unfortunately for navigating the network these pages act as trap doors, allowing the user to reach a new network with no way to get back.

Finally, I did not want the user to be able to use the category index as it allows the user to reach every single node via the hierarchical tree without actually looking for information and following edges leading from the start-node.

In order to not let a user reach a page that is not part of the extracted network all links leading to these pages were removed as explained in section 4.2. Over 500 potential nodes were not accepted into the network during the data scraping process.

4.2 Data Scraping

Since the *Wikipedia for Schools 2013* release is still young no official dataset was available. Hence, the extraction of the network was performed by me. At first, the nodes themselves were extracted. This was done by running through all html-files provided in the dump and looking for article type pages. As mentioned in section 4.1 a number of pages were not allowed into the network. Avoiding index or category nodes was easy because in the dump they are all in a single folder called index. SOS-Schools Project nodes on the other hand were mixed with normal articles. In order to filter them out the HTML-content was parsed looking for specific meta-tags. These were *Copyright © SOS Children* and *Copyright (c) SOS Children's Villages UK* because at least one of them was present on every node of that type. The names of all valid nodes were then extracted from the *¡title¿* element.

After the extraction of all relevant nodes their links were crawled to detect the edges within the network. This was, once again, done by parsing the DOM-tree of every node. All found links were cleaned by fixing the URL-encoding and correcting the relative path and were then compared to the pool of valid nodes. Using this method made sure every edge was found. As mentioned before the articles try to follow the Wikipedia Manual of Style[1] but in many cases do not enforce them in a very strict fashion. In regards of analysing edges this meant that each node (A) could have multiple edges to any node B. To represent this in the data, a multiplicity attribute was introduced indicating how many edges of the same type exist.

The last step of the data scraping process was to extract the categories shown in table 4.1. First the hierarchical tree had to be constructed. This was done in the same was as extracting the links. Since a parent always links to all his children a top-down approach was chosen starting from the root subject index node, adding all found child-categories into a frontier and recursively searching through them. Once this was completed the nodes were mapped to their categories. Again, the links in the category nodes were analysed and compared to the pool of valid article nodes. Secondly, each article node contains a short related subjects section at its top. This part was analysed and mapped to the pool of found category nodes. Analysis revealed that these two mappings did not fully overlap. Since neither mapping was incorrect they were merged and the conjoint mapping was used from here on in.

4.3 Categories

Quickly raised in the previous section, every node in the network is assigned to at least one and at most two categories. This category assignment has been performed by the staff that created the Wikipedia subset. These categories help to understand how the user navigates through the network.

Categories are structured in a hierarchical tree, meaning there are small amount of top level categories and each of them has a number of children. Note, layers intersect meaning that any node in the tree can have multiple parent nodes.

To simplify the analysis the tree was reduced to its top level. For every node the tree was traversed reversely and the every resulting top-level-ancestor was assigned to it.

[1] `http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style`

Top Level Category	Nodes Amount
Art	92
Business Studies	148
Citizenship	322
Countries	386
Design and Technology	270
Everyday Life	442
Geography	1288
History	822
IT	164
Language and Literature	212
Mathematics	261
Music	175
People	777
Religion	170
Science	1305

Table 4.1: **Dataset Categories** - Top level categories present in *Wikipedia for Schools 2013* and the amount of nodes they contain. Each individual article page is assigned between 1 and 2 categories. The hierarchical category-tree was traversed in reverse to extract the top levels. Hence, the sum of node amounts listed in this table is larger than the amount of actual pages.

Note that the sum of the node amounts listed in table 4.1 is far larger than the actual amount of nodes in the network, because of multiple categories for each node.

4.4 Network Metrics

Metric	Value
Nodes	5873
Edges	245128
Largest SCC	5573
Bi-Directional Edges	31666
Node Degrees	MIN: 1, MAX: 1834 (5874)*, AVG: 84.4762
Node In-Degrees	MIN: 0, MAX: 1631 (5873)*, AVG: 41.7381
Node Out-Degrees	MIN: 1, MAX: 376, AVG: 41.7381
Categories (Top Level)	15
Categories (Overall)	170
Categories per Node	MIN: 1 , MAX: 2, AVG: 1.16
Shortest Path Lengths	MIN: 0, MAX: 8, AVG: 3.0169
Clustering Coefficient	0.157847
Maximum Eigenvalue	121.0995
Average Betweenness	0.000325

Table 4.2: **Network Metrics** - The most important metrics for the network. Values marked by an asterisk include a node that is linked to by every other node in the network. The shortest path of 0 exists because multiple nodes have edges to themselves.

Table 4.2 lists the characteristic numbers for *Wikipedia for Schools 2013*. Note, the degree rows have two values for their maximum. This is caused by a unique node. Every node in the entire network has an edge to the node *Wikipedia Text on Creative Commons* because the information and imagery presented on each node is licensed under the Creative Commons license. As this skews the maximum values they were put in parentheses and the lower value, without this Creative Commons node, is a more accurate representation of the network. The maximum eigenvalue represents this very prominent node.

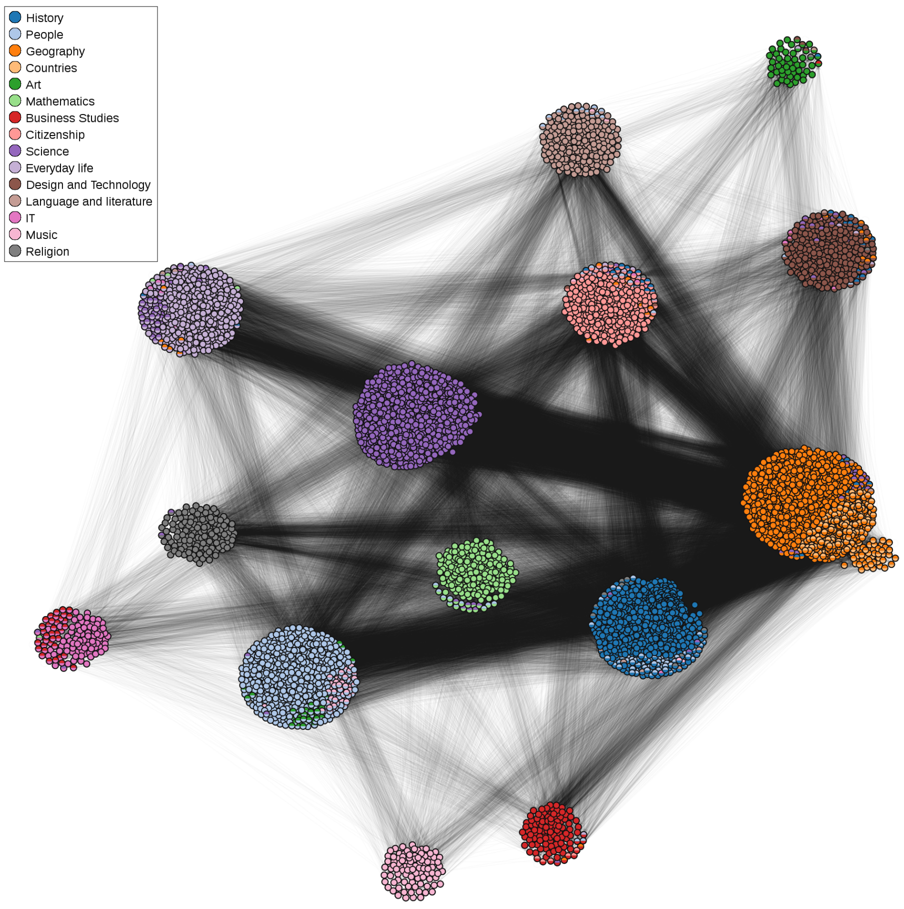


Figure 4.1: **Network Visualisation** - A visualisation of the network using the sfdp layout with *mu* = *3.0* to strengthen the connective force of nodes of the same connected component. Each colour represents a category and nodes having more than 1 colour are assigned to multiple categories. To visualise the category similarities edges where both nodes have multiple categories have a higher edge weight if all categories match. Doing so causes *Countries* and *Geography* to form one single cluster since the set of nodes in Countries is a subset of the set of node in Geography. The visualisation illustrates the focus of the network. Since it aims to match the UK National Curriculum the most strongly represented clusters are *Science*, *History* and *Geography*.

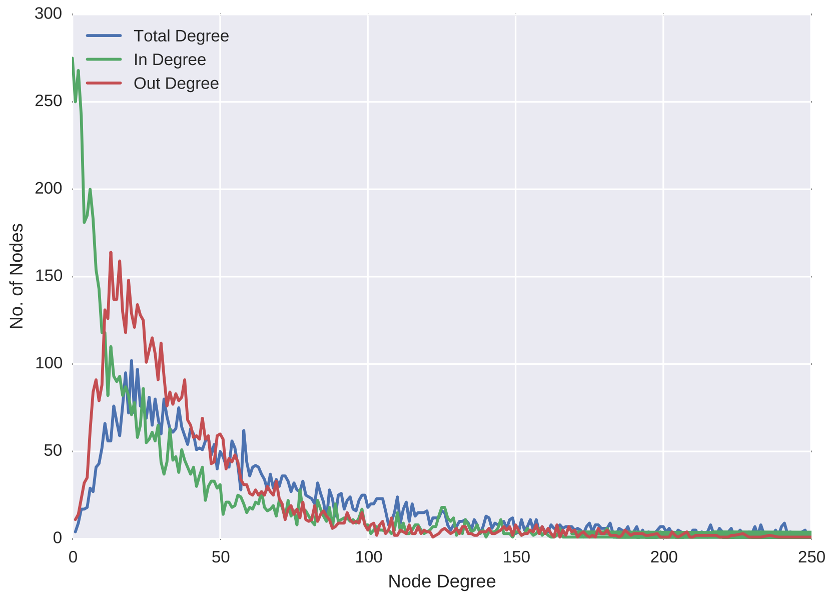


Figure 4.2: **Degree Distribution** - The different degree types visualised throughout the network. A high amount on nodes have a low in-degree. This is representative for nodes containing articles about very specific topics. The curve decreases rapidly following the power law. The out-degree graph on the other hand follows increases at first before then asymptotically submitting to the power law. Thus only few nodes have an out-degree of less than 5. The curve then rapidly reaches its peak at approximately 15 after which it slowly decreases again. Over 95% of all nodes have a total degree lower than 100 and nodes with a higher degree than that can be classified as hubs. These hubs networks (*Hub Highways*) are important for navigation. Note, there are nodes with degree values larger than 250 though they just prolong the power law-tendril in the visualisation.

First visualised in figure 4.1 and detailed in figure 4.2 and table 4.2 the network is well connected. Each node links, on average, to more than 40 other nodes. The connections are not just strong within the respective category but inter-category-edges are very common as well. Note, due to their strong relation every node in the category *Countries* is part of the category *Geography*. Hence figure 4.1 displays only 14 connected components despite there being 15 categories.

