FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

FindMeEvidence

Hochpräzises, webbasiertes Information Retrieval zur Entscheidungsunterstützung von Ärzten in der medizinischen Routine

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medizinische Informatik

eingereicht von

Mag.rer.soc.oec. Georg Petz, Bakk.rer.soc.oec.
Matrikelnummer 0001553

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung: Privatdoz. Dr. Allan Hanbury
Mitwirkung: Ass.-Prof. Mag. Dr. Matthias Samwald

Wien, 21. September 2015

Georg Petz | Allan Hanbury

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

FindMeEvidence

High precision, web-based information retrieval for decision support of physicians in their medical routine

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Medical Informatics

by

Mag.rer.soc.oec. Georg Petz, Bakk.rer.soc.oec.
Registration Number 0001553

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Privatdoz. Dr. Allan Hanbury
Assistance: Ass.-Prof. Mag. Dr. Matthias Samwald

Vienna, 21st September, 2015

Georg Petz	Allan Hanbury

Erklärung zur Verfassung der Arbeit

Mag.rer.soc.oec. Georg Petz, Bakk.rer.soc.oec.
Ortsstraße 33, 2301 Rutzendorf

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 21. September 2015

____
Georg Petz

Danksagung

Ich bedanke mich bei Herrn Ass.-Prof. Mag. Dr. Matthias Samwald und Privatdoz. Dr. Allan Hanbury für die Vergabe und Betreuung der Diplomarbeit. Sie haben mich während der Anfertigung meiner Diplomarbeit mit viel Geduld begleitet und mich mit zahlreichen Tipps, Anregungen und Motivation unterstützt.

Nicht zuletzt möchte ich mich auch bei allen Teilnehmern der Online-Evaluation bedanken. Ohne deren Engagement und Feedback wäre diese nicht möglich gewesen.

Acknowledgements

Firstly, I would like to express my gratitude to my supervisors Ass.-Prof. Mag. Dr. Matthias Samwald and Privatdoz. Dr. Allan Hanbury. They have have supported me throughout my thesis with their patience, knowledge, and motivation.

Also, I like to thank the participants in the survey. They took the time from their busy schedules. Without their participation and feedback, the evaluation of FindMeEvidence would not have been possible.

Kurzfassung

Das Open Source Projekt „FindMeEvidence“ hat sich als Ziel die Schaffung einer Suchmaschine für die medizinische Praxis gesetzt. Im Rahmen dieser Diplomarbeit wurde an einer funktionellen Erweiterung einer bereits bestehenden Version gearbeitet. Abschließend wurde die aktuelle Version von FindMeEvidence einem Usability-Test unterzogen.

Version 1.1 des Suchsystems findet sich auf http://FindMeEvidence.org/ mit aktuell 980 452 Apache Solr Dokumente, der Code ist unter https://github.com/matthias-samwald/find-me-evidence/ verfügbar. Auf der öffentlichen Docker Registry („Docker Hub“)[1] ist ein Image mit FindMeEvidence unter der Image ID `msamwald/find-me-evidence` erhältlich. Das Image beinhaltet alles um FindMeEvidence lokal laufen zu lassen. Der Index kann nun ganz nach den eigenen Bedürfnissen erzeugt werden. Zusätzlich bleibt bei einer lokalen Installation die Suchhistorie geschützt, weil sie schwerer in die Hände Dritter gelangen kann. Konkret wurde für die neue Version ein Service für die Übersetzung von deutscher und spanischer Begriffe entwickelt. Es wurde an der Informationsextraktion, dem Ranking und der Beurteilung der Vertrauenswürdigkeit von PubMed Artikeln gearbeitet. Zusätzlich werden jetzt auch Links zum Digital Object Identifier (DOI) und PubReader™ angezeigt. Außerdem werden Open Access (OA) Artikel dementsprechend in den Suchergebnissen markiert und Assessment Daten des Wikipedia Release Version Tools werden verwendet um Wikipedia Artikel mit schlechter Bewertung zu kennzeichnen.

Der online durchgeführte Usability-Test lieferte durchwegs gute Ergebnisse. Ein Standard Usability Scale (SUS)-Score von 84 von 100 attestiert eine sehr gute Usability. Ebenso ist erwähnenswert, dass 73% aller an FindMeEvidence gestellten Fragen beantwortet werden konnten. Einige Probleme konnten wir allerdings bei dem Service für die Übersetzung ausfindig machen.

[1] https://hub.docker.com/

Abstract

This diploma thesis developed and evaluated strategies for improving the retrieval of medical content through the web. The work is based on the existing FindMeEvidence 1.0 open-source search engine, and refines and extends this search engine. Among the new features of FindMeEvidence are translation support for German and Spanish during query entry, an improved algorithm for finding key assertions, Open Access signalling for PubMed, quality signalling for Wikipedia, linking to mobile-friendly articles (PMC PubReader$^{\text{TM}}$), and providing permanent links to the Digital Object Identifier (DOI). An online evaluation of FindMeEvidence was conducted from May-July 2015. With a Standard Usability Scale (SUS) rating of 84 FindMeEvidence has an above average rating. Also the majority (73%) of the self-formulated question asked by the participants during the usability test were successfully answered with FindMeEvidence. The latest release of is FindMeEvidence 1.1. It builds on the outputs of the FindMeEvidence project whose goal is to improve efficient access to medical evidence on the web by providing a free, easily customisable, light-weight solution for medical information retrieval. The source code is hosted on GitHub and Docker is used for packaging and distribution of the software. This makes it easy to to create a local installation of FindMeEvidence to fit the needs of an organisation. All others are encouraged to use http://findmeevidence.org/ whose index currently contains 980,452 documents (statistics calculated on August 3rd, 2015) from a clinically relevant subset of PubMed, a clinically relevant subset of Wikipedia, Merck Manual Professional Edition, Medscape, National Guideline Clearinghouse, BestBETs, and ATTRACT.

Contents

List of figures

List of tables

List of Listings

CHAPTER 1

Introduction

1.1 Aim of the work

The aim of this diploma thesis is to develop and evaluate strategies for improving the retrieval of medical content through the web. The work is based on the existing FindMeEvidence 1.0 open-source search engine, and refines and extends this search engine.

1.2 Motivation

The World Wide Web has become an important source of information for medical practitioners where web search engines could help answer questions medical doctors pursue in daily medical routine [You10, PBWB06, Mas08]. Unfortunately specialised medical search engines are often locked away behind a paywall and cannot be accessed by many medical professionals. A sizeable fraction of medical practitioners reported an unwillingness to pay for clinically oriented search engines and would prefer freely available, advertisement-supported medical web sites over web sites behind a paywall [KGS+13]. This could be a reason why the self-reported use of commercial point-of-care databases (e.g., UpToDate®) was shown to be relatively limited, even though these search engines provide reliable, evidence-based clinical information. Furthermore, the confidentiality of the search history and the integrity of the search results is often not ensured.

1.3 The FindMeEvidence project

A freely available, well-designed search engine for medical practitioners could complement currently available medical information sources on the web, and could in turn have a positive impact on the quality of health care. For this reason, an open-source, mobile-friendly search engine optimised for medical information needs - FindMeEvidence (http://findmeevidence.org/) - was implemented. The goal of the FindMeEvidence project is to improve efficient access to medical evidence on the web by providing a free, easily customisable, light-weight solution for medical information retrieval. A local installation of FindMeEvidence makes it easier to deploy security measures to protect the confidentiality, integrity and availability (CIA triad) of the sensitive data that occurs during search (e.g., search history). Results from the http://khresmoi.eu/ research project are used as a basis for this project.

1.4 Methodological approach

The methodological approach consists of the following tasks:

Task 1 Reviewing the state-of-the-art and current state of FindMeEvidence 1.0.

Task 2 Developing techniques for assisting professionals that are not native English speakers in formulating queries for English medical information sources

- E.g.: In the autocomplete list generated when user enters a query, English suggestions are made when the user enters a keyword in German (note: all text corpora in the search system are English)

Task 3 Refining strategies for rapidly seeing key assertions (i.e., approximating the precision of a Question Answering (QA) system without building a QA system that automatically answers questions posed by humans in a natural language)

- Improving algorithms for identifying and showing key assertions (e.g., statements in the conclusion section)

Task 4 Developing strategies for rapidly judging the level of evidence for the included data sources

- Mechanism for judging evidence of studies in PubMed (e.g., based on the Evidence Pyramid, number of cases, journal impact factor...)
- Mechanism for detecting problematic content in Wikipedia (e.g., flagging for signs of unvalidated edits / possible vandalism; judging citations listed as evidence in Wikipedia)
- Augmenting search results with credibility information

Task 5 Formative evaluation of the system

- Preliminary evaluation with physicians in training in Austria and/or an English-speaking region
- Setting up a simple evaluation system (user is identified, tries to find results for a query while his/her behaviour is logged in the backend, finally fills out a very short form describing how happy he/she is with the system and the results he/she got)

- Conducting evaluation through the web

Task 6 Establishing the system as an open-source project

- Ensuring that the code is in a shape that allows others to contribute to the codebase and adapt it to local needs (sufficient documentation, code quality, sorting out licensing issues with third-party components)

1.5 Structure of the work

Chapter 2 (State of the art / analysis of existing approaches) gives an overview of existing research regarding retrieval of medical content through the web and examines existing search solutions (including FindMeEvidence). An introduction to the online evaluation platform used for the usability testing of FindMeEvindence is given in Chapter 3 (Methodology). Illustrative details on the implementation of FindMeEvidence 1.1 are given in Chapter 4 (Implementation). Additionally non-trivial implementation issues are explained in more detail. The results of the online evaluation are discussed in Chapter 5 (Evaluation of the system). Chapter 6 (Establishing the system as an open-source project) shows our efforts in establishing FindMeEvidence as an open-source project. Thoughts, opinions and personal observations are portrayed in Chapter 7 (Critical reflection). Finally, Chapter 8 (Summary and future work) points out the next logical step that should be taken and provides a short and concise summary of the master thesis.

CHAPTER 2

State of the art / analysis of existing approaches

Section 2.1 reviews medical search literature and Section 2.2 gives an overview of existing research regarding retrieval of medical content through the web and examines existing search solutions.

2.1 Literature studies

It has been shown that web-based search engines such as Google can aid the diagnostic process [TN06]. A study by Westbrook et al. demonstrated that the availability of an online information retrieval system increased the percentage of correctly answered medical questions from 21% to 50% in a group of 75 clinicians [WCG05]. In a recent study conducted among 500 European medical professionals, they reported frequently using general-purpose search engines (e.g., Google), medical research databases (e.g., PubMed), and - perhaps surprisingly - Wikipedia to answer medical questions online [KGS+13]. It should be noted that the first formally peer-reviewed, and edited, Wikipedia article from WikiProject Medicine & Pharmacology was published in *Open Medicine - Vol 8, No 4 (2014)* [Mas14] with the clinical topic Dengue Fever [HdWBB14]. A potential problem with these search engines is that most of them either return large amounts of clinically irrelevant or untrustworthy content (Google), or that they are mainly focused on primary scientific literature that makes selection of clinically relevant publications very time-consuming (PubMed).

2.2 Comparison and summary of existing approaches

FindMeEvidence is the only currently available system or medical search that is open-source, built on an industry-strength information retrieval engine (Apache Solr), and optimised for cross-platform and mobile web access. The index contains 824,474 documents (statistics calculated on October 31st, 2013) from a clinically relevant subset of PubMed, a clinically relevant subset of Wikipedia, Merck Manual Professional Edition, Medscape, National Guideline Clearinghouse, BestBETs, and ATTRACT. The web front-end can be used on a very wide variety of browsers and devices. The entire source code can be found here: https://code.google.com/p/bricoleur-fast-medical-search/source/checkout. Samwald and Hanbury [SH14] conducted a preliminary comparative evaluation of FindMeEvidence. A list of medical queries (N=36) was submitted to FindMeEvidence as well as to the TRIP database. FindMeEvidence results met success criteria for 25 (69,4%) of the queries, while TRIP Database results met criteria for 17 (47,2%) of the queries.