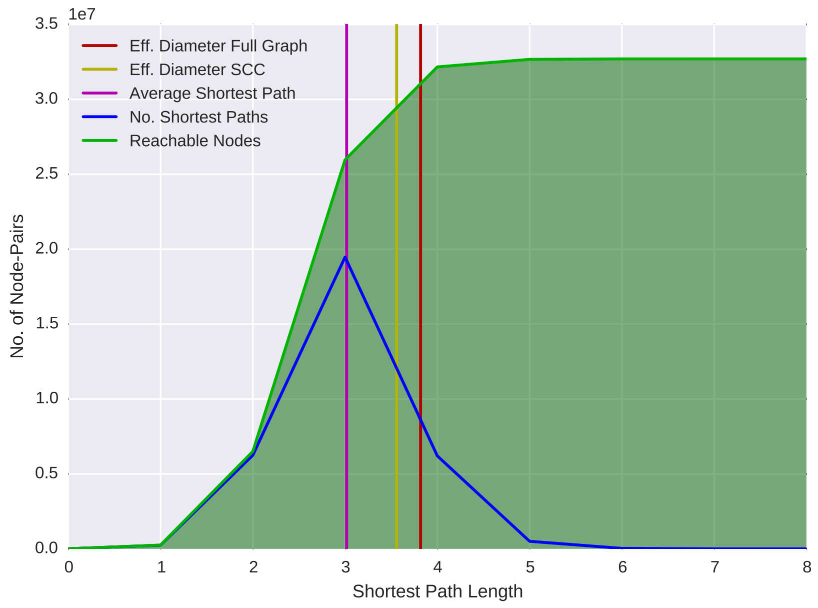


Figure 4.3: **Shortest Path Distribution** - The distribution of shortest path pairs within the network. Starting at a random node most other nodes can be reached within 1 to 5 hops. A peak can be noticed at a shortest path length of 3 which is very close to the average (3.0169). Since the network is directed with only 5573 of the 5873 nodes in its largest strongly connected component not all nodes are actually reachable rendering the diameter for the full graph to be infinite. The effective diameter, $\delta_{0.9}$, indicates the amount of hops needed to reach at least 90% of the nodes in the network. Incorporating the incompleteness of the graph this results in two values, the effective diameter for the full graph (3.8184) and for only the largest strongly connected component (3.5589). Thus, whilst the average distance is 3 more than 90% of all nodes can be reached with only 1 extra hop.

The distribution of shortest paths within the *Wikipedia for Schools 2013* is visualised in figure 4.3. On average individual pairs of nodes are approximately 3 hops away from each other. That being said; it is not guaranteed that there even exists a path between two given nodes. Since the network is directed a link from any node A to and node B does not necessarily mean there exists the same link in the other direction. The largest strongly connected component (SCC) consists of 5573 out of the overall 5873 nodes and hence, 300 nodes are unreachable from within resulting in 1787707 pairs of nodes where no possible path exists. Thus the effective diameter, $\delta_{0.9}$, indicating how many hops are required on average to reach 90% of the network, represents a more meaningful characteristic value. Within the connected component this distance is 3.5589. If the impossible paths are included to the total the average hops required increase to 3.8184. For a person navigating the network this means that despite there being almost 6000 nodes in the network it is very likely that navigating to any node in the network can be done in 4 hops or less.

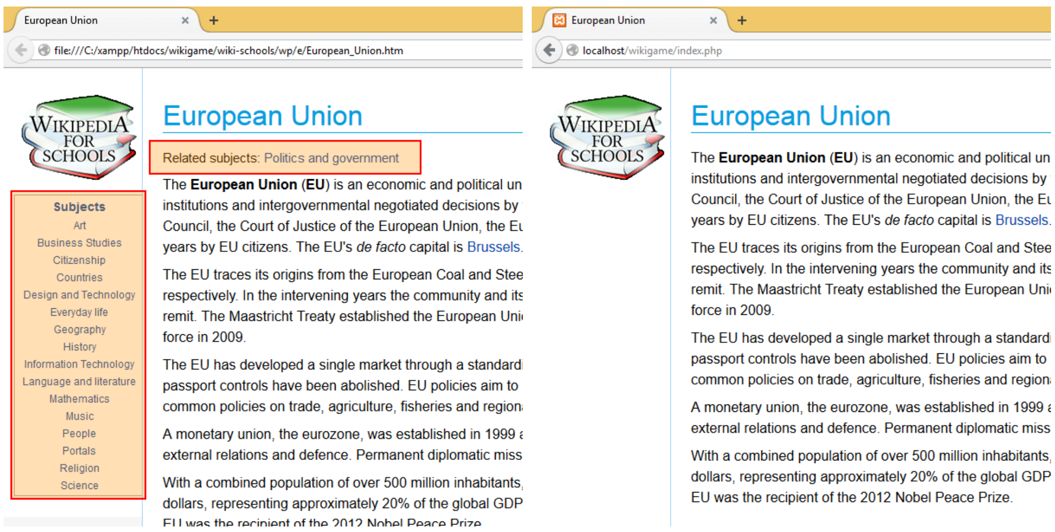
A high average degree and the short minimum distance between two nodes are indicative for a *Small World*-network. (Milgram, 1967) Since the degree distribution (figure 4.2) asymptotically follows the power law and the presence of hubs it can as well be described as a *scale-free* network. Scale-free networks are even more dense than those described by Milgram. The mathematical definition for such networks is the relation between the average shortest path distance and the amount of nodes. For scale-free networks this is defined as $L \propto \log \log N$ where L is the average shortest path and N is the total amount of nodes within the network. (Cohen, Havlin, and Ben-Avraham, 2003) The underlying network in *Wikipedia for Schools 2013* follows this rule by having 5873 nodes and an average shortest path of 3.0169.

4.5 Node Modifications

The section about data scraping 4.2 mentioned that some changes to the pages were made in order to allow for the study. This directly violates the initial idea of not touching the underlying network at all but these changes were essential because without them a user would often arrive in dead ends and trap doors atypical for an actual web environment.

Links to the aforementioned SOS-Schools pages were removed to avoid trapdoors. Secondly, static text passages have been removed from each page. These include links to the category overview form every page and a section linking to the same article on *Wikipedia*. The latter were removed as they serve no purpose and allow the user to escape the controlled study environment.

Next, links to image pages were removed. These pages are used to provide a higher resolution version of the image clicked, provides meta-data about the image such as dpi and the camera used and show copyright information. But they provide no additional information for the navigation process and the only way the user can go back is via the use of the browser history. Hence, in order to minimize user confusion these links were disabled. The original image is still visible, but is no longer clickable.


European Union

file:///C:/xampp/htdocs/wikigame/wiki-schools/wp/e/European_Union.htm

WIKIPEDIA FOR SCHOOLS

Subjects
Art
Business Studies
Citizenship
Countries
Design and Technology
Everyday life
Geography
History
Information Technology
Language and literature
Mathematics
Music
People
Portals
Religion
Science

European Union

Related subjects: Politics and government

The **European Union** (**EU**) is an economic and political un
institutions and intergovernmental negotiated decisions by
Council, the Court of Justice of the European Union, the Eu
years by EU citizens. The EU's *de facto* capital is Brussels.

The EU traces its origins from the European Coal and Stee
respectively. In the intervening years the community and its
remit. The Maastricht Treaty established the European Uni
force in 2009.

The EU has developed a single market through a standard
passport controls have been abolished. EU policies aim to
common policies on trade, agriculture, fisheries and region

A monetary union, the eurozone, was established in 1999 a
external relations and defence. Permanent diplomatic miss

With a combined population of over 500 million inhabitants,
dollars, representing approximately 20% of the global GDP
EU was the recipient of the 2012 Nobel Peace Prize.

European Union

localhost/wikigame/index.php

WIKIPEDIA FOR SCHOOLS

European Union

The **European Union** (**EU**) is an economic and political un
institutions and intergovernmental negotiated decisions by
Council, the Court of Justice of the European Union, the Eu
years by EU citizens. The EU's *de facto* capital is Brussels.

The EU traces its origins from the European Coal and Stee
respectively. In the intervening years the community and its
remit. The Maastricht Treaty established the European Uni
force in 2009.

The EU has developed a single market through a standard
passport controls have been abolished. EU policies aim to
common policies on trade, agriculture, fisheries and region

A monetary union, the eurozone, was established in 1999 a
external relations and defence. Permanent diplomatic miss

With a combined population of over 500 million inhabitants,
dollars, representing approximately 20% of the global GDP
EU was the recipient of the 2012 Nobel Peace Prize.

Figure 4.4: **Node Modification Screenshots** - The original node on the left and the by the tracker modified version on the right with their differences highlighted. The navigation on the left and related subjects at the top were removed. After the changes users are no longer able to reach every page by just navigating through a hierarchical tree.

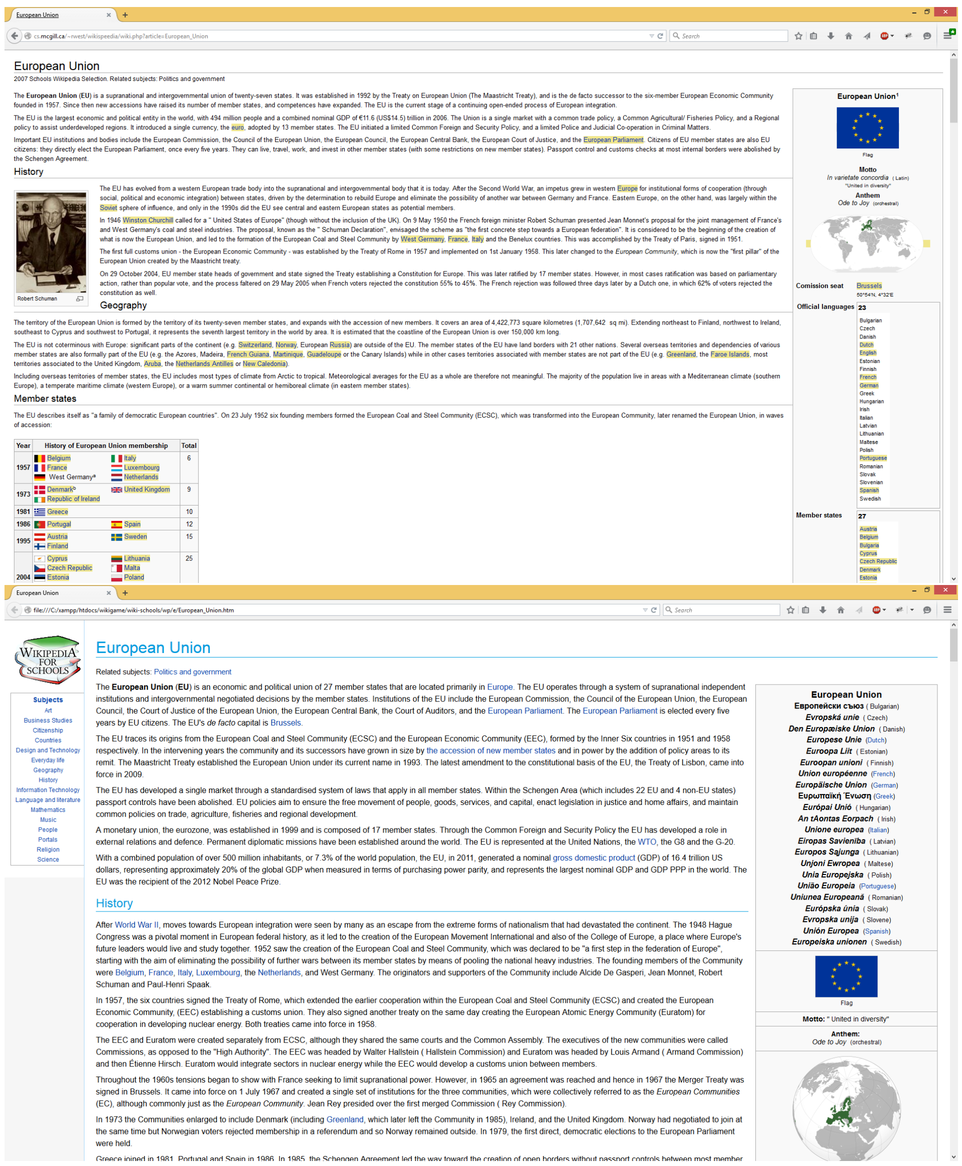
Finally the category index present on every site was removed. As already explained in section 4.1 users were not supposed to solve every mission by simply navigating through the same tree every time. Hence, the navigation on the left-hand side of the screen was removed. In figure 4.4 the final product is seen compared to its original form.