2.2.1 Techniques for assisting professionals that are not native English speakers

It seems that the ability of physicians in Europe countries to phrase questions with the proper English terms is inferior to their ability of understanding English medical text [KGS+13]. The conclusion of the study of Meats *et al.* [MBHG07] is that better training or better search interfaces are required to assist users and enable more effective searching. FindMeEvidence 1.0 uses the PubMed autocompletion web service for the autocompletion of queries but there is no translation support during query entry. The Unified Medical Language System (UMLS) metathesaurus could be used for the translation of queries and as a starting point for autocompletion. Griffon *et al.* [GCR+12] evaluated the performance of such an UMLS synonym expansion to query PubMed. Unfortunately only a small increase in recall compared to the PubMed Automatic Term Mapping (ATM)[1] was observed. Eichmann *et al.* [ERS98] describe the use of the UMLS metathesaurus for French and Spanish queries on the OHSUMED text collection, a subset of MEDLINE. Their results indicate that for Spanish the UMLS metathesaurus based method appears equivalent to multilingual dictionary based approaches and less favourable results for French.

2.2.2 Strategies for rapidly seeing key assertions

FindMeEvidence1.0 extracts the first paragraph from Wikipedia and the conclusion paragraph in the PubMed abstract to show them as a preview in the Search Engine Results Page (SERP) (Figure 2.1). Experts in [POH13] indicate the title, abstract, introduction (first paragraph), keywords and conclusion as the key areas of articles. These sections are used to evaluate the relevance of the information.

[1] map end-user queries to the Medical Subject Headings (MeSH) thesaurus and other search field descriptors

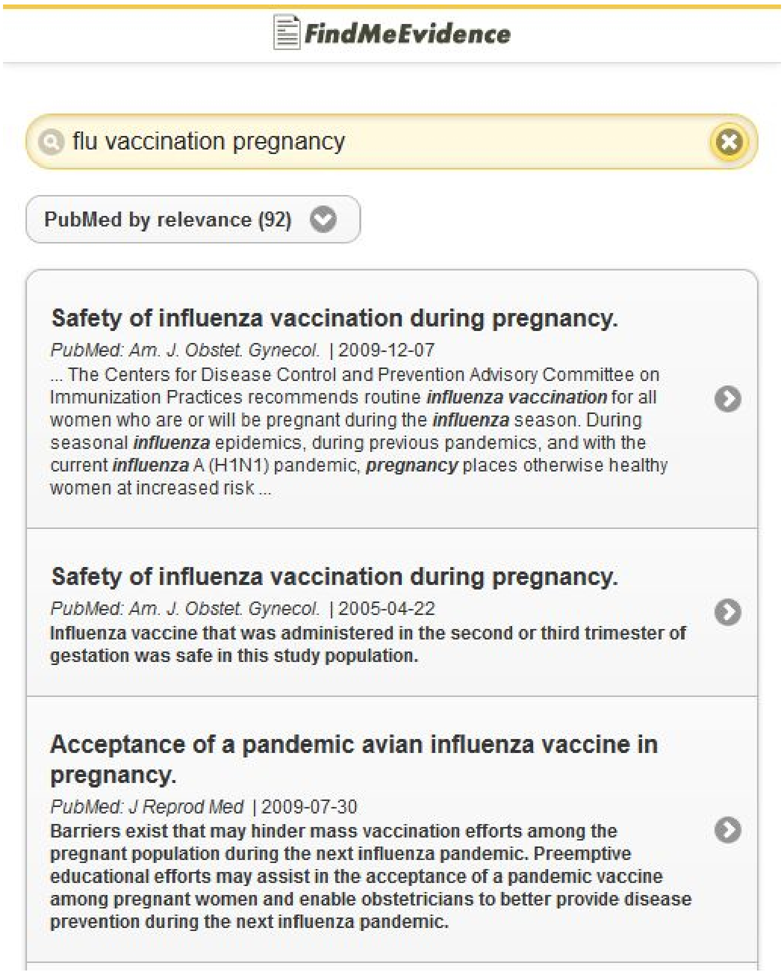


Figure 2.1: Screenshot of a FindMeEvidence 1.0 SERP containing clinically relevant research results. In the second and third result, key findings were identified in the abstracts and are shown in the result previews.

2.2.3 Strategies for rapidly judging the level of evidence

No strategies for rapidly judging level of evidence in PubMed and WikiProject Medicine & Pharmacology have been developed for FindMeEvidence 1.0. Research [DNHK09] shows that under certain conditions the level-of-evidence rating can be obtained from reading an article's abstract alone.

Judging level of evidence in PubMed

To assess the credibility of web information Pattanaphancha *et al.* [POH13] suggest 10 elements (namely, the author's name, the author's affiliation, the author's position, the publication medium, the title or abstract, the publication date of the content, the last modification date of the content, information of the editorial process, a list of references, and the number of times that information has been cited). They can be used as a starting point for judging level of evidence in PubMed.

Trustworthiness or accuracy of web resources

Google's Knowledge-Based Trust (KBT) [DGM$^+$15] scores the trustworthiness or accuracy of web resources by evaluating the correctness of their factual information. Based on the Knowledge Vault project, knowledge triples are extracted from web sources. KBT distinguishes errors made in the extraction process and factual errors. A high KBT is only given if the extracted triples are correct and correctly extracted. The algorithms are implemented in FlumeJava that builds on the concepts and abstractions for data-parallel programming introduced by MapReduce [DG08], a programming Model for processing and generating large datasets. Unfortunately Google's MapReduce code used for the extraction of knowledge triples is not available to the public for its proprietary use. Gaudinat *et al.* [GCBC11] present an operational trust model for health web pages based on the The Health On the Net (HON)code activity and results. First of all, a database of training data is created by a manual certification of health web pages based on the 8 principles of the The Health On the Net (HON)code. After sufficient training data is available a supervised learning algorithm analyses the training data and produces an inferred function, which can be used for the automatic detection of principles.

Featured articles in Wikipedia

Featured articles are considered to be the best articles Wikipedia has to offer, as determined by Wikipedia's editors but less than 0.1% of the English Wikipedia articles are labeled as featured [AS12]. At the time of this writing in WikiProject Medicine & Pharmacology only 57 of 31,654 (WikiProject Medicine) and 9 of 9,207 (WikiProject Pharmacology) are featured articles. To automatically assess the information quality of the remaining 99.9% the relevant literature mentions a variety of approaches. Vuong *et al.* [VLS$^+$08] propose three models, namely the Basic model and two Controversy Rank (CR) models to find controversial articles. Adler and de Alfaro [AdA07] present a content-driven reputation system for Wikipedia authors where the credibility of sentences is calculated by analyzing edit histories. Another way to use the edit history is followed by Zeng *et al.* [ZAD+06]. They use a dynamic Bayesian network trust model that utilized rich revision information in Wikipedia. McGuinness *et al.* [MZS+06] estimate Wikipedia trustworthiness by using an additional tab (Trust tab). The system of Adler *et al.* [ACd+08] to assign trust to Wikipedia articles uses the revision history of each article, as well as information about the reputation of the contributing authors. Four different quality models are proposed by Hu *et al.* [HLS+07] : Naive, Basic, PeerReview, and ProbReview. As also proposed in [Blu08] the Naive model takes the length as a simple indicator that gives insight into the amount of information contained in the article. The Basic model measures the quality of an article by the aggregation of authorities from all of its authors and the PeerReview model takes also the review behaviour into account. Finally, the ProbReview model introduces the probability that a user submitting a revision has reviewed a word in a document. Filatova [Fil09] shows that information overlap in multilingual Wikipedia can be used for placing information facts into a pyramid structure that can be used for information trustworthiness verification.

CHAPTER 3

Methodology

Section 3.1 recites the languages and frameworks used for software development. An introduction to the online evaluation platform used for the usability testing of FindMeEvindence is given in Section 3.2 .

3.1 Languages and frameworks

FindMeEvidence was built around the open-source Apache Solr 4 information retrieval system. Server-side scripts are written in PHP 5. They use XML Path Language (XPath) and Structured Query Language (SQL) for data preparation. The front-end is developed with jQuery Mobile. Our Integrated Development Environment (IDE) of choice is Netbeans. The software development parts of Items **Task 2** to **Task 4** of this master thesis can build on the already available server-side scripts and web front-end in PHP 5 from FindMeEvidence 1.0.

3.2 Online evaluation platform

An online evaluation platform was set up to evaluate FindMeEvidence regarding its usability. For the evaluation the usability testing technique was used.

Usability testing observes representative end users using the product to perform realistic tasks and collects empirical data. Its origin lies in the classical approach for conducting a controlled experiment where a specific hypothesis is formulated and then tested by isolating and manipulating variables under controlled conditions. Furthermore control groups must be employed to validate results and a large enough sample size is required to measure differences. At the end the hypothesis is either confirmed or rejected. This approach usually is not feasible with usability testing. To make informed decisions about design to improve products no hypotheses formulation and testing is necessary! [Rub08]

3.2.1 Basic elements of usability testing

Rubin [Rub08] defines 7 basic elements of usability testing:

1. Development of research questions or test objectives rather than hypotheses.
2. Use of a representative sample of end users which may or may not be randomly chosen.
3. Representation of the actual work environment.
4. Observation of end users who either use or review a representation of the product.
5. Controlled and sometimes extensive interviewing and probing of the participants by the test moderator.
6. Collection of quantitative and qualitative performance and preference measures.
7. Recommendation of improvements to the design of the product.

3.2.2 Usability testing procedure

Google Forms was used to create a questionnaire and collect the answers. The following steps were carried out before and during each round:

Step1: Orientation

An orientation e-mail was sent to each participant. The e-mail gave information about the study and the evaluation procedure. The full orientation e-mail (in German) can be found in the Appendix on p. 81.

Step 2: Demographics and search preferences

Users were asked to fill out questions on demographic characteristics (Table A1) and search preferences (Table A2).

Step 3: Search tasks

All participants were asked to use FindMeEvidence to gather information to answer 3 different medical question (Tables A3 to A5).

Step 4: Translation button usage

Immediately after finishing all search tasks the users were asked two questions about the translation button usage (Table A6).

Step 5: SUS and final feedback

All participant were asked to fill out the SUS questionaire (Table A7) and were given the chance to provide feedback as well as make suggestions for improvement (Table A8).

3.2.3 Piwik log file analysis

To observe end users who either use or review FindMeEvidence (Section 3.2.1: Basic element Item 4 of usability testing) the log files of the web server have to be analysed. Apache HTTP Server log files can be easily analyzed with Piwik. Bitnami[1] even offers a virtual appliance of Piwik to run in the cloud or locally on VirtualBox. Log Analytics (an alternative to the Javascript tracking method of Piwik) parses the log files of the HTTP Server and imports the data into Piwik. So-called Visitor Profiles summarise and list all visitors visits. Figure 3.1 shows the visited pages in chronological order of a http://findmeevidence.org/ visitor. The following points interpret each step in detail:

1. A new visitor enters `http://findmeevidence.org/`.
2. FindMeEvidence Search: ‘Metamphetamine adverse drug reactions’ (attempt #1)
3. FindMeEvidence Search: ‘Metamphetamine adverse drug events’ (attempt #2)
4. FindMeEvidence Search: ‘Metamphetamine adr’ (attempt #3)
5. FindMeEvidence Search: ‘Methamphetamine adr’ (attempt #4)
6. FindMeEvidence Search: ‘Metamfetamin’ (attempt #5)
7. FindMeEvidence PubMed Result Preview: ‘The clinical toxicology of metamfetamine.’
8. FindMeEvidence Search: ‘Methamphetamine’ (attempt #6)
9. Visitor clicks on Wikipedia article twice: 2x HTTP 302 redirect to `https://en.wikipedia.org/wiki/Methamphetamine/` (for explanation of redirects see Section 4.4.4)
11. FindMeEvidence Search: ‘Methamphetamin’
12. FindMeEvidence Search: ‘Metamphetamin’
13. FindMeEvidence Search: ‘Metamphetamine’
14. FindMeEvidence Search: ‘methamphetamine adverse reactions’
15. FindMeEvidence Search: ‘metamphetamine’
16. FindMeEvidence Search: ‘metamfetamine’

[1] https://bitnami.com/stack/piwik

If the user determines `https://en.wikipedia.org/wiki/Methamphetamine` as the best search result, the number of attempts is simply the number of all searches done before. In this case 6.

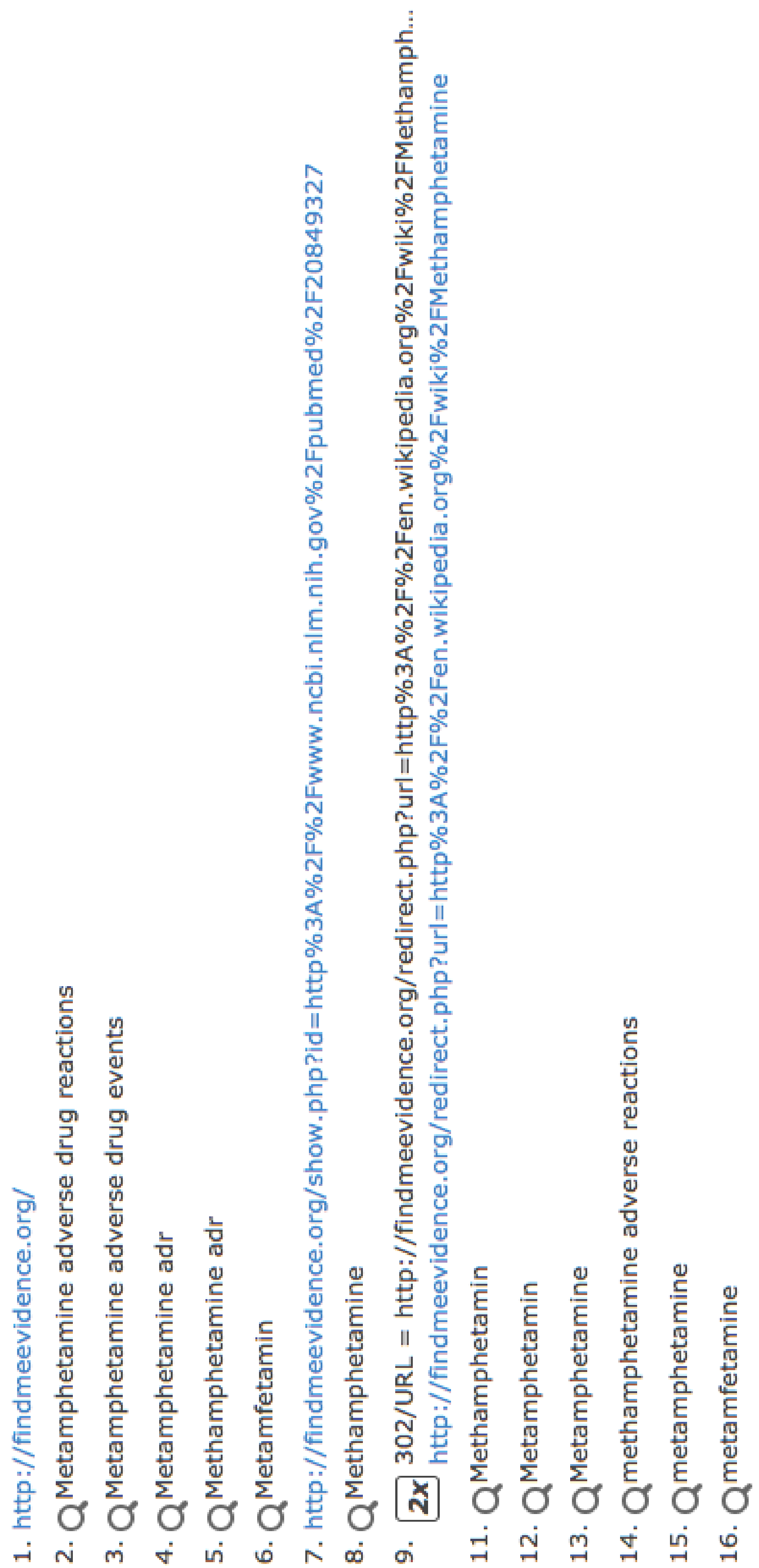


Figure 3.1: Piwik log file analysis

CHAPTER 4

Implementation

This Chapter gives illustrative details on the implementation of FindMeEvidence 1.1. How the techniques for assisting professionals that are not native English speakers are implemented is described in Section 4.1. Algorithms for identifying and showing key assertions are specified in Section 4.2. Section 4.3 defines the strategies for rapidly judging level of evidence. FindMeEvidence enhancements and bug fixes are listed in Section 4.4. An overview of the system architecture is given in Section 4.5.

4.1 Query translation support

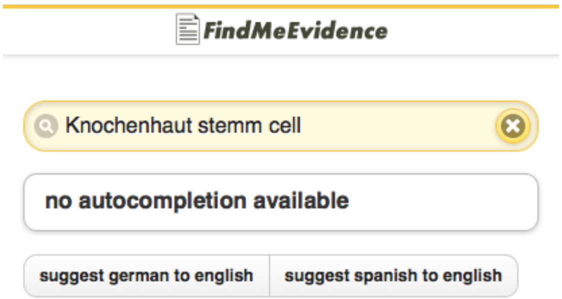
For the new version of FindMeEvidence techniques for assisting professionals that are not native English speakers in formulating queries for English medical information sources were developed (**Task 2**). Version 1.1 of FindMeEvidence comes with translation support for German and Spanish during query entry.

4.1.1 Translation support example: 'perisoteum stem cell'

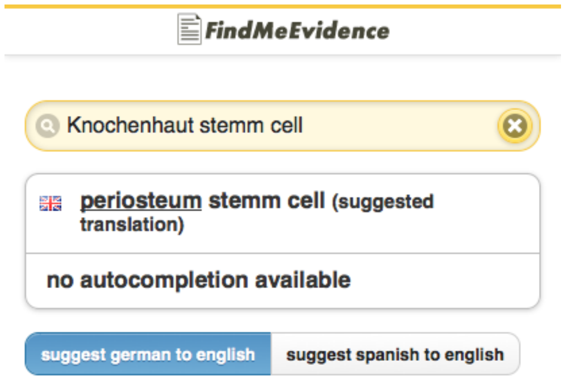
E.g., the PubMed autocompletion web service alone returns no suggestions for 'Knochenhaut[1] stemm cell' (Figure 4.1a). If the option 'suggest german to english' is selected and the cursor is placed at the end or inside of 'Knochenhaut' the suggested translation 'periosteum' is displayed as the suggested translation (Figure 4.1b). Clicking on the suggested translation replaces the german term in the query with 'periosteum', sends 'perisoteum stemm cell' to the PubMed autocompletion web service and displays the results (Figure 4.1c). Even with a spelling error: 'perisoteum stem**m** cell' the PubMed autocomplete suggests 'perisoteum stem cell'. FindMeEvidence can be configured to automatically detect the language of the browser and only show the corresponding translation button. If a Spanish speaking FindMeEvidence user has activated the option

[1] German for periosteum

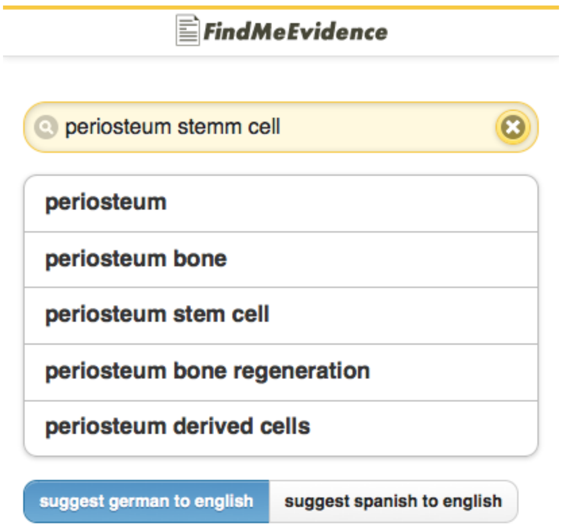
'suggest spanish to english' begins to type 'periosto[2]' the translation support kicks in and displays the suggested translation (Figure 4.2).



(a) no PubMed suggestions



(b) suggested translation



(c) PubMed suggestions

Figure 4.1: Query formulation assistance (German)

[2]Spanish for periosteum

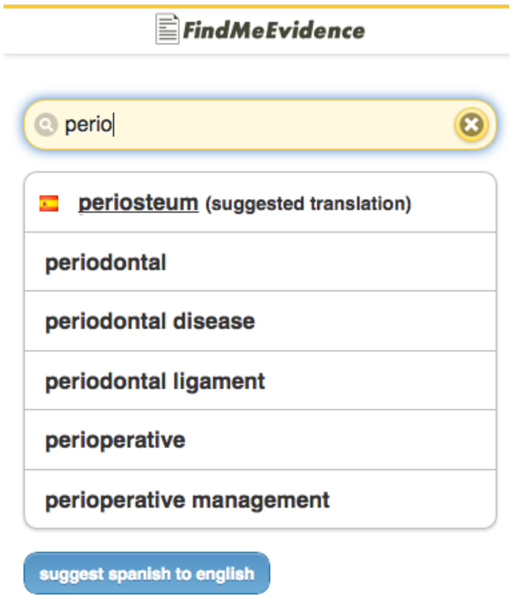


Figure 4.2: Query formulation assistance (Spanish)

4.1.2 Translation support dictionary

The FindMeEvidence translation support is based on 36,413 wikipedia article titles from WikiProject Medicine & Pharmacology and 27,455 preferred terms from the MeSH descriptors of the year 2015. The MediaWiki API langlinks property is used to get a list of all language links from the provided pages to other languages (e.g api.php?action=query&titles=Merkel%20cell%20carcinoma&prop=langlinks). The included translation in the Interwiki link of the wikipedia article is extracted and indexed. There are 7,353 Spanish langlinks and 9,498 german langlinks for the 36,413 wikipedia articles in FindMeEvidence. The remaining wikipedia article titles and the preferred MeSH terms are translated with the Yandex.Translate API.

Index-time and query-time Apache Solr analyzer

Apache Solr can be set up to have multiple configurations and search indexes (SolrCores) on the same Solr instance. FindMeEvidence uses one for the medical datasets and the other one for the translation support dictionary (Figure 4.9). Listing 1 shows the configuration of the index-time and query-time Solr analyzer used for indexing and querying the translated terms. Solr offers various field types but only `solr.TextField` has an analyzer configuration. Each analyzer has a chain that specifies an ordered sequence of processing steps that converts the original text into a sequence of terms. An analyzer chain optionally starts with a list of `CharFilterFactories`. They operate at a character level to perform manipulations. We use a `solr.PatternReplaceCharFilterFactory` (extends `CharFilterFactory`) for both analyzers to replace the '-' character with a whitespace. The next step in the chain is a tokenizer. As its name already suggests it tokenizes text. In our case the `WhitespaceTokenizerFactory` is used to tokenize by whitespace. The chain ends by a list of optional `TokenFilterFactories`. Both analyzer types use a `LowerCaseFilterFactory` to lowercase the letters in each token. Additionally at index-time the `PatternReplaceFilterFactory` removes text inside brackets and a `EdgeNGramFilterFactory` creates n-grams from the beginning edge of an input token. The n-grams are created at index-time so that at query-time the query term can be matched directly without any n-gram analysis. Defining the n-grams filter in the querying stage of analyses could lead to false positive hits! Using a `EdgeNGramFilterFactory` during index-time makes the index a bit larger, but on the other hand a wildcard search is too slow for autosuggest. Another problem with wildcard search is that there is little control over the ranking of documents returned [PT14]. Very good results regarding index size and retrieval time were obtained with with a minimum n-gram length of 4 (the `minGramSize` attribute) and a maximum length of 25 (the `maxGramSize` attribute). Additionally by utilising the norm function[3]: `norm(translationfieldname) desc` we are boosting shorter terms (Listing 2).

```
<fieldType name="text_translation" class="solr.TextField"
        positionIncrementGap="100">

 <analyzer type="index">
        <charFilter class="solr.PatternReplaceCharFilterFactory"
                pattern="-" replacement=" "/>
        <tokenizer class="solr.WhitespaceTokenizerFactory"/>
        <filter class="solr.LowerCaseFilterFactory"/>
        <filter class="solr.PatternReplaceFilterFactory"
                pattern="\(.*\)" replacement=""/>
        <filter class="solr.EdgeNGramFilterFactory"
                minGramSize="4" maxGramSize="25" side="front"/>
 </analyzer>

 <analyzer type="query">
        <charFilter class="solr.PatternReplaceCharFilterFactory"
                pattern="-" replacement=" "/>
        <tokenizer class="solr.WhitespaceTokenizerFactory"/>
        <filter class="solr.LowerCaseFilterFactory"/>
 </analyzer>

</fieldType>
```

Listing 1: Index-time and query-time Apache Solr analyzer

[3]the product of the index time boost and then length normalisation factor

```php
<?php
$request_url .= "/select?q=" . urlencode($extracted_word)
        . "&sort=norm(" . $language . ")+desc&wt=xml&df="
        . $language;
```

Listing 2: Apache Solr query to get translation

Classification of the translation service

The choices of autocomplete are by definition based on a search within controlled vocabulary for entries matching a particular query [RRT13]. Hence we can classify our service as autocomplete with the sole difference that we only offer one choice, the term we think is the correct translation, based on the mapping of a 1:1 dictionary.