4.6 Comparison to previous Versions of Wikipedia for Schools

The Wikipedia for Schools dumps are a very common testing environment for this kind of user study. In the related works section some other studies are already mentioned. Over the years numerous versions of this dump have been released. For their research on the abandonment of navigation paths Scaria et al. use data collected by *Wikispeedia* which is based on a dump from 2007. (Scaria et al., 2014)

In terms of structure there were no significant changes over the years. Never versions generally consist of more articles and have some errors fixed. It has to be noted that some articles were renamed and others completely removed. This makes a direct mapping very difficult. The category hierarchy evolves over time. During the release of version 2007 and 2013 the most significant difference is the vast increase of science and technology related subjects.

The reason as to why *Wikipedia for Schools 2013* was chosen for this study described in the next chapter (5) can be seen in figure 4.5. For the user study explained in the next chapter I wanted to provide an environment that does not look outdated and goes along with what a user commonly finds on a live wiki on the Internet.


European Union
cs.mcgill.ca/~rwest/wikispeedia/wiki.php?article=European_Union
Search

European Union

2007 Schools Wikipedia Selection. Related subjects: Politics and government

The **European Union** (**EU**) is a supranational and intergovernmental union of twenty-seven states. It was established in 1992 by the Treaty on European Union (The Maastricht Treaty), and is the de facto successor to the six-member European Economic Community founded in 1957. Since then new accessions have raised its number of member states, and competences have expanded. The EU is the current stage of a continuing open-ended process of European integration.

The EU is the largest economic and political entity in the world, with 494 million people and a combined nominal GDP of €11.6 (US$14.5) trillion in 2006. The Union is a single market with a common trade policy, a Common Agricultural/ Fisheries Policy, and a Regional policy to assist underdeveloped regions. It introduced a single currency, the euro, adopted by 13 member states. The EU initiated a limited Common Foreign and Security Policy, and a limited Police and Judicial Co-operation in Criminal Matters.

Important EU institutions and bodies include the European Commission, the Council of the European Union, the European Council, the European Central Bank, the European Court of Justice, and the European Parliament. Citizens of EU member states are also EU citizens: they directly elect the European Parliament, once every five years. They can live, travel, work, and invest in other member states (with some restrictions on new member states). Passport control and customs checks at most internal borders were abolished by the Schengen Agreement.

History

Robert Schuman

The EU has evolved from a western European trade body into the supranational and intergovernmental body that it is today. After the Second World War, an impetus grew in western Europe for institutional forms of cooperation (through social, political and economic integration) between states, driven by the determination to rebuild Europe and eliminate the possibility of another war between Germany and France. Eastern Europe, on the other hand, was largely within the Soviet sphere of influence, and only in the 1990s did the EU see central and eastern European states as potential members.

In 1946 Winston Churchill called for a " United States of Europe" (though without the inclusion of the UK). On 9 May 1950 the French foreign minister Robert Schuman presented Jean Monnet's proposal for the joint management of France's and West Germany's coal and steel industries. The proposal, known as the " Schuman Declaration", envisaged the scheme as "the first concrete step towards a European federation". It is considered to be the beginning of the creation of what is now the European Union, and led to the formation of the European Coal and Steel Community by West Germany, France, Italy and the Benelux countries. This was accomplished by the Treaty of Paris, signed in 1951.

The first full customs union - the European Economic Community - was established by the Treaty of Rome in 1957 and implemented on 1st January 1958. This later changed to the *European Community*, which is now the "first pillar" of the European Union created by the Maastricht treaty.

On 29 October 2004, EU member state heads of government and state signed the Treaty establishing a Constitution for Europe. This was later ratified by 17 member states. However, in most cases ratification was based on parliamentary action, rather than popular vote, and the process faltered on 29 May 2005 when French voters rejected the constitution 55% to 45%. The French rejection was followed three days later by a Dutch one, in which 62% of voters rejected the constitution as well.

Geography

The territory of the European Union is formed by the territory of its twenty-seven member states, and expands with the accession of new members. It covers an area of 4,422,773 square kilometres (1,707,642 sq mi). Extending northeast to Finland, northwest to Ireland, southeast to Cyprus and southwest to Portugal, it represents the seventh largest territory in the world by area. It is estimated that the coastline of the European Union is over 150,000 km long.

The EU is not coterminous with Europe: significant parts of the continent (e.g. Switzerland, Norway, European Russia) are outside of the EU. The member states of the EU have land borders with 21 other nations. Several overseas territories and dependencies of various member states are also formally part of the EU (e.g. the Azores, Madeira, French Guiana, Martinique, Guadeloupe or the Canary Islands) while in other cases territories associated with member states are not part of the EU (e.g. Greenland, the Faroe Islands, most territories associated to the United Kingdom, Aruba, the Netherlands Antilles or New Caledonia).

Including overseas territories of member states, the EU includes most types of climate from Arctic to tropical. Meteorological averages for the EU as a whole are therefore not meaningful. The majority of the population live in areas with a Mediterranean climate (southern Europe), a temperate maritime climate (western Europe), or a warm summer continental or hemiboreal climate (in eastern member states).

Member states

The EU describes itself as "a family of democratic European countries". On 23 July 1952 six founding members formed the European Coal and Steel Community (ECSC), which was transformed into the European Community, later renamed the European Union, in waves of accession:

Year	History of European Union membership		Total
1957	Belgium France West Germany[a]	Italy Luxembourg Netherlands	6
1973	Denmark[b] Republic of Ireland	United Kingdom	9
1981	Greece		10
1986	Portugal	Spain	12
1995	Austria Finland	Sweden	15
2004	Cyprus Czech Republic Estonia	Lithuania Malta Poland	25

European Union[1]

Flag

Motto
In varietate concordia (Latin)
"United in diversity"

Anthem
Ode to Joy (orchestral)

Comission seat	Brussels 50°54′N, 4°32′E
Official languages	23 Bulgarian Czech Danish Dutch English Estonian Finnish French German Greek Hungarian Irish Italian Latvian Lithuanian Maltese Polish Portuguese Romanian Slovak Slovenian Spanish Swedish
Member states	27 Austria Belgium Bulgaria Cyprus Czech Republic Denmark Estonia

European Union
file:///C:/xampp/htdocs/wikigame/wiki-schools/wp/e/European_Union.htm
Search
WIKIPEDIA FOR SCHOOLS

Subjects
Art
Business Studies
Citizenship
Countries
Design and Technology
Everyday life
Geography
History
Information Technology
Language and literature
Mathematics
Music
People
Portals
Religion
Science

European Union

Related subjects: Politics and government

The **European Union** (**EU**) is an economic and political union of 27 member states that are located primarily in Europe. The EU operates through a system of supranational independent institutions and intergovernmental negotiated decisions by the member states. Institutions of the EU include the European Commission, the Council of the European Union, the European Council, the Court of Justice of the European Union, the European Central Bank, the Court of Auditors, and the European Parliament. The European Parliament is elected every five years by EU citizens. The EU's *de facto* capital is Brussels.

The EU traces its origins from the European Coal and Steel Community (ECSC) and the European Economic Community (EEC), formed by the Inner Six countries in 1951 and 1958 respectively. In the intervening years the community and its successors have grown in size by the accession of new member states and in power by the addition of policy areas to its remit. The Maastricht Treaty established the European Union under its current name in 1993. The latest amendment to the constitutional basis of the EU, the Treaty of Lisbon, came into force in 2009.

The EU has developed a single market through a standardised system of laws that apply in all member states. Within the Schengen Area (which includes 22 EU and 4 non-EU states) passport controls have been abolished. EU policies aim to ensure the free movement of people, goods, services, and capital, enact legislation in justice and home affairs, and maintain common policies on trade, agriculture, fisheries and regional development.

A monetary union, the eurozone, was established in 1999 and is composed of 17 member states. Through the Common Foreign and Security Policy the EU has developed a role in external relations and defence. Permanent diplomatic missions have been established around the world. The EU is represented at the United Nations, the WTO, the G8 and the G-20.

With a combined population of over 500 million inhabitants, or 7.3% of the world population, the EU, in 2011, generated a nominal gross domestic product (GDP) of 16.4 trillion US dollars, representing approximately 20% of the global GDP when measured in terms of purchasing power parity, and represents the largest nominal GDP and GDP PPP in the world. The EU was the recipient of the 2012 Nobel Peace Prize.

History

After World War II, moves towards European integration were seen by many as an escape from the extreme forms of nationalism that had devastated the continent. The 1948 Hague Congress was a pivotal moment in European federal history, as it led to the creation of the European Movement International and also of the College of Europe, a place where Europe's future leaders would live and study together. 1952 saw the creation of the European Coal and Steel Community, which was declared to be "a first step in the federation of Europe", starting with the aim of eliminating the possibility of further wars between its member states by means of pooling the national heavy industries. The founding members of the Community were Belgium, France, Italy, Luxembourg, the Netherlands, and West Germany. The originators and supporters of the Community include Alcide De Gasperi, Jean Monnet, Robert Schuman and Paul-Henri Spaak.