4.2 Relevance of information

Task 3 deals with refining strategies for rapidly seeing key assertions by improving algorithms for identifying and showing key assertions. Prior to showing the key assertions in the SERP the positions in the search result have to be determined (ranking).

4.2.1 Ranking

Google Scholar and PubMed

Beel and Gipp [BG09a, BG09b] partly reverse engineered Google Scholar's ranking algorithm. They show that articles with a high number of citations are moved to a higher rank in the list of articles retrieved. Also a high weight is put on the author's name and the impact factor of the journal. Google Scholar puts no or low weight on the publication date. Hence older articles are found more in top positions than recent articles. This behaviour strengthens the Matthew Effect. (The Matthew Effect describes that authors with a high status are more often cited than those who are not as well known [Mer68].) A study of Nourbakhsh *et al.* [NNW$^+$12] shows that articles relevant to clinical questions retrieved by Google Scholar appear to have a higher number of citations and to come from journals with higher impacter factors than those retrieved from the PubMed database.

FindMeEvidence

To improve precision at the top ranks of results returned we would like to give more preference to documents that have a high number of reverse citations but we have to keep the Matthew Effect in mind. To counteract the Matthew Effect a positive weight to more recent documents is given at the same time. For implementation we use the so called Solr boost function parameter (bf).

Reverse citations boost

Equation 4.1 boosts a document depending on its reverse citations (Figure 4.3). As indicated in Table 4.1 the retrieval of the number of citations is only partly supported by PubMed. FindMeEvidence uses the Entrez Programming Utilities (E-utilities) **EFetch** to retrieve PubMed records. Only **ELink** (responds to a list of UIDs in a given database with either a list of related IDs in the same database or a list of linked IDs in another Entrez database) can be used to get the number of reverse citations. Meaning an HTTP request has to be sent for each PubMed article to query the number of citations. Hence no efficient way to harvest the number of reverse citations for hundreds of thousands PubMed articles exists.

$$\log_{10}(citedin + 1) \tag{4.1}$$

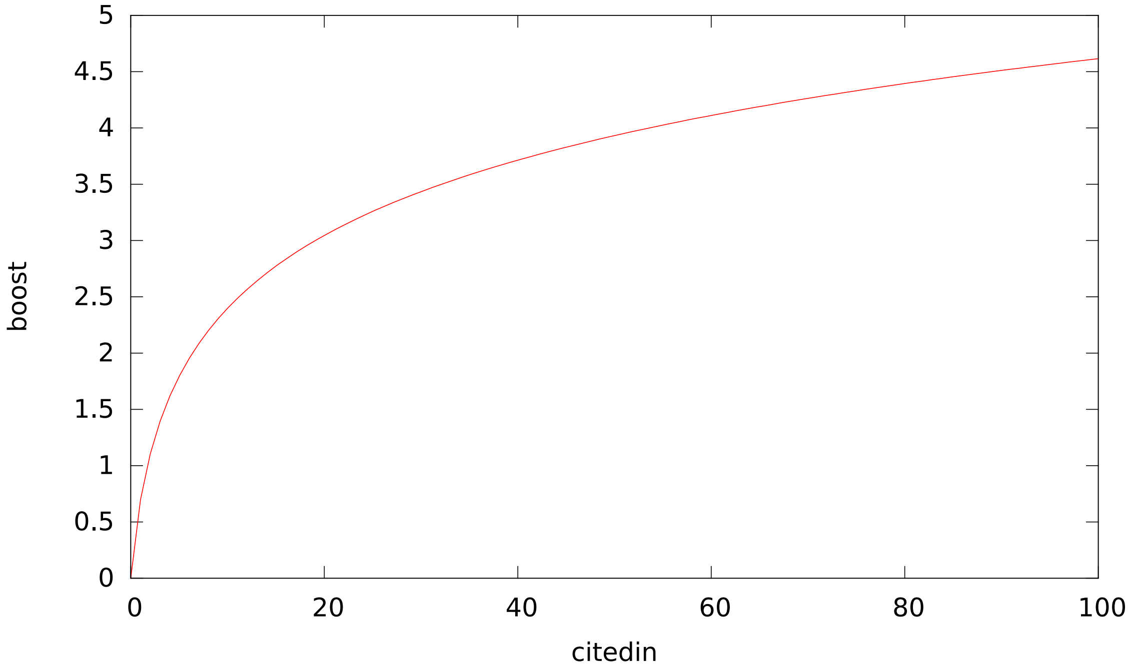


Figure 4.3: Reverse citations boost ($\log_{10}(citedin + 1)$)

Recency boost

To boost documents based on their date a reciprocal function with recip(x,m,a,b) implementing a/(m*x+b) (Equation 4.2 and Figure 5.2) is being used [Kuc13].

$$\frac{a}{m \cdot x + b} \tag{4.2}$$

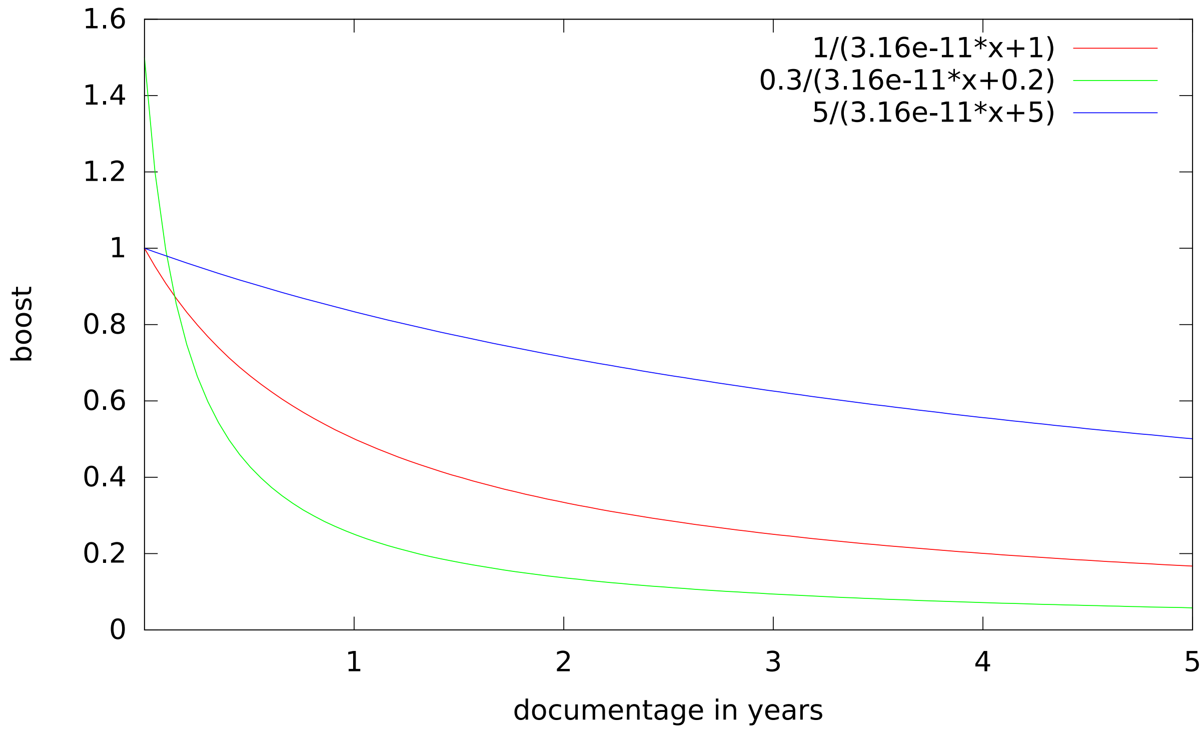


Figure 4.4: Recency boost ($\frac{a}{m \cdot x+b}$)

Setting $a = b = 1$, $x = documentage$, $m = 3.16e^{-11}ms$ and combining the two boosts we get the boost function (Equation 4.3).

$$bf = \frac{1}{3.16e^{-11} \cdot documentage + 1} + \log_{10}(citedin + 1) \tag{4.3}$$

4.2.2 Key assertions

In the next step the key assertions of the articles are shown to the user. For PubMed articles it is checked by an XPath 2.0 query if markup of the conclusion section:

```
boolean(/PubmedArticle/MedlineCitation/Article/
Abstract/AbstractText[@NlmCategory='CONCLUSIONS'])
```

exists. 511,706 of 820,066 articles from PubMed in FindMeEvidence have a markup of the conclusion section. If this is the case we already found the key assertion. If the markup approach fails we use a regular expression to find the conclusion section. If the regular expression also is not successful we could simply extract the last 4 sentences.

4.3 Quality and credibility of information

To develop strategies for rapidly judging level of evidence for included data sources is defined by **Task 4**. Mechanisms for judging evidence of studies in PubMed and for detecting problematic content in Wikipedia help the user to judge the quality and credibility of information. The last part of **Task 4** deals with augmenting search results with quality and credibility information.

4.3.1 Normative trustworthiness criteria for PubMed

Table 4.1 shows 10+1 normative trustworthiness criteria based on a study of Pattanaphancha *et al.* [POH13]. The quantitative analysis of questionnaires suggests 10 elements (as already mentioned in Section 2.2). We additionally added the impact factor of the the journal as the 11th but decided later on that it is no reliable trustworthiness criteria for FindMeEvidence. In addition the qualitative analysis of Pattanaphancha *et al.* stresses the importance of the title, abstract and conclusion in assessing the relevance of the information to the user. If some elements of supportive information are missing, users need to search for other pieces of supportive information to assess the trustworthiness. The key item used to find additional supportive information is the author's name. If the affiliation is not present most of the experts suggested searching for the affiliation of the author using the author's name [Pat14]. The author's position is not present in the PubMed record. Hence we cannot retrieve it and as there are no peer-reviewed or refereed journals in PubMed too we can also skip the editorial process criteria. The last modification date in PubMed could be assumed by the date on which the information is published. As a consequence we can summarise the two in the main criteria 'currency'. The number of citations was already discussed in Section 4.2.1. In summary, as shown in Table 4.1 it can be said that FindMeEvidence 1.1 supports the display of all important normative trustworthiness criteria.

4.3.2 Open Access (OA) - Signaling

The PMC OA Subset contains articles that are protected by copyright but are made available under one of the Creative Commons (CC) licenses from Table 4.2. FindMeEvidence signals the user OA of PMC articles via the PMC OA symbol (Figure 4.5). 116,468 of 820,066 articles from PubMed in FindMeEvidence are OA.