In 1957, the six countries signed the Treaty of Rome, which extended the earlier cooperation within the European Coal and Steel Community (ECSC) and created the European Economic Community, (EEC) establishing a customs union. They also signed another treaty on the same day creating the European Atomic Energy Community (Euratom) for cooperation in developing nuclear energy. Both treaties came into force in 1958.

The EEC and Euratom were created separately from ECSC, although they shared the same courts and the Common Assembly. The executives of the new communities were called Commissions, as opposed to the "High Authority". The EEC was headed by Walter Hallstein (Hallstein Commission) and Euratom was headed by Louis Armand (Armand Commission) and then Étienne Hirsch. Euratom would integrate sectors in nuclear energy while the EEC would develop a customs union between members.

Throughout the 1960s tensions began to show with France seeking to limit supranational power. However, in 1965 an agreement was reached and hence in 1967 the Merger Treaty was signed in Brussels. It came into force on 1 July 1967 and created a single set of institutions for the three communities, which were collectively referred to as the *European Communities* (EC), although commonly just as the *European Community*. Jean Rey presided over the first merged Commission (Rey Commission).

In 1973 the Communities enlarged to include Denmark (including Greenland, which later left the Community in 1985), Ireland, and the United Kingdom. Norway had negotiated to join at the same time but Norwegian voters rejected membership in a referendum and so Norway remained outside. In 1979, the first direct, democratic elections to the European Parliament were held.

Greece joined in 1981, Portugal and Spain in 1986. In 1985, the Schengen Agreement led the way toward the creation of open borders without passport controls between most member

European Union
Европейски съюз (Bulgarian)
Evropská unie (Czech)
Den Europæiske Union (Danish)
Europese Unie (Dutch)
Euroopa Liit (Estonian)
Euroopan unioni (Finnish)
Union européenne (French)
Europäische Union (German)
Ευρωπαϊκή Ένωση (Greek)
Európai Unió (Hungarian)
An tAontas Eorpach (Irish)
Unione europea (Italian)
Eiropas Savienība (Latvian)
Europos Sajunga (Lithuanian)
Unjoni Ewropea (Maltese)
Unia Europejska (Polish)
União Europeia (Portuguese)
Uniunea Europeană (Romanian)
Európska únia (Slovak)
Evropska unija (Slovene)
Unión Europea (Spanish)
Europeiska unionen (Swedish)

Flag

Motto: " United in diversity"

Anthem:
Ode to Joy (orchestral)

Figure 4.5: **Wikipedia for Schools 2007 vs 2013** - The 2 versions of Wikipedia for Schools compared. *Wikipedia for Schools 2007*, at the top, lacks a navigation entirely whilst *Wikipedia for Schools 2013* at the bottom resembles the actual live *Wikipedia* in many aspects.

5 User Study

To evaluate the potential of the new tracking features a user study was conducted using the dataset described above. Ensuring comparability was a big concern. Hence, a common study type was chosen. This chapter serves as an overview over the study, how it was performed, how the specific parameters were created and provides an overview over the participants. The results of the study can be found in section 6.

5.1 Idea

As a first study it was important that it took place in a controlled environment. The goal was to evaluate the new features and not test the tracker for stability on sites that could potentially attack it using JavaScript. Hence, the dataset described in section 4 was used. As for the study method I wanted to be able to compare my results with existing studies. West, Pineau, and Precup first described it in his paper on Wikispeedia. There each user was given missions to complete and each of these missions consists of given start and target pages. The user then had to navigate through the network using only the links available on the current site. In their paper only links were traced as the research goal was different. (West, Pineau, and Precup, 2009) Since the path through the network is recorded by this new tracker the path through the network and the completion or abandon rates were the base for comparison of the initial results. When the old features behave the same way as they do in related literature the results of the new features can be compared with them.

5.2 Study Process

Before beginning the recording process each user was first made familiar with their task, the trackers user interface and the underlying Wikipedia network. This was done by providing two sample missions, the first one from *Ben Affleck* to *Curium* and the second one from *Wars of the Roses* to *Apollo*. The first mission aimed to provide the overview over the task at hand and for their second training mission the participants were instructed to make use of the abort-button to make sure they were aware that not completing any specific mission was an option. Whilst finishing this training set users were allowed to ask questions.

Mission #	Start	Target
1	Krakatoa	São Tomé and Príncipe
2	Rococo	Tourette Syndrome
3	Bed Bug	Winter Olympic Games
4	Arnold Schwarzenegger	Damascus
5	Second Boer War	Lizard
6	Bran	Abugida
7	Adam Brody	Plains Zebra
8	Coconut Oil	Child Abandonment
9	Scottish Highlands	Eukaryote
10	Angelina Julie	1952
11	Eustreptospondylus	The Boat Race
12	Benjamin Franklin	Suriname
13	Soil pH	Zionism
14	French Southern and Antarctic Lands	Irrational Number
15	Meteoroid	Magpie

Table 5.1: **The Mission Set** - The list of missions each participant was asked to complete. For every mission the start page was loaded and the user had to find a route to the target.

During the study the participating users were asked to complete a set consisting of 15 missions. In order to compare the individual users each user was given the same set found in table 5.1. How each of them was generated is explained in the next subsection. Since this amount of missions can take a long time users were allowed to interrupt and abort the study at any time. Hence, not every participant completed the full set. When a session was aborted only the fully recorded missions, completed or aborted, were taken into consideration. During the entire recording phase users were not allowed to ask questions about the network. Only technical issues were answered.

After completing the missions the study participants were asked to provide feedback on the tracker and to not disclose any information about the *Wikipedia for Schools 2013* network and the individual missions to any other study participant, because they would be given the same set of missions.

5.2.1 Expectation

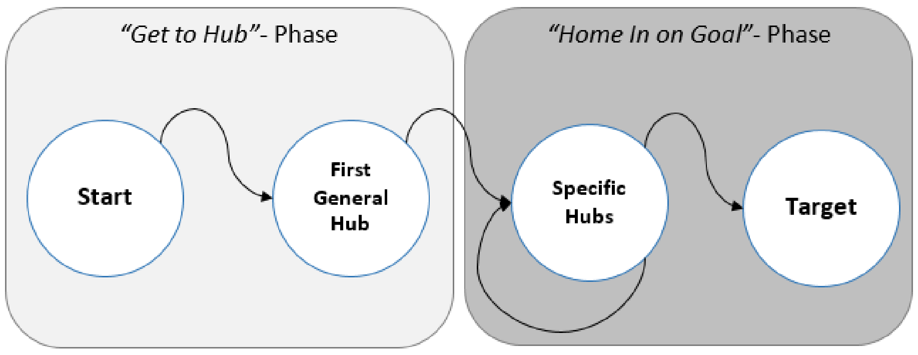


Figure 5.1: **Navigation Expectations** - A user's expected path through the network. First a general hub is being searched. Following other more and more specific hubs the user navigates to the right target category homing in on the target and finally jumps to the target page.

Figure 5.1 path outlines how a user was expected to tackle a mission. When first thrown onto the start page the initial step is to figure out the current location within the network. This is done by reading the text. Then the first course of action is to head for a large page about a known topic and is known as the *get away*-phase. These pages are high degree nodes, here on forth called hubs, and represent important nodes within the network and often link to other hubs creating highways through the network. (West, Pineau, and Precup, 2009)

Once a user has arrived at the first hub about a known topic search will now begin for a hub related to the target page. Following the hub-highways through first general and then specific hubs the user will eventually reach a specific hub related to the target-page. He then searches for the link to the actual target. West, Pineau, and Precup call this phase the *home in on goal*-phase.

The longer the user looks for a specific hub or the target link the higher the risk of a mission abortion becomes. After initial self-tests I expected users to reach a specific hub containing a link to the goal page, but often fail to find the final link meaning at the time of mission-abortion they are only a single hop away.

After completing a game or a sub-path it is to be expected that the user remembers his route. The next time a mission whose goal page is in the same category comes up he will try to follow the same route for as long as possible. In missions whose goal is any kind of animal he will most likely always go for the *Animalia*-hub as fast as possible and then search from there. The expectation here is that the user will do so even if there is a shorter path present because staying in known territory is more likely to lead to the target than to venture into the unknown.

5.2.2 Mission Generation

Each mission from the study set listed in table 5.1 followed a defined set of characteristics. This was important so that individual missions could be compared. Primary parameter was the distance from start to finish. This was done by calculating the shortest path. Secondly the shortest path was only allowed to use normal text-links in the headlines or flow-text. Info boxes, icons, commonly used for nations, were still clickable by the user but the shortest path had to be able to be completed using just standard links. Following these characteristics each mission was solvable the same way shifting the focus on the links themselves. The minimum visited nodes along the shortest path from start to finish was set to 4 meaning completing each mission involved at least 3 hops. This distance was set so that the expectation of visiting at least 2 hubs, a general and a specific one, could be met. Additionally all missions were set to be distinct, each individual start or goal page would appear only once. Due to the limited size of the network the nodes along the shortest path are not distinct.

The actual generation was done by building a set of candidates that each follow the aforementioned criteria about distance and links. From this set the missions were picked at random making sure no start or goal page appeared twice. Hence, there is no connectivity between two consecutive missions.

5.2.3 Study Participants

For the study a set of 27 participants was chosen. The aim was to have a spectrum of adults who have at least some experience with computers and the internet. Since the Wikipedia network is entirely in English it was indispensable that each participant had a solid understanding of the language. Despite of that the study was not restricted to users whose mother tongue was English.

Participants were between 19 and 41 years of age with over 50% between 22 and 27. The level of education ranged from secondary level to masters degrees. Prior to starting the recording session each user was asked to rate their own computer and internet experience on a scale from 1 to 10. For the computer experience all users rated themselves above average (5). In terms of internet experience the rating was mostly identical. Only 3 users differed in their rating. One participant giving itself a higher score in internet experience and two giving themselves a lower rating.

Finally users were asked whether they had enough sleep the night before, whether they wear glasses and if they had any caffeine containing beverages before the study. 10 participants usually wear glasses and 7 wear contact lenses. Two of them did not wear them during the study. All claimed to have had enough sleep and 15 stated to having drank coffee or other caffeine-containing beverages, such as energy drinks on the day of the study.

Each participating user completed at least 4 and at most all 15 missions. Added up this combines to 197 completed missions. These missions were either completed or manually aborted because the user thought unable to find the target page. During one session connection issues rendered the recordings of a single mission worthless. Hence the results shown in chapter 6 are based on the remaining 196 recorded missions.

6 Results

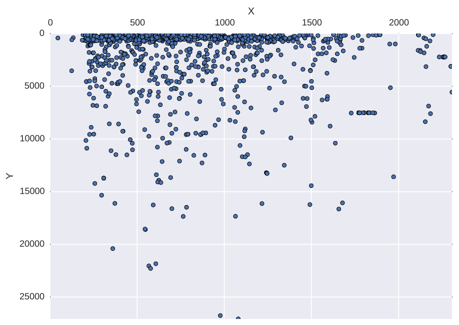
The study described in chapter 5 using the tracking system explained in chapter 2 provided data about a user's navigation patters in a wiki-based network. This section visualises the capabilities of this new data and tries to reason about a user's behaviour. Next the prevalent features will be analysed and checked if new features can explain the data extracted from the old features. Before drawing the final conclusion the potential issues regarding the study methodology, like *gamification*, will be outlined. The conclusion then tries to evaluate the viability of these new features and locates anchor points for future research, which is explained in chapter 7.

6.1 New Navigational Features in Action

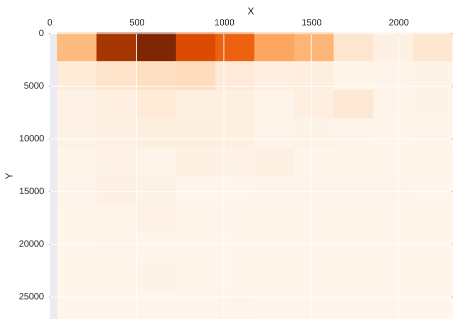
Focus of this work lies on the explanation and evaluation of the new features from chapter 2. Four experiments were performed in order to visualise the results. The first two focus on the location of clicks. The questions were where users click in general and where were the lucrative, non-lucrative or neutral clicks. Second major focus was the analysis of how much of any visited page was actually seen by the user before the page was left and finally a closer look was taken at the actual links that were clicked during the recording sessions.

6.1.1 Click Location Analysis

The first step of my analysis was to take a look where users clicked during their recording session. Figure 6.1 shows the X and Y coordinates of each click performed. 6.1a marks each individual click and 6.1b collects them



(a)



(b)

Figure 6.1: **Click Scatter** - Aggregated user clicks during the recording sessions. Values are taken from the X and Y coordinates of the clicks with [0,0] being the top left corner.

into a heat map. The darker the area the more clicks were performed in the respective area. This data suggests that links positioned at the top of each page get clicked more frequently than links further down. Note that there is a distinct empty space in the scatter figure along its left side. This is space reserved for the side navigation. This navigation has been removed (seen in figure 4.4) in order to stop the user from performing the same hierarchical navigation to complete every single mission.

Whilst this data already suggests a favouritism for higher positioned links a user's viewport plays a huge role in the assessment of user clicks. Figure 6.2 takes the same data and combines it with information about this viewport. Using the width and height of a browser window the page was transformed into viewport-frames. These frames form a grid on the page's content where each cell being the size of the user's browser window. Since the Wikipedia network used for the study does not have a fixed width the *X-scroll-offset* is always 0 and the viewport width is identical with the page's content width. Hence, each cell spans across the full width of the page. The final calculation as to in which viewport-frame the user has clicked has been reduced to the following: *floor(Y-scroll-offset / viewport-height)*. The viewport frame for each click was calculated with this formula as well. Figure 6.2 visualises the results. It is noticeable that over 60% of all clicks performed were within the

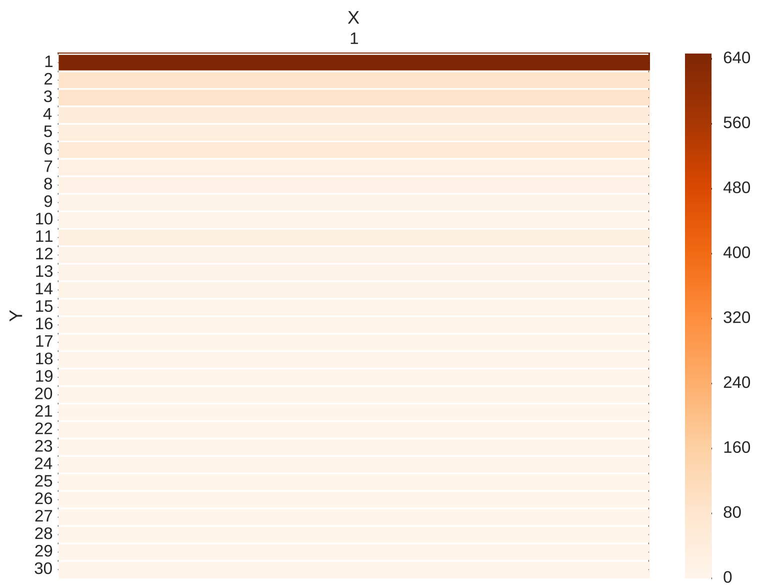


Figure 6.2: **Clicks per Viewport-Frame** - Each click was assigned to its viewport frame and plotted as a heat-map. The colour markers indicate the click frequency for each frame. Since the width of all pages in the network are dynamic each frame covers the entire width.

topmost viewport-frame. This reinforces the hypothesis that users click on what they see at the moment the website is loaded. Additionally it suggests that in a Wikipedia style network more general links are at the top of the page. The hub-highways explained in section 5.2.1 therefore usually have their links inside the first frame. The data suggests that when a link should be followed it should be positioned at the top. For finding information it does not necessarily mean that a link at the top is a good one for finding the target. The next subsections looks at the space covered and rates of lucrativeness for each individual click.

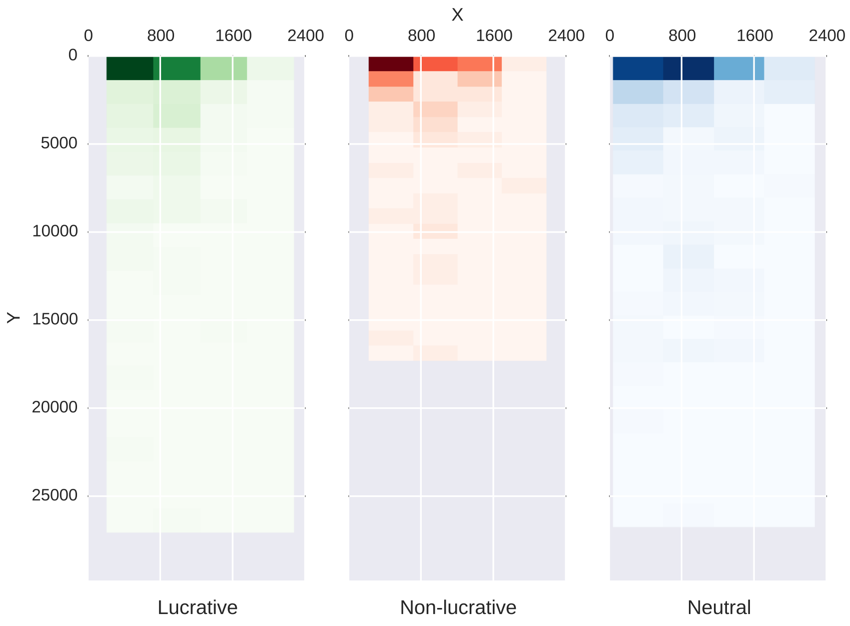


Figure 6.3: **Lucrativeness Heatmaps** - Each type of click (lucrative, non-lucrative and neutral) displayed individually. At first glance it is noticeable that users prefer to click on links in the top left corner of the website. All 3 click-types show very little difference. Thus, the position can be used to estimate a link's importance but not its quality.

6.1.2 Lucrative Page Areas

Each click was put into one of three categories: lucrative, non-lucrative or neutral. When the click reduced the distance to the target it was deemed lucrative. If the target distance increased then it was non-lucrative, a failure and if the distance remained the same it was a neutral click. Of all the clicks performed 31.457% were lucrative, 10.928% non-lucrative and the remaining 57.595% were classified as neutral, neither bringing the user closer to the target nor pushing him further away. In short this means over 89% of all clicks brought the user closer to the target or at least put it somewhere else the same distance away.

Figure 6.3 shows the click heat-map split up for each of the three click types.

All 3 types concentrate on the top. This was to be expected considering the overall click distribution and states that the click-coordinates alone do not provide enough information to evaluate a click.

6.1.3 Page Coverage

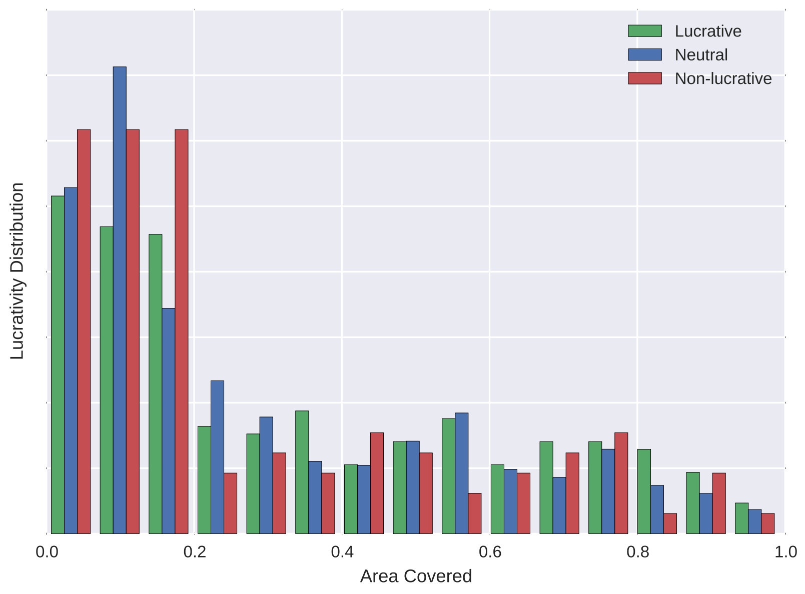


Figure 6.4: **Coverages vs Success** - Comparison of page coverage and click lucrativeness for each individual hop. All pages having only a single viewport-frame were removed for this analysis. The height of a bar indicates its contribution to the total amount of click of that type. A very low coverage (< 0.2) has the highest contribution to non-lucrative clicks. Thus, reading more content on a page results in a higher probability for a successful navigation.