FindMeEvidence uses the PMC OA Web Service for harvesting the PMCIDs. For efficient lookup all the ids retrieved are inserted into a SQLite Version 3 database. This database is used during index-time of FindMeEvidence to flag OA PubMed articles. It is also possible to harvest the ids with Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH). FindMeEvidence temporarily supported OAI-PMH harvests. The support was skipped abandoned because the PMC OA Web Service has better performance and less overhead. OAI-PMH always returns the whole record! Inspired by Sompel *et al.* [SYH03] FindMeEvidence also supports on the fly OA-signaling, if

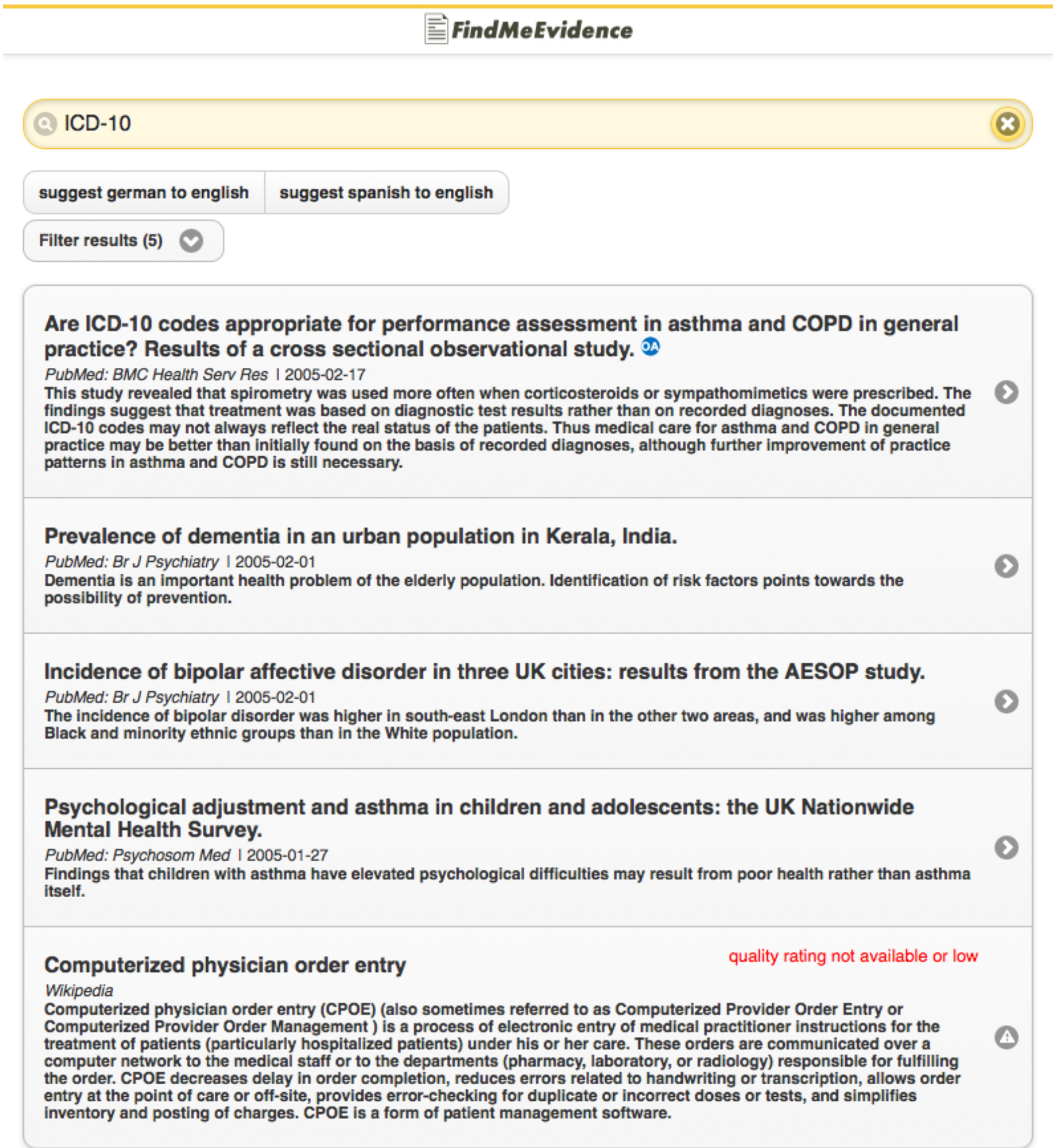


Figure 4.5: Search results with quality and credibility information (oa-signaling in the first result and quality rating in the last one)

Criterion	PubMed support	FindMeEvidence 1.1 support
Author's affiliation	✓	-
Author's name	✓	✓
Author's position	-	-
Content of the title, abstract or conclusion	✓	✓
Editorial process	-	-
Impact factor of journal (journal name)	-	-
Last modification date (currency)	✓	✓
List of references	✓	-
Number of citations (how often it has been referenced)	✓	∼
Publication date (currency)	✓	✓
Publication medium	✓	-

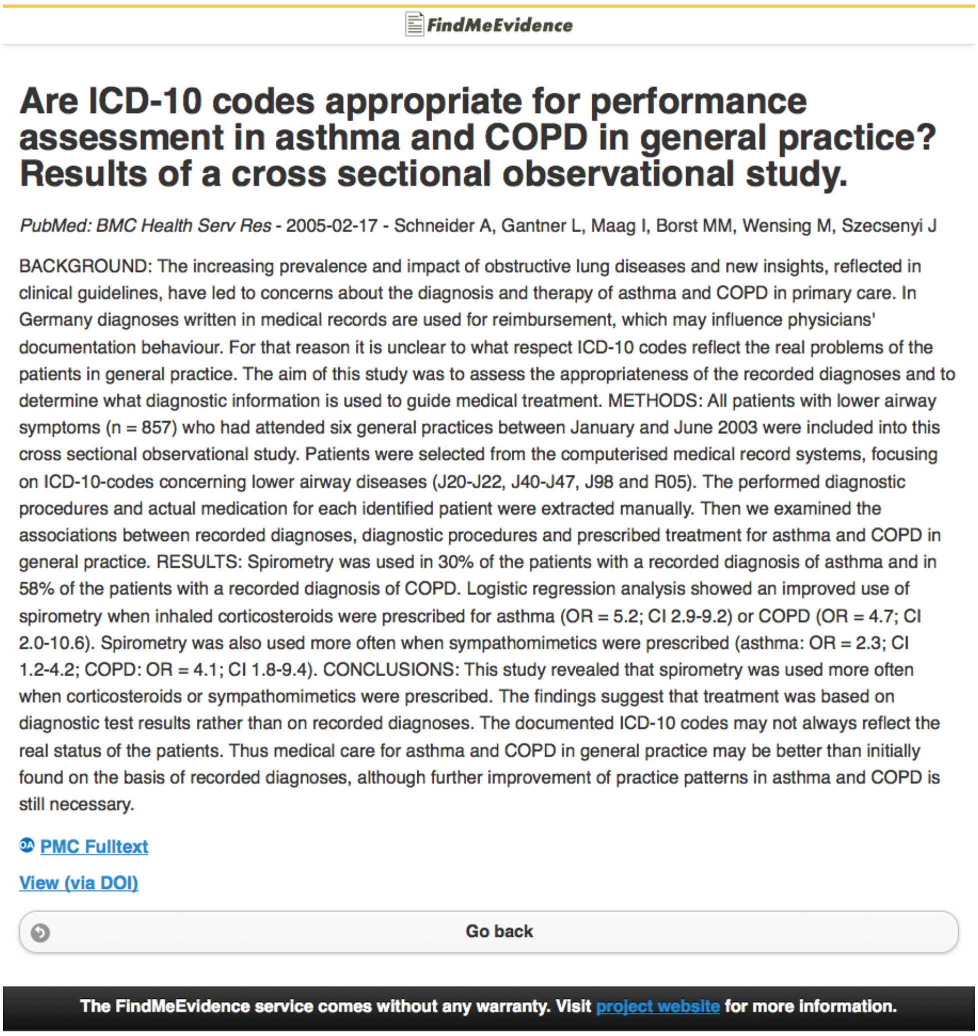
-: fully supported ✓: not supported ∼: partly supported

Table 4.1: 11 normative trustworthiness criteria for PubMed

License type
Any CC license
CC BY (Attribution)
CC BY-ND (Attribution, no derivatives)
CC BY-NC (Attribution, noncommercial)
CC BY-NC-ND (Attribution, noncommercial, no derivatives)
CC BY-NC-SA (Attribution, noncommercial, share-alike)
CC BY-SA (Attribution, share-alike)

Table 4.2: CC licenses of the PMC OA Subset

activated. Additional metadata retrieved via OAI-PMH (not only the OA information) can be displayed in the PubMed result preview (Figure 4.6).



FindMeEvidence

Are ICD-10 codes appropriate for performance assessment in asthma and COPD in general practice? Results of a cross sectional observational study.

PubMed: BMC Health Serv Res - 2005-02-17 - Schneider A, Gantner L, Maag I, Borst MM, Wensing M, Szecsenyi J

BACKGROUND: The increasing prevalence and impact of obstructive lung diseases and new insights, reflected in clinical guidelines, have led to concerns about the diagnosis and therapy of asthma and COPD in primary care. In Germany diagnoses written in medical records are used for reimbursement, which may influence physicians' documentation behaviour. For that reason it is unclear to what respect ICD-10 codes reflect the real problems of the patients in general practice. The aim of this study was to assess the appropriateness of the recorded diagnoses and to determine what diagnostic information is used to guide medical treatment. METHODS: All patients with lower airway symptoms (n = 857) who had attended six general practices between January and June 2003 were included into this cross sectional observational study. Patients were selected from the computerised medical record systems, focusing on ICD-10-codes concerning lower airway diseases (J20-J22, J40-J47, J98 and R05). The performed diagnostic procedures and actual medication for each identified patient were extracted manually. Then we examined the associations between recorded diagnoses, diagnostic procedures and prescribed treatment for asthma and COPD in general practice. RESULTS: Spirometry was used in 30% of the patients with a recorded diagnosis of asthma and in 58% of the patients with a recorded diagnosis of COPD. Logistic regression analysis showed an improved use of spirometry when inhaled corticosteroids were prescribed for asthma (OR = 5.2; CI 2.9-9.2) or COPD (OR = 4.7; CI 2.0-10.6). Spirometry was also used more often when sympathomimetics were prescribed (asthma: OR = 2.3; CI 1.2-4.2; COPD: OR = 4.1; CI 1.8-9.4). CONCLUSIONS: This study revealed that spirometry was used more often when corticosteroids or sympathomimetics were prescribed. The findings suggest that treatment was based on diagnostic test results rather than on recorded diagnoses. The documented ICD-10 codes may not always reflect the real status of the patients. Thus medical care for asthma and COPD in general practice may be better than initially found on the basis of recorded diagnoses, although further improvement of practice patterns in asthma and COPD is still necessary.

PMC Fulltext

View (via DOI)

Go back

The FindMeEvidence service comes without any warranty. Visit project website for more information.

Figure 4.6: Pubmed result preview (PubReader™ link: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC549048/?report=reader and DOI link: http://dx.doi.org/10.1186/1472-6963-5-11)

4.3.3 WikiProject Medicine & Pharmacology article assessments

FindMeEvidence utilises the Wikipedia Release Version Tools[4] to create a list of relevant articles from WikiProject Medicine & Pharmacology and search for article assessment data. Figure 4.7 shows a returned article list. Unfortunately there is no Application Programming Interface (API) available. Hence we have to parse HTML to extract the link to the article and its quality rating. For parsing we use the PHP Simple HTML DOM Parser[5].

An analysis of 36,296 WikiProject Medicine & Pharmacology articles in FindMeEvidence shows that the majority (38.5%) are Stubs (Figure 4.8). Stubs are very short articles with a very basic description of the topic. They are the lowest class of the normal classes. Only articles of quality FA | A | GA | B | C | Start are shown without a warning in FindMeEvidence. A detailed description of the the featured qualities in FindMeEvidence is given in Table 4.3.

Result	Article	Importance		Quality		Review Release		Score
50	Receptor (biochemistry) (t · h · l)	Top	2011-12-28 (t)	Start	2007-08-13 (t)			1341
51	Rectal administration (t · h · l)	Top	2010-07-07 (t)	Start	2010-07-07 (t)			526
52	Sedative (t · h · l)	Top	2008-02-21 (t)	Start	2008-02-21 (t)			1293
53	Side effect (t · h · l)	Top	2013-02-23 (t)	Start	2013-02-23 (t)			526
54	Sympathomimetic drug (t · h · l)	Top	2008-01-22 (t)	Start	2008-01-22 (t)			1171
55	Thiazide (t · h · l)	Top	2007-01-23 (t)	Start	2007-01-23 (t)			1137
56	Tocolytic (t · h · l)	Top	2008-03-06 (t)	Start	2008-03-06 (t)			1006

Figure 4.7: Wikipedia Release Version Tools article list (importance & quality)

[4] https://tools.wmflabs.org/enwp10/
[5] http://sourceforge.net/projects/simplehtmldom/

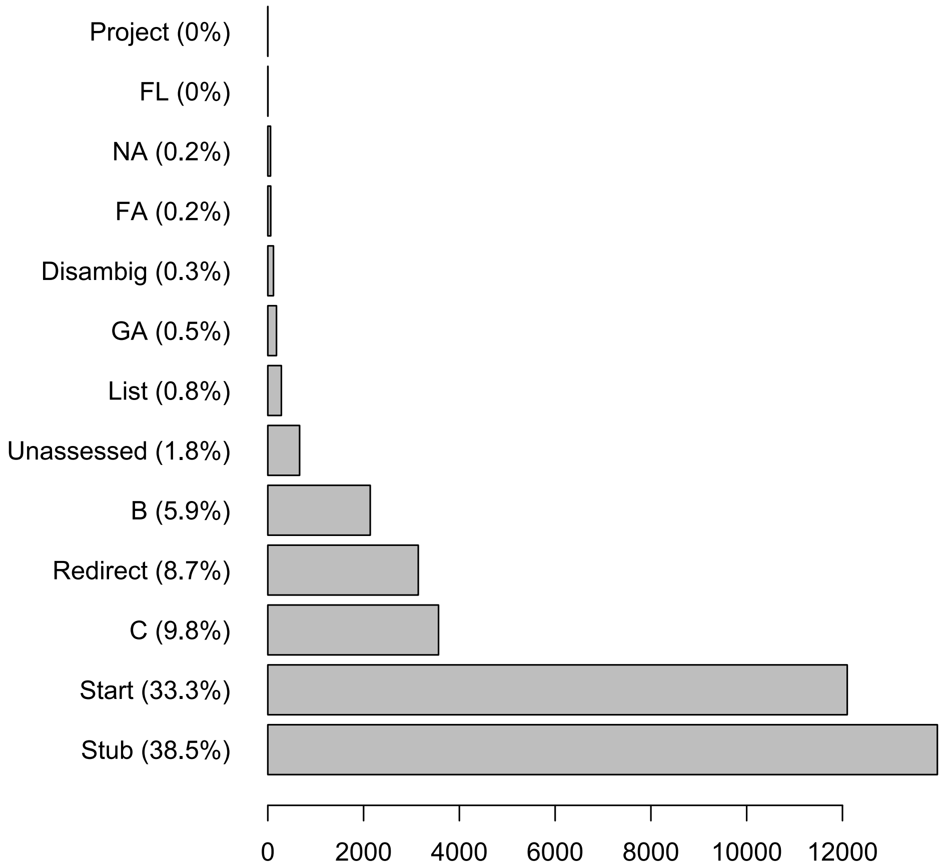


Figure 4.8: WikiProject Medicine & Pharmacology quality assessment

Class	Description
FA (Featured Article)	Featured Articles passed an official review and are considered to be the best articles Wikipedia has to offer.
A (A-Class)	The article provides a well-written, clear and complete description of the topic. (Good article status is not a requirement for A-Class and **not used by WikiProject Medicine & Pharmacology**)
GA (Good Article)	Good Articles passed an official review and meet 6 criteria (well-written, verifiable, broad in its coverage, neutral, stable and illustrated).
B (B-Class)	The article is mostly complete and meets 6 C-Class criteria: suitably referenced with inline citations, reasonably covers the topic, has a defined structure, is reasonably well-written, contains supporting materials where appropriate, and presents its content in an appropriately understandable way.
C (C-Class)	The article still has significant problems or require substantial cleanup.
Start	The article has already a usable amount of good content but is still incomplete.

Table 4.3: Featured WikiProject Medicine & Pharmacology qualities in FindMeEvidence

4.4 FindMeEvidence enhancements

4.4.1 Persistent links for PubMed records

FindMeEvidence uses the E-utilities, a set of eight server-side programs that provide a stable interface into the Entrez query and database system at the National Center for Biotechnology Information (NCBI), to retrieve PubMed records. The records contain among other things the PMCID, release date, and DOI (Listing 3). If the release date does not lie in the future a link[6] to the PubReader™, an alternative web presentation to read literature in PMC more reader-friendly on tablets and other small screen devices, is shown in the PubMed result preview (Figure 4.6). FindMeEvidence uses the DOI to provide an actionable and persistent link (Figure 4.6). For resolving the DOI the service http://dx.doi.org/ is used.

4.4.2 Client-side calls to the PubMed autocomplete service

The server-side PHP script to call the PubMed autocomplete service was replaced by a direct asynchronous JavaScript + XML (Ajax) call from the browser to the PubMed API.

[6] `<a href="http://www.ncbi.../?report=reader">PMC Fulltext</a>`

```
<?php
// Fetch PMCID
$pmc = $article->xpath
("/PubmedArticle/PubmedData/ArticleIdList/ArticleId[@IdType='pmc']");

// Fetch PMC release date
$history = $article->PubmedData->History;
$history_year = $history->xpath
("//PubMedPubDate[@PubStatus=\"pmc-release\"]/Year");
$history_month = $history->xpath
("//PubMedPubDate[@PubStatus=\"pmc-release\"]/Month");
$history_day = $history->xpath
("//PubMedPubDate[@PubStatus=\"pmc-release\"]/Day");

// Fetch DOI
$doi = $article->xpath
("/PubmedArticle/PubmedData/ArticleIdList/ArticleId[@IdType='doi']");
```

Listing 3: Extraction of PMCID, release date and DOI

4.4.3 All authors in the the PubMed result preview

The latest version of FindMeEvidence lists all authors of the article in the PubMed result preview (Figure 4.6) .

4.4.4 HTTP redirects to track user

To track clicks with Piwik on outgoing links (Section 3.2.3) PHP's header function is used to redirect to the result page (Listing 4).

```
<?php
header("Location: " . $_GET ["url"]);
exit;
```

Listing 4: redirect.php

4.4.5 Apache Solr upgrade

The index was upgraded from version 4.4.0 to 4.10.4.

4.5 Architecture

An overview of the system architecture is shown in Figure 4.9. Samwald *et al.* [SH14, SSP+14] give an overview of the original architecture. The heart of FindMeEvidence is an Apache Solr instance with 2 SolrCores. One is used for the medical datasets and the other one for the translation support dictionary. Content is added to the index from 5 different components namely Medical websites, PubMed, Wikipedia, Yandex Translate, and DBpedia.

- **Medical websites:** A script based on the PHPCrawl library[7] and the html2text PHP script[8] crawls and indexes relevant content from each page.
- **PubMed:** A clinically relevant subset of PubMed is downloaded and indexed. For the expansion of the abbreviations in the abstract 'A Simple Algorithm For Identifying Abbreviation Definitions in Biomedical Text' [SH03] is used.
- **Wikipedia:** A clinically relevant subset of Wikipedia (WikiProject Medicine & Pharmacology) is downloaded and indexed.
- **Yandex:** Wikipedia article titles with no English translation and the preferred MeSH terms are translated with the Yandex.Translate API to build a dictionary. The dictionary is used for the query translation support (Section 4.1).
- **DBpedia:** A synonym dictionary based on Wikipedia page redirects is created and used for synonym injection during query processing. Potential problems of this solution are described in Section 7.1.

The web front-end makes use of server-side PHP, client-side Javascript (including Ajax), and jQuery Mobile. Autocompletion of queries typed in by users is based on client-side calls to the PubMed autocompletion web service (Section 4.4.2).

[7] http://phpcrawl.cuab.de/
[8] http://www.howtocreate.co.uk/php/html2texthowto.html

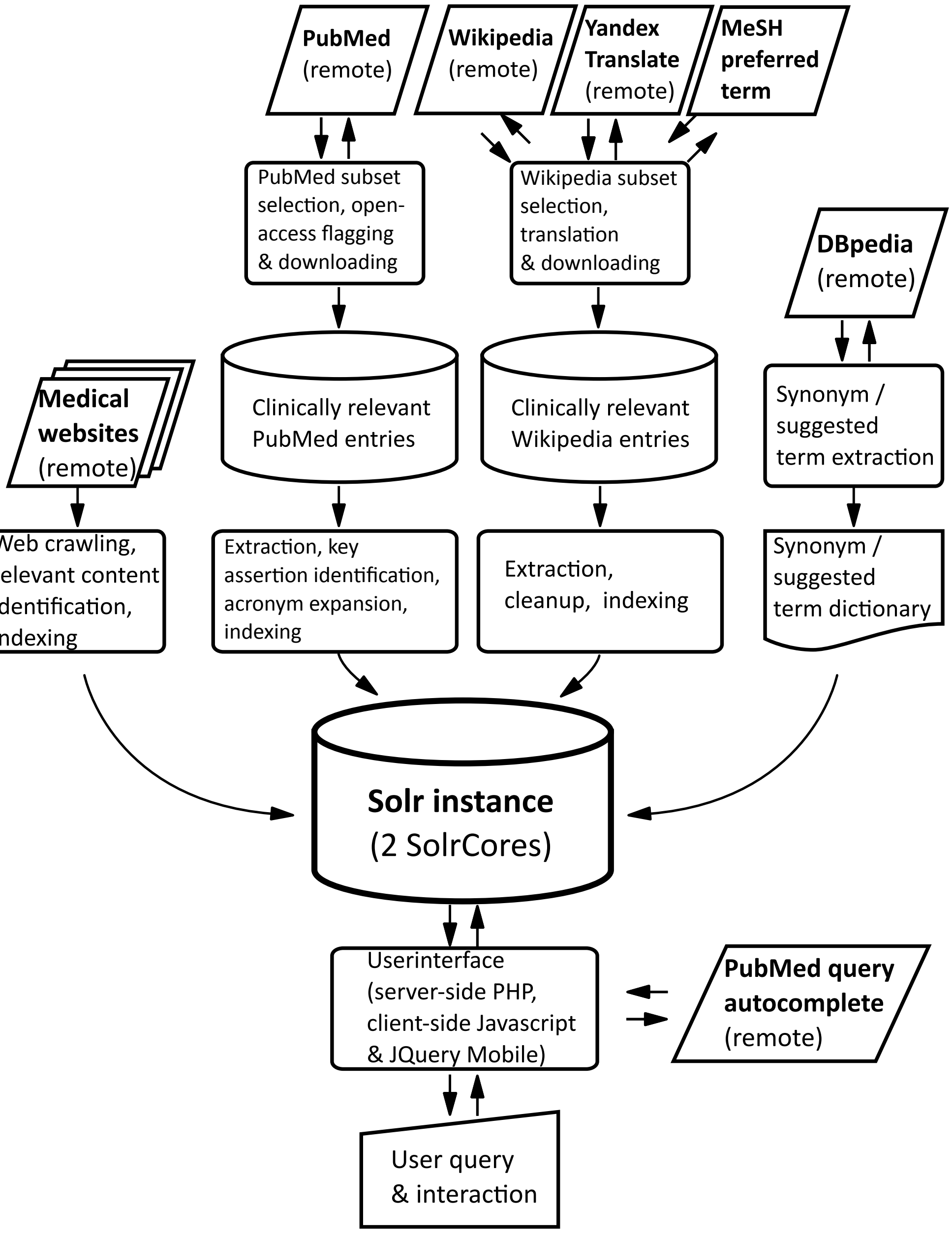


Figure 4.9: FindMeEvidence architecture

CHAPTER 5

Evaluation of the system

As defined in **Task 5** a simple evaluation system was set up to evaluate FindMeEvidence in Austria by physicians in training . The online evaluation platform used is described in detail in Section 3.2. Research questions are elaborated in Section 5.1. Section 5.2 shows the results of the online evaluation and Section 5.3 discusses the inferences we draw from the observation we made along with analysing the data.