As a next step the area seen by the user was analysed. This was performed by keeping track of every area of the page's content the user visited prior to leaving the page. The resulting page-coverage was then added to the click. Figure 6.4 illustrates which area covered contributed how much to the respective click class. Since pages consisting of only a single viewport-frame

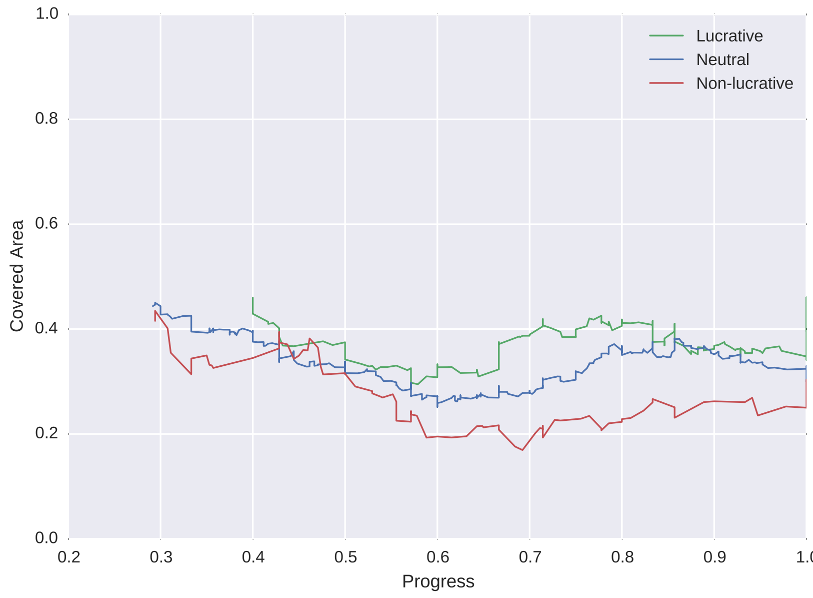


Figure 6.5: **Mean Coverage and Success during Missions** - Mean Page coverage and click-lucrativeness rate over the course of a mission. The line indicating lucrative clicks is consistently above the other types. This reinforces the assumption that higher coverage results in more lucrative clicks.

would always show 100% coverage, they were removed from this analysis in order to not skew the results. It is noticeable that very low coverage contributes more to the group of non-lucrative clicks than to the lucrative class. Hence, reading more content of the page and therefore increasing the coverage before choosing a link to follow results in a reduced probability of a non-lucrative click.

Links to more specific pages are in most cases not positioned at the top. Hence, once the user is looking for the first hub and the very specific target page higher page coverage was observed. Figure 6.5 visualizes this observation. The Y axis symbolises the percentage how much of a page was seen prior to leaving and the X axis marks the mission progress. For this progress every performed hop was considered, not just lucrative clicks. 1.0 marks the end of the mission. This does not necessarily mean successfully completed because a manual abortion as well counts as completion. The only time higher coverage results in more non-lucrative clicks than any other class is during 40% and 50% of the mission progress. This was in most cases caused by the user reaching the first hub on the highway and then essentially taking a wrong turn ending up in a wrong section. During the same period backtracks spike as well suggesting that the user realised he had taken a wrong link and returning to the previous site. Apart from this one instance higher coverage consistently causes a higher success probability.

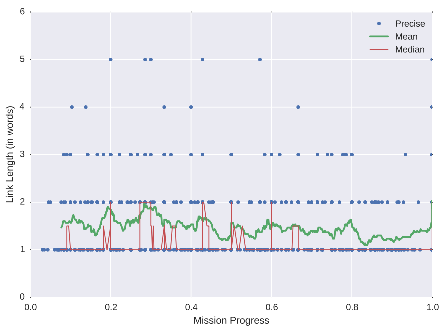
This coverage analysis can be used to construct an updated version of the dataset's network from chapter 4. All links that were in areas not covered by the user could be removed. The resulting graph would then no longer be the actual network, but the network as seen by the user.

6.1.4 Link Analysis

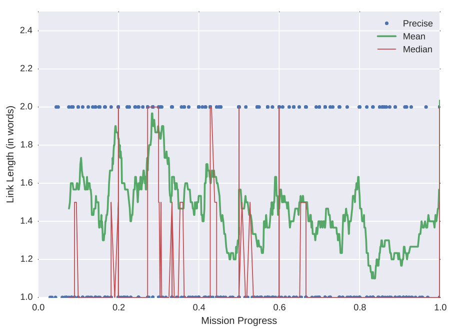
The final step of the new feature analysis was to look at the links clicked during the recording sessions. Since most links contain a 1 or 2 words long anchor texts it was assumed that the users click on links of that length. As users were only able to click on text links and could not navigate the network by any other means, like for instance linking images, this analysis became possible.

Figure 6.6 visualises the results of the analysis. The mean stays between 1 and 2 at all times reinforcing the assumption. During the first half of a mission an orientation towards 2 word long links can be noticed. During this phase the user looks for the first node on the hub-highway towards the target. The largest nodes in the network, having the highest degrees are United States and United Kingdom. Note that both of these pages have a 2 words long title and, hence, 2 words long anchor texts in their respective links. Since the nodes with the highest in degree are the easiest ones to be reached they often serve as the aforementioned entry page to the hub-highway and therefore cause the orientation of the mean link length during that time.

Whilst on the hub highway users visit general high degree nodes. These general nodes generally have 1-word titles and links causing the mean link



(a)



(b)

Figure 6.6: **Link Lengths** - Average length of links clicked throughout the sessions. (a) showing the full spectrum and (b) zooms in to better analyse the averages. The Y axis represents the length of a link's anchor text and the X axis shows the mission progress.

length to drop. One last spike can be noticed at the end. Due to some target pages having 2 words long titles the users were forced to click on longer links in order to actually complete the mission. Because of this involuntary click that spike in the graph should be ignored.

6.2 Prevalent Features

6.2.1 Time Spent

How much time a user spends on any page is influenced by a multitude of variables. When the user believes to know which direction to go the time spent should decrease. Additionally personal preferences play a role. Using the built in browser search allows for a much faster analysis of the current page. Figure 6.7 shows that most clicks were performed within the first 100 seconds. It is noticeable that for the shortest time-frame the contribution to the total count of non-lucrative clicks is lower than for all other classes. Hence, spending more time on a page results in a higher chance of a non-lucrative click. A possible explanation for this behaviour is that, as previously stated, when a user knows where to go the page is left rapidly.

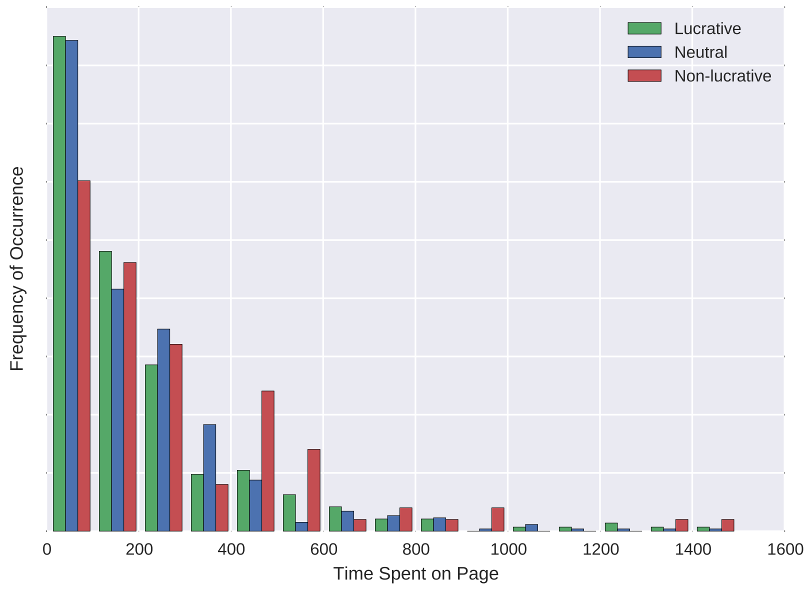


Figure 6.7: **Time Spent per Page** - Time spent on a page prior to clicking a link. The X axis shows the time spent in seconds and Y shows the frequency. The values are normalised so the height of the bar indicates the contribution to the total amount of clicks of that type.

A comparison of these values and the page coverage values shows that a longer stay on a site does not necessarily increase the coverage. Comparing lucrative clicks using the Pearson correlation shows $r = 0.0819$ and $sig = 0.1654$ meaning these two variables have a very weak correlation (Figure 6.8). This result can once again be explained by differences in a user's browsing behaviour. Skimming for links causes a high coverage in a short amount of time and when being distracted the opposite occurs.

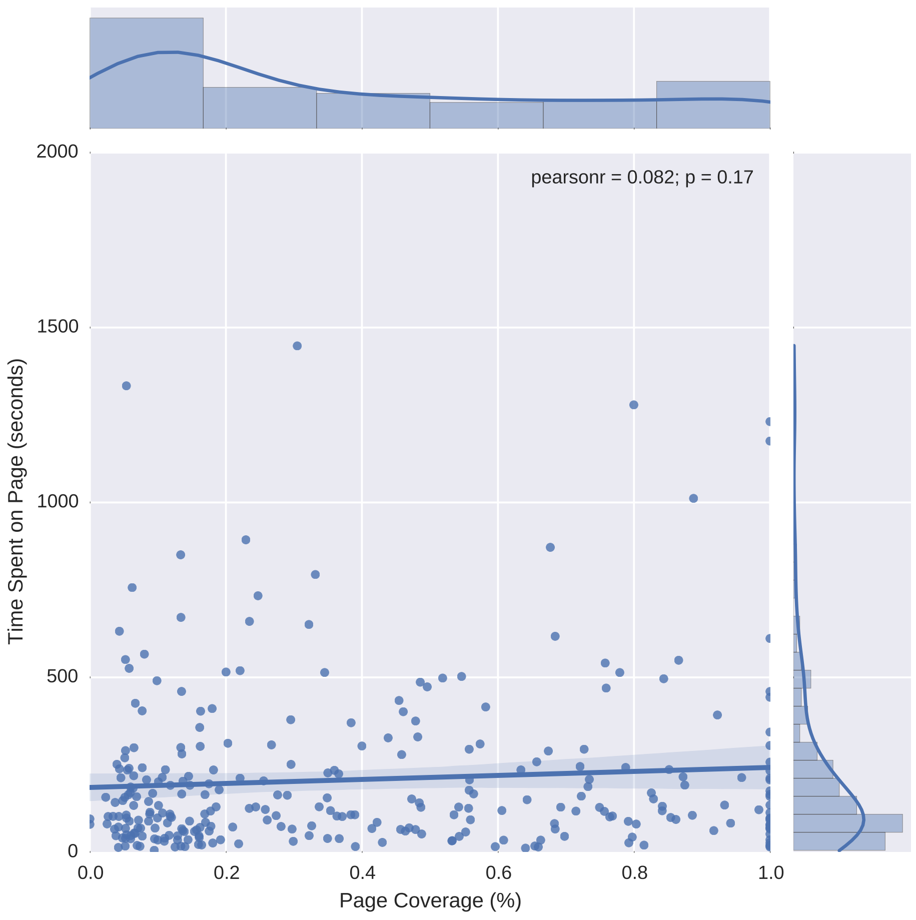


Figure 6.8: **Time/Coverage Correlation** - The correlation of the time users spend on a page before clicking on a link and the percentage of the page they have seen during that time. During the study a very weak correlation was noticed. This means that spending more time on a page does not necessarily mean seeing more of the page. On the other hand usage of browser built-in search functions cause a high coverage in a short amount of time.

6.2.2 Mission Success

Overall a success rate of 78% was recorded. In comparable experiments a success rate of only 49% was achieved. (Scaria et al., 2014) An explanation for this difference is the fact that for the study in this thesis users were presented with a set. Hence, every user completed a multitude of missions eventually learning the network and over time increasing their performance.