5.1 Research questions

The following research questions describe the issues and questions that need to be resolved by the online evaluation. They provide insight into the effectiveness, efficiency, usability, usage of the translation service, and overall user satisfaction.

- **effectiveness:**
  - Did the users succeed in solving all tasks?
  - Do the search results answer the questions asked or not?
- **efficiency:**
  - How quickly the users could find the correct answer?
- **usability:**
  - Are the users satisfied with the ease of use and learnability?
  - Are there major usability flows that prevent user from completing the tasks?
- **usage of the translation service:**
  - Study the use of the translation service.

- **overall user satisfaction:**
  - What do users like or do not like?
  - What requires change?

5.2 Evaluation results

The online evaluation of FindMeEvidence was conducted from May-July 2015.

5.2.1 Demographic questionnaire analysis

5 participants took part in the online usability testing. Rubin [Rub08] claims that research indicates that testing 4 to 5 participants will expose the vast majority of usability problems. Overall, 40% were female and 60% male (Figure 5.1). As illustrated in Figure 5.2 3 age groups were represented. 40% between 18-25 years, 40% between 26-32 years and 20% between 33-40 years. 80% of the participants were physicians in training and 20% were working in R&D at a medical university (Figure 5.3) with 0-5 years of work experience (Figure 5.4).

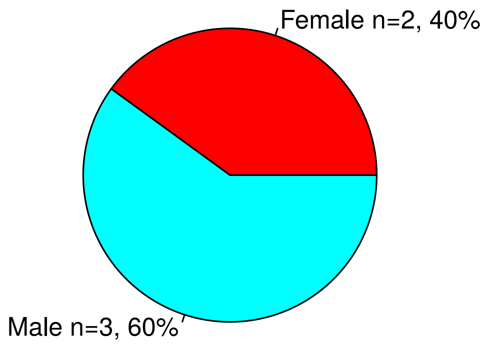


Figure 5.1: Gender distribution

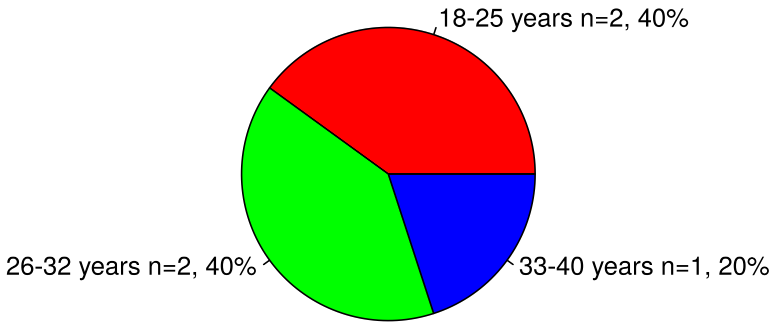


Figure 5.2: Age distribution

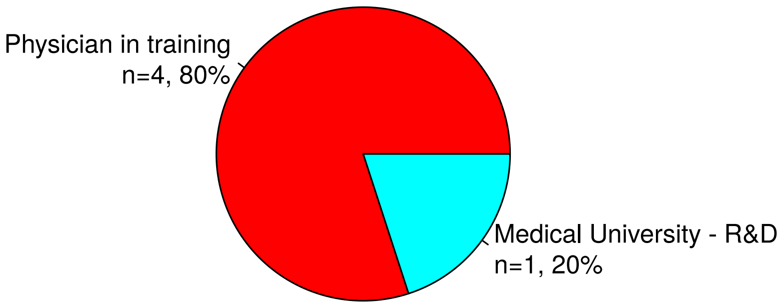


Figure 5.3: Distribution of occupational groups

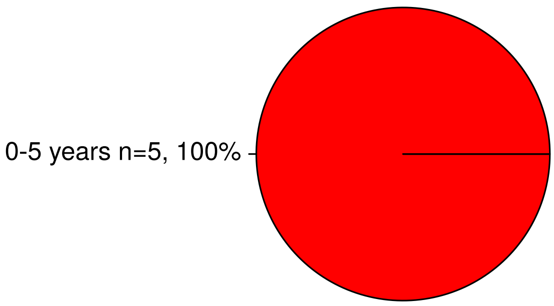


Figure 5.4: Work experience in years

The full questionnaire can be viewed in Table A1.

5.2.2 Search preferences

All of the participants reported using online resources like Wikipedia and Google to search for medical information. 80% of the participants use UpToDate® and PubMed. 20% use Google Scholar, Medical Forums and other sources. No one is using FindMeEvidence. (Figure 5.5 and 5.6) Figure 5.7 shows the distribution of the devices used to access online medical information. Everyone uses a desktop computer or laptop. Furthermore 60% are using a smartphone and only 20% a tablet.

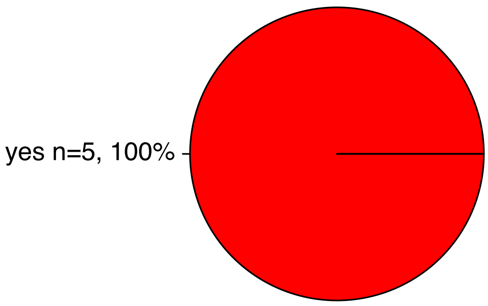


Figure 5.5: Daily usage of online resources

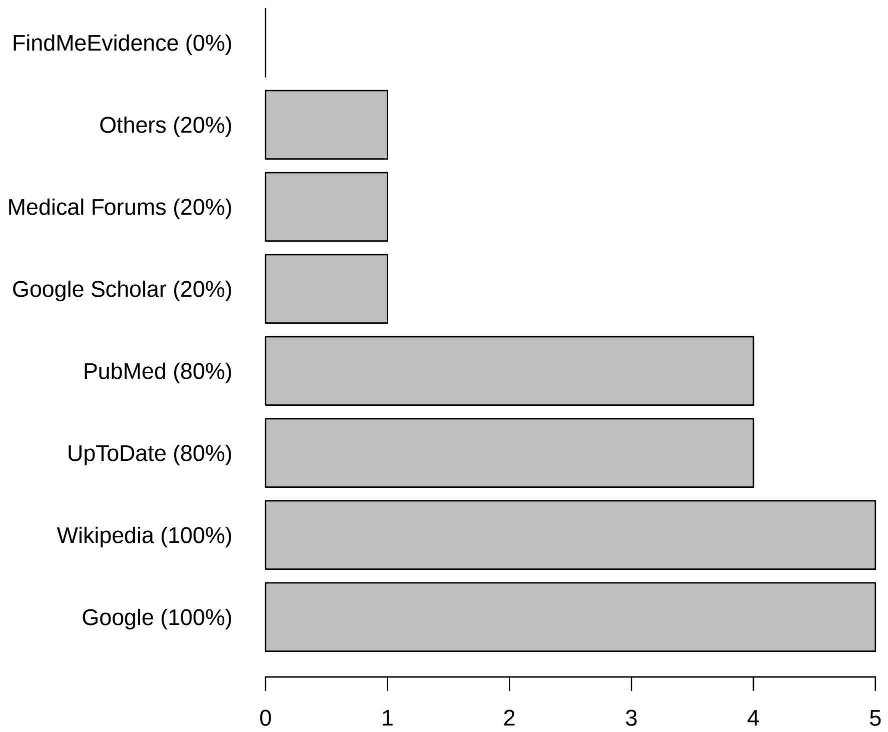


Figure 5.6: Usage of medical online resources

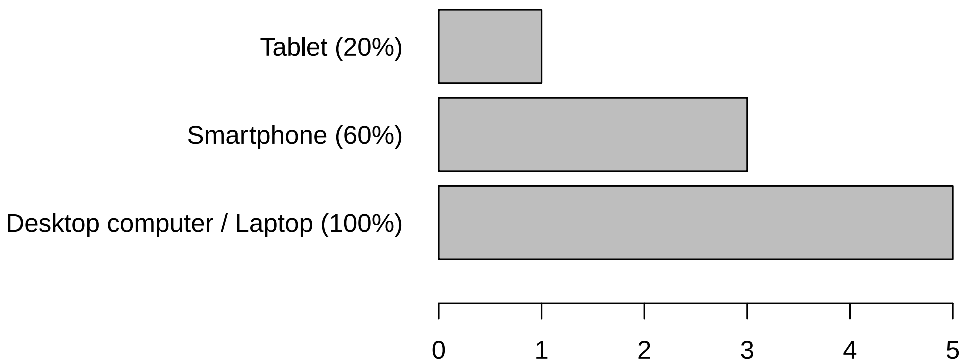


Figure 5.7: Devices used to access online medical information

The full questionnaire can be viewed in Table A2.

5.2.3 Search tasks

First the participants had to formulate a question concerning drug side effects of an arbitrary active pharmaceutical ingredient. The second question had to deal with a freely chosen disease and its symptoms. The last task was intended to perform a medical question from the everyday life. Participants wrote down the question and the time spent per task. If they could find acceptable results they also noted the query phrase and the best result. If they were not successful we kindly ask them to describe the reason for failure. The number of attempts to get to the best result was determined by a Piwik analysis as described in Section 3.2.3.

Hard- and Software used in online evaluation

As illustrated in Figure 5.8 all participants stated to use a desktop browser for the online test. An analysis of the apache httpd server log files (Section 3.2.3) reveal the combinations of browser and operating system used (Table 5.1).

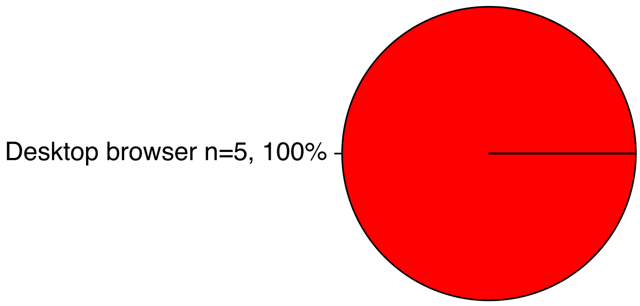


Figure 5.8: Used hardware

Browser	Operating system
Chrome 43.0	Windows 8.1
Firefox 31.0	Windows 7
Chrome 41.0	GNU/Linux
Firefox 38.0	Mac 10.6
Firefox 29.0	Windows 7

Table 5.1: Combinations of browser and operating system

The full questionnaire can be viewed in Table A2.

Search task 1

Participants were asked to formulate a question concerning drug side effects of an arbitrary active pharmaceutical ingredient and use FindMeEvidence to find an appropriate answer. 80% succeeded in solving search task 1 (Figure 5.9). Figure 5.10 illustrates the distribution of the self reported time spend on search task 1.