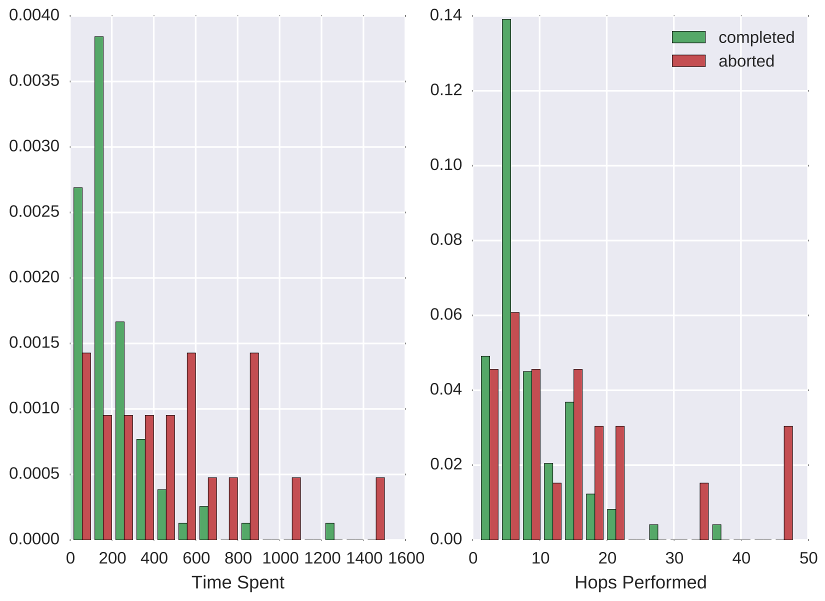


Figure 6.9: **Success vs Time and Hops** - Success and Abortion rates for missions. The left plot shows the timespan in seconds and the right one indicates the total amount of hops before a mission was completed or aborted. The data has been normalised and therefore a bar shows the contribution to the total amount of a class.

It was expected that the more time is spent on a mission the more likely it gets aborted. Figure 6.9 shows that the majority of successfully completed missions were finished in less than 10 jumps and took less than 5 minutes (300 seconds). Once these values are exceeded a user is more likely to abort than to complete a mission. This is because the user gets annoyed or actually believes being unable to find a way to the target.

Figure 6.10 visualises the success and abortion rates per hop. As already

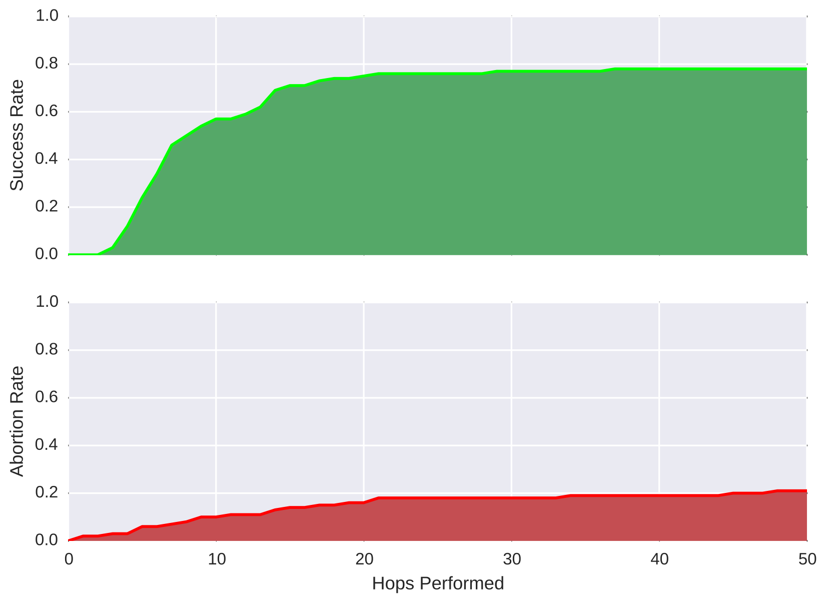


Figure 6.10: **Success Rates** - Success and Abortion rates over hops. The Y-axis indicates the percentage of missions completed or aborted after X amount of hops

mentioned most users completed the missions within 10 hops (60% of all missions). For the abortion rate one can note a consistent increase until hop 20. This results, on average, in a 5% abortion rate per hop. Since after the 20th hop over 90% of all missions are either completed or aborted the rates up to that mark provide a good indicator for the overall user behaviour. The abortion rate is noticeable lower than comparable experiments as they showed a 10% drop-out rate per click.(West and Leskovec, 2012). Users learning the network and improving their navigation patters over the course of a recording session and gamification (6.3) are the most probable causes.

When aborting a mission is was expected that the user was very close to the target. (Section 5.2.1) Figure 6.11 confirms this expectation. The average distance from the target was 1.647 with a median of 1.0. Users were, when cancelling a mission, usually only a single hop away from the target. This

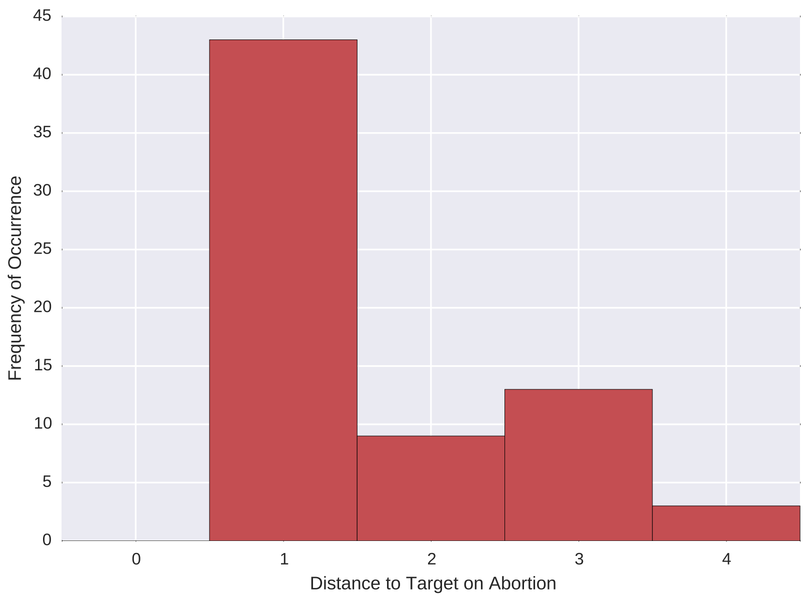


Figure 6.11: **Abortion Target Distance** - The distance to the target upon abortion. In most cases users were very close to the target when cancelling a mission. A distance of 1 states that the link to the final node could not be found despite being on a page containing at least 1 correct link.

means that they were on a site containing a link to the final objective but could not find it. In two cases on the other hand users went in the completely wrong direction. A target distance of 4 or higher means that they were further away from the target than they started out on. Abortions here suggest that the user was hopelessly lost or lacked knowledge of the target category altogether. Naturally, no case exists where a user aborted with a distance of 0 as the target is already reached at that point.

6.3 Issues

During the construction of the tracker and the analysis of the study questions naturally arose, most notably the issue of gamification. When put into a controlled environment a user behaves differently than out in the wild. Fully aware of being observed the user tries to perform as good as possible. As to whether that means finding the target in the shortest amount of time or in the least clicks varies on a user-to-user basis. But since the user always exactly knows which page to go to attempts to optimise are common, essentially turning a recording session into a game for the user. Reference systems like "Wikispeedia: An Online Game for Inferring Semantic Distances between Concepts." even have the terms game and speed in their titles further reinforcing that notion. In "Find it if you can: a game for modeling different types of web search success using interaction data" a target page is not specified. The goal is to acquire the target information on any site within the network. The user is then required to enter the URL of the page they found the information on. (Ageev et al., 2011) Whilst this removes the problem of having a single target page new ones arise. Each page has to be manually annotated in order to abstract the information available on them. Otherwise manual analysis of the user submitted URLs is required. Either option takes a lot of time and manpower. Hence, the single-target-page model was kept.

Secondly estimating the difficulty of a given mission is non-trivial. For this study only the length of the shortest path was taken into consideration. That way both very long and very short missions can be avoided. Yet, it is possible to have missions that follow a 3 hop tendril having only a single possible path to the actual target page. When only a limited set of missions is required these unrepresentative missions can be removed manually, as done for this study. Further possibilities for creating better missions are outlined in chapter 7 in section 7.3.1.

These issues, whilst important for fully accurate research on human behaviour on the Internet, are not that vital in this thesis as the aim is to introduce and explain newly extracted features using the study as example. For any further research using this tracker the issues raised in this section have to be addressed.

6.4 Discussion and Conclusion

The overarching goal throughout my research was to obtain a better and more detailed understanding of human navigation on the Internet and overcome the limitations restricting previous research. Analysis of data provided by *Wikipedia* or *The Wiki Game* can only reveal so much when the recorded information only reveals which page has been visited at what point throughout the game mission. Thus, I raised 3 research questions in section 1.1 and tried to answer them over the course of this thesis.

The initial research question (**RQ 1**) asked how new features can be obtained in the first place. Introducing ideas from classic remote user testing and combining them research on *Wikipedia* networks allows for the introduction of a series of new features to analyse human input during the navigation process. This new data extends the traditional click traces used in related publications and thus, shifts the focus from analysing how humans navigate through a network to how single pages are being tackled. Naturally, the traditional systems are not replaced but mouse movements and information about the user's viewport give scientists new ways to explore human navigation and behaviour on the internet. Newly introduced features, most notable the page coverage, can be used to provide possible explanations for lucrative and non-lucrative clicks.

Once new features were decided the question rose as how to track them. This required the development of the tracking software system described in chapter 2 of this thesis. With the inclusion of modern technologies and browser features I realised a client-server-architecture. The tracking is done purely within the user's web browser and any domain knowledge is confined to the server. Hence, each module has a specific and clearly separated set of tasks. Recording JavaScript events provides the fastest and most accurate way to track human input without relying on external software. A WebSocket then sends the information to the server. This two-way communications channel allows the transmission of data at high rates and thus allowing a very precise tracking and answers the question as to what a tracking system capable of recording direct human input could look like (**RQ 2**).

To answer the final question (**RQ 3**) as to whether these newly introduced features can provide useful information I performed a user study in chapter 5 using the *Wikipedia for Schools 2013* dataset which is explained in chapter 4. The study reinforces the statement that whilst navigating users use these hub-highways to reach their destination. During that time pages are only skimmed and in almost all cases links are followed only when they are on the screen initially after loading. Users that read more of the content on their current page have a higher chance to perform a lucrative click stating that more reading actually pays off.

A general user will click on links at the very top of any given page when trying to move on. Thus, the positioning of links is vital to ensure a good navigability of a website network. The same applies to advertisers because any advert is wasted when its position on the website was never visited by the user. Recommender system can be used to not only support the navigation within a network but can as well help the user to navigate on individual pages by showing possibly interesting content right at the top.

These findings and the new tracking system build a foundation for further research on the topic. One practical use being the improvements on the layout of information networks like *Wikipedia*. The old and new features merge together to help finding a more complete model on human navigation.

7 Future Work

This work introduces new, by remote user testing inspired, features for the research on human navigation on the web. The tracking system capable of recording them builds the foundation for future projects in this field aimed to further improve the understanding of people browsing the web. Because of the simple portability and extendibility various studies are planned using this new software to collect data. This final chapter provides an overview over these planned studies and outlines the planned extensions to the tracker.

7.1 EEG Study

Mouse movements record the human interaction with an input device. The next step going even further into analysing the user itself is to actually monitor the body. Certain bio-signals have already been analysed in past research. (Chen, Anderson, and Sohn, 2001). The laboratory of Brain-Computer Interfaces[1] at the Graz University of Technology aims to record activity inside the human brain. The data collected this way will be combined with the data recorded by the tracking system. Understanding which regions of the brain are active during the individual phases of the navigation process might shed light on the big question of what makes a user click a specific link.

[1] https://bci.tugraz.at/

7.2 Recommender Systems

The user study (section 5) omitted the trackers ability to show various links on any given page using an overlay (Hint-overlay seen in 2.4). One of the intended purposes for this module was to display recommendations. A recommender system can be attached to the server module and calculate potentially useful links. This recommender has access to the recorded features allowing for a new kind of recommender system. Suggested links can base on the user's mouse movement or current viewport frame. Adding such a module to the tracker might not only bring research further to the goal of understanding human navigation but can actively improve it.

Daniel Lamprecht (TU Graz) performs research on recommendation networks and aims use a modified version of the tracking system to evaluate the acceptance of results presented by existing recommender systems on *IMDB*[2].

7.3 Tracker Extensions

Since the tracking software is built in a modular fashion extensions are not only quick to be implemented but vital to adapt to new research and studies.

7.3.1 Alternative Mission Models

The missions users were asked to complete in chapter 5 faced two issues. First, the difficulty of missions was not equal through a set and secondly giving the user a defined target page makes the study suffer from gamification. Hence, a new approach to generating missions is required in order to solve these problems. One possible solution is to ask the user to search for a piece of information instead of a specific site, as done so in other studies. (Ageev et al., 2011) Still, this piece of information can be difficult to find

[2]http://www.imdb.com/

for some, but easy for others based on the background knowledge of the user. Thus, would be required to estimate how well the public knows the target information. An approach to answering this question is to analyse the in degree of the pages the information can be found on. When an article is referenced by many other articles it is not only easier to find based on the number of paths leading to the target but can additionally be expected to be more known in general. Hence, the more non-intersecting paths to all pages containing the information exist, the easier the mission can be expected to be.

7.3.2 Control Server and Client

In its current form the tracker does not allow for a change of parameters during recording sessions. To allow for study-methods such as *A/B testing* this ability has to be implemented. The Control Server is a secondary server running inside the tracking systems python server and can hence actively change settings within the server. The Control Client is a separate piece of software and can be implemented as a website for easy usability. How these modules function in detail or communicate is not specified yet.

7.4 Mobile Device Tracking

With mobile devices such as smartphones and tablets becoming increasingly popular the client module of the tracker has to be augmented. Since the server is oblivious of the specific type of JavaScript event the required changes are purely in the client. For these mobile devices, focussing heavily on touch input, the *mouse* events are replaced by *Pointer Events*. These new types of events provide a unified interface for user input. Extending the tracker with the ability to capture these events allows not only for tracking on mobile devices but even introduces new possible features such as the tap pressure or the tilt of the whole device. (W3C, 2015b) Thus, future studies can be performed on mobile devices as well using this very tracking system.

Acknowledgements

First, I want to thank Denis Helic for sparking my interest in this field of research and his continuous support and supervision throughout this master's project and thesis. Then I want to thank Florian Geigl, Daniel Lamprecht and Simon Walk for welcoming me to their group, providing a fruitful research environment, helping me whenever I needed and introducing me various Python packages such as Pandas and graph-tool. Finally I wholeheartedly thank Lorenz Jäger and Bernhard Kerbl who are close friends for almost 15 years firstly for their friendship and secondly for introducing me to this now completed curriculum here at the Graz University of Technology.

Bibliography

Ageev, Mikhail et al. (2011). "Find it if you can: a game for modeling different types of web search success using interaction data." In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, pp. 345–354 (cit. on pp. 64, 68).

Agichtein, Eugene, Eric Brill, and Susan Dumais (2006). "Improving web search ranking by incorporating user behavior information." In: *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 19–26 (cit. on p. 5).

Amazon (2015). *Amazon Mechanical Turk*. URL: https://www.mturk.com/ (visited on 01/27/2015) (cit. on p. 3).

Apache Software Foundation (2015). *Apache HTTP Server Project*. URL: http://httpd.apache.org/ (visited on 01/27/2015) (cit. on p. 12).

Arroyo, Ernesto, Ted Selker, and Willy Wei (2006). "Usability tool for analysis of web designs using mouse tracks." In: *CHI'06 Extended Abstracts on Human Factors in Computing Systems*. ACM, pp. 484–489 (cit. on p. 6).

Bates, Marcia J (1989). "The design of browsing and berrypicking techniques for the online search interface." In: *Online review* 13.5, pp. 407–424 (cit. on p. 5).

Bilenko, Mikhail and Ryen W White (2008). "Mining the search trails of surfing crowds: identifying relevant websites from user activity." In: *Proceedings of the 17th international conference on World Wide Web*. ACM, pp. 51–60 (cit. on p. 5).

Blackmon, Marilyn Hughes et al. (2002). "Cognitive walkthrough for the web." In: *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, pp. 463–470 (cit. on p. 31).

Brin, Sergey and Lawrence Page (1998). "The anatomy of a large-scale hypertextual Web search engine." In: *Computer networks and ISDN systems* 30.1, pp. 107–117 (cit. on p. 4).

Chen, Mon Chu, John R Anderson, and Myeong Ho Sohn (2001). "What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing." In: *CHI'01 extended abstracts on Human factors in computing systems*. ACM, pp. 281–282 (cit. on pp. 6, 28, 67).

Clemesha, Alex (2015). *The Wiki Game*. URL: http://thewikigame.com/ (visited on 01/27/2015) (cit. on pp. 4, 65).

Cohen, Reuven, Shlomo Havlin, and Daniel Ben-Avraham (2003). "Structural properties of scale free networks." In: *Handbook of graphs and networks* (cit. on p. 41).

Darnell, Ben (2015). *Tornado Web Server*. URL: http://www.tornadoweb.org (visited on 01/27/2015) (cit. on p. 23).

ECMA-International (2015). *The JSON Data Interchange Format*. URL: http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf (visited on 01/27/2015) (cit. on p. 24).

Finkelstein, Lev et al. (2001). "Placing search in context: The concept revisited." In: *Proceedings of the 10th international conference on World Wide Web*. ACM, pp. 406–414 (cit. on p. 4).

Google (2015a). *Google*. URL: http://www.google.com/ (visited on 01/27/2015) (cit. on p. 5).

Google (2015b). *Google Analytics*. URL: http://www.google.com/analytics/ (visited on 01/27/2015) (cit. on p. 9).

Granka, Laura A, Thorsten Joachims, and Geri Gay (2004). "Eye-tracking analysis of user behavior in WWW search." In: *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 478–479 (cit. on p. 6).

Hong, Jason I and James A Landay (2001). "WebQuilt: a framework for capturing and visualizing the web experience." In: *Proceedings of the 10th international conference on World Wide Web*. ACM, pp. 717–724 (cit. on p. 7).

Milgram, Stanley (1967). "The small world problem." In: *Psychology today* 2.1, pp. 60–67 (cit. on pp. 3, 41).

Rayner, Keith (1998). "Eye movements in reading and information processing: 20 years of research." In: *Psychological bulletin* 124.3, p. 372 (cit. on p. 6).

Robertson, Stephen E (1977). "Theories and models in information retrieval." In: *Journal of Documentation* 33.2, pp. 126–148 (cit. on p. 5).

Scaria, Aju Thalappillil et al. (2014). “The last click: why users give up information network navigation.” In: *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, pp. 213–222 (cit. on pp. 4, 5, 43, 60).

Singer, Philipp et al. (2013). “Computing semantic relatedness from human navigational paths on Wikipedia.” In: *Proceedings of the 22nd international conference on World Wide Web companion*. International World Wide Web Conferences Steering Committee, pp. 171–172 (cit. on p. 4).

SOS Children (2007). *Wikipedia for Schools 2007*. URL: `http://schools-wikipedia.org/` (visited on 01/27/2015) (cit. on pp. 3, 43, 44).

SOS Children (2013). *Wikipedia for Schools 2013*. URL: `http://schools-wikipedia.org/` (visited on 23/03/2014) (cit. on pp. 18, 30, 33, 34, 36, 37, 40, 41, 43, 44, 47, 66).

Von Ahn, Luis, Mihir Kedia, and Manuel Blum (2006). “Verbosity: a game for collecting common-sense facts.” In: *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, pp. 75–78 (cit. on p. 3).

von Hertzen, Niklas (2015). *html2canvas - Screenshots with JavaScript*. URL: `http://html2canvas.hertzen.com/` (visited on 01/27/2015) (cit. on p. 17).

W3C (2015a). *HTML5*. URL: `http://www.w3.org/TR/html5/` (visited on 01/27/2015) (cit. on pp. 10, 24).

W3C (2015b). *Pointer Events*. URL: `http://www.w3.org/TR/pointerevents/` (visited on 01/27/2015) (cit. on p. 69).

W3C (2015c). *Web Storage*. URL: `http://www.w3.org/TR/webstorage/` (visited on 01/27/2015) (cit. on pp. 24, 26).

West, Robert (2015). *Wikispeedia*. URL: `http://wikispeedia.net/` (visited on 01/27/2015) (cit. on pp. 3, 4, 9, 10, 29, 43).

West, Robert and Jure Leskovec (2012). “Human wayfinding in information networks.” In: *Proceedings of the 21st international conference on World Wide Web*. ACM, pp. 619–628 (cit. on pp. 4, 62).

West, Robert, Joelle Pineau, and Doina Precup (2009). “Wikispeedia: An Online Game for Inferring Semantic Distances between Concepts.” In: *IJCAI*, pp. 1598–1603 (cit. on pp. 3, 4, 45, 48, 64).

Wikimedia Foundation (2015). *Wikipedia*. URL: `http://www.wikipedia.org/` (visited on 01/27/2015) (cit. on pp. 3, 4, 18, 33, 41, 44, 65, 66).