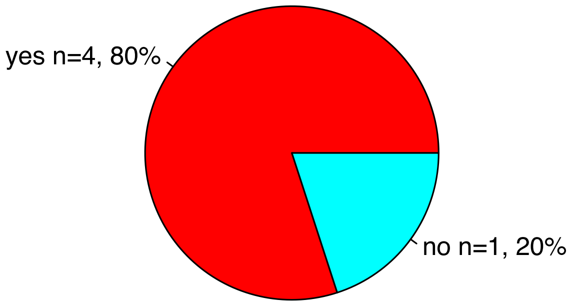


Figure 5.9: Search task 1 success

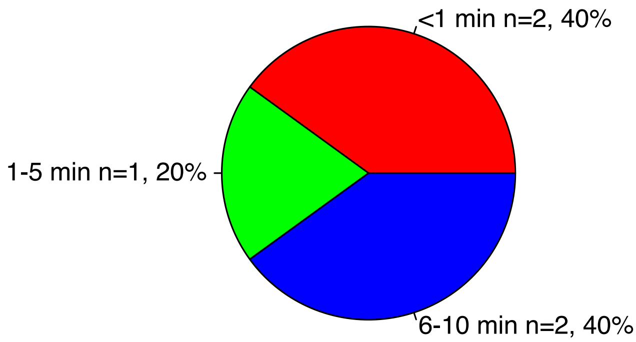


Figure 5.10: Search task 1 time duration

All successful queries are listed in Table 5.2. If the active pharmaceutical ingredient in search number 1 is misspelled: 'metamphetamine' FindMeEvidence returns no acceptable result. Successful questions number 3 and 4 are searching for the same active ingredient: Pregabalin. Number 4 uses its trade name Lyrica! Table 5.3 lists the one

No.	Active pharmaceutical ingredient (in German)	Attempts	Time	Query phrase	Best result
1	Methamphetamin	6	1-5 min	Methamphetamine	Wikipedia
2	Allopurinol	1	<1 min	Allopurinol adverse effects	Medscape
3	Pregabalin	6	6-10 min	Pregabalin renal	Medscape
4	Pregabalin	3	<1 min	Lyrica sideeffect	Medscape

Table 5.2: Successful search task 1 questions

No.	Active pharmaceutical ingredient (in German)	Attempts	Time
1	Phenprocoumon	6	6-10 min

Table 5.3: Unsuccessful search task 1 questions

and only unsuccessful question. Unfortunately it was not possible to find side effects of Phenprocoumon.
The full questionnaire can be viewed in Table A3.

Search task 2

Participants were asked to formulate a question concerning symptoms of an arbitrary disease and use FindMeEvidence to find an appropriate answer. 100% succeeded in solving the task (Figure 5.11). No participant needed more than 5 min to solve the task, one even less than 1 min (Figure 5.12).

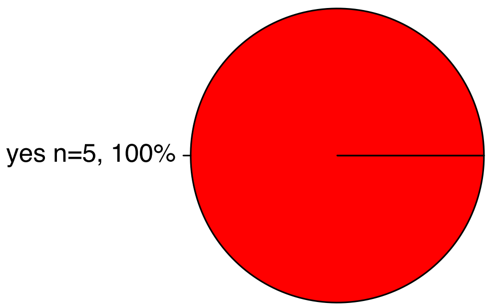


Figure 5.11: Search task 2 success

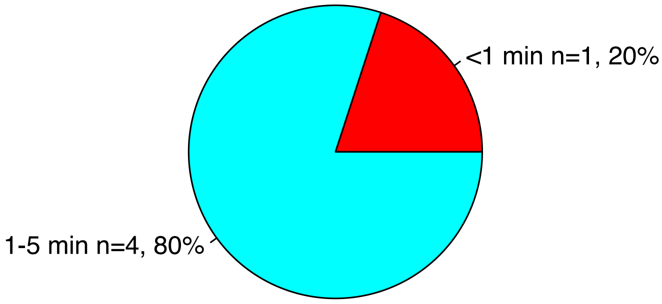


Figure 5.12: Search task 2 time duration

All successful queries are listed in Table 5.4. The translation service was activated for question number 1 and 2. It successfully translated 'Muskeldystrophie Typ Duchenne' to 'muscular dystrophy Typ Duchenne' and 'ovarialkarzinom symptome' to 'ovarian cancer symptome'. As the screenshot in Figure 5.13 illustrates, the PubMed autocomplete service delivers better results. There seems to be a usability problem with the query translation support (described in Section 4.1) in FindMeEvidence. Users did not figure out how to use the placement of the cursor to get translation support. 6 translations were offered for query phrase number 3. Unfortunately none had acceptable quality but the PubMed autocomplete and the translation service together would deliver a presentable result if correctly used (Figure 5.14). The query phrase 'muscular dystrophy typ duchenne' returns an empty result set (Figure 5.15). Whereas 'muscular dystrophy duchenne' has a much higher recall. Figure 5.16 illustrates that a filter by Wikipedia lists the article `Duchenne_muscular_dystrophy` as the number 1 search result.

No.	Disease (in German)	Attempts	Time	Query phrase	Best result
1	Muskeldystrophie Typ Duchenne	4	1-5 min	duchenne muscular dystrophy symptoms	Wikipedia
2	Ovarialkarzinom	2	1-5 min	ovarian cancer symptoms	PubMed
3	Familäre Hypercholesterinämie	2	1-5 min	familial hypercholesterolemia	Medscape
4	Psoriasis	1	1-5 min	psoriasis symptoms	Medscape
5	GBS=Guillain-Barré-Syndrom	1	<1 min	guillain barre syndrome	Medscape

Table 5.4: Successful search task 2 questions

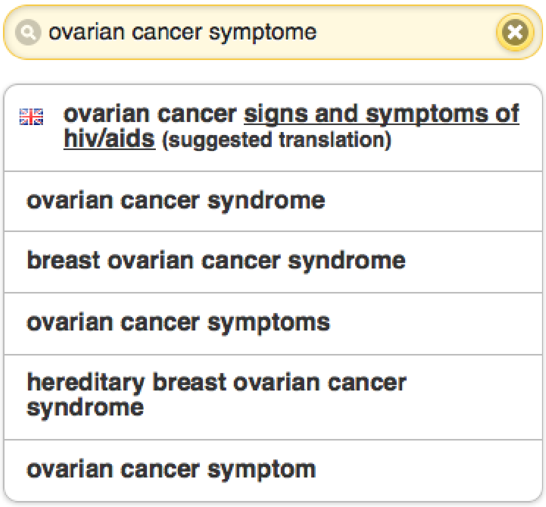
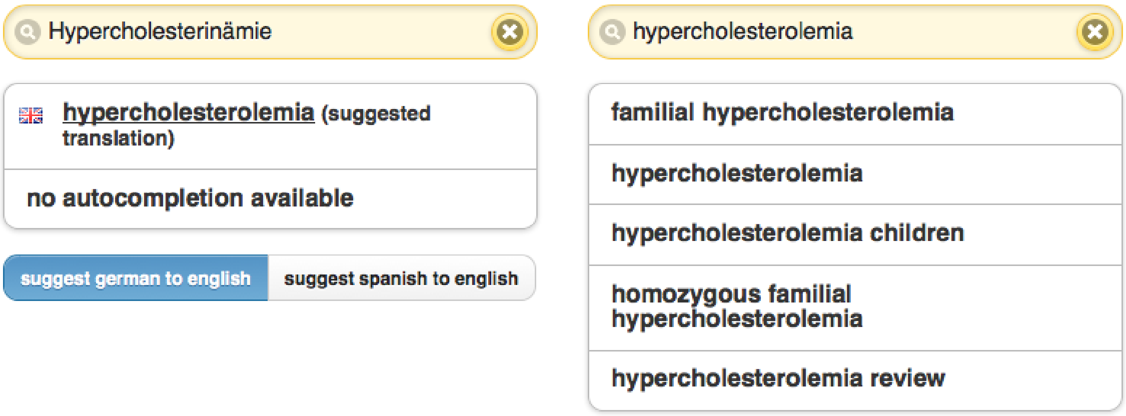


Figure 5.13: Wrong translation for ‘symptome’



(a) Translation for ‘Hypercholesterinämie’ (b) Autocomplete for ‘hypercholesterolemia’

Figure 5.14: Familiäre Hypercholesterinämie

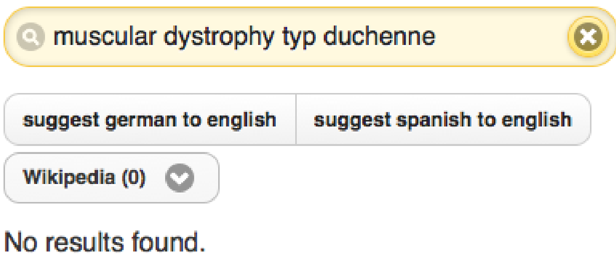


Figure 5.15: Empty SERP for ‘muscular dystrophy typ duchenne’

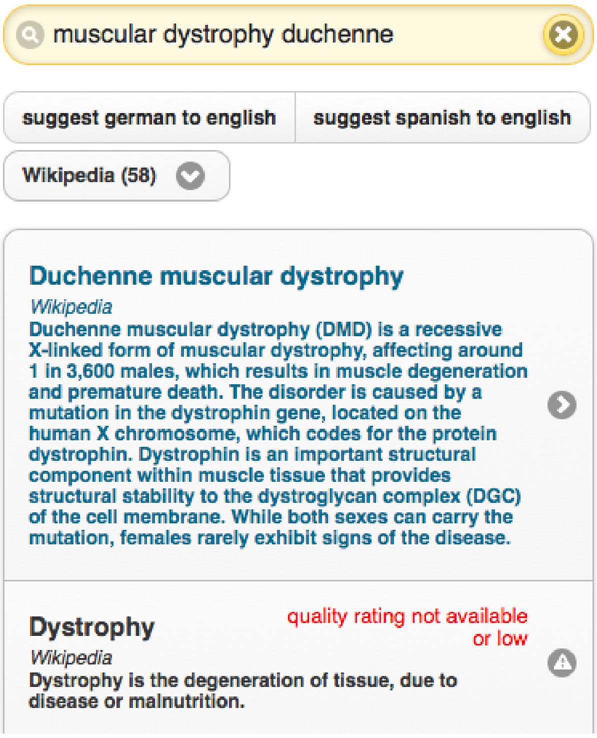


Figure 5.16: SERP for 'muscular dystrophy duchenne'

The full questionnaire can be viewed in Table A4.

Search task 3

Users were asked to use FindMeEvidence to find information relevant to a typical medical question that they had formulated themselves. More than half of the questions (60%) could not be answered (Figure 5.17). Participants spent 5-10 min on the unsuccessful search tasks (Figure 5.18).

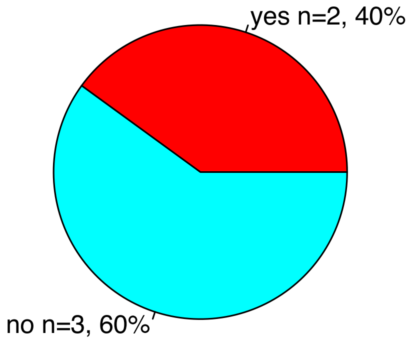


Figure 5.17: Search task 3 success

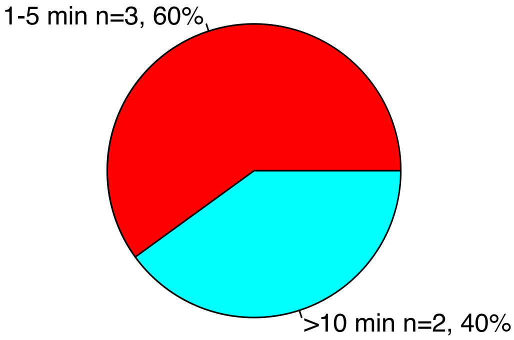


Figure 5.18: Search task 3 time duration

All successful queries are listed in Table 5.5 and all unsuccessful in Table 5.6.

No.	Medical question (in German)	Attempts	Time	Query phrase	Best result
1	Mit welchen Medikamenten interagiert Clarithromycin?	1	1-5 min	clarithromycin drug interactions	Medscape
2	Sepsis-Score	1	1-5 min	score for sepsis	PubMed

Table 5.5: Successful search task 3 questions

No.	Medical question (in German)	Attempts	Time
1	Wie hoch ist die Prävalenz kryptogener epileptischer Anfälle?	9	>10 min
2	Betablocker und Verapamil-Wechselwirkung. Halbwertszeiten. Konkret: Wie lange muss man nach Verapamilgabe durch den Notarzt (zur Herzfrequenzsenkung bei zB tachycardem Vorhofflimmern) warten um weitere Herzfrequenzsenkung mit zB Metoprolol zu betreiben.	4	>10 min
3	Schlaganfallsprophylaxe mit persistierendem Foramen ovale	3	1-5 min

Table 5.6: Unsuccessful search task 3 questions

The full questionnaire can be viewed in Table A5.

5.2.4 Search tasks overall success

Figure 5.19 summarises the results of all search tasks. The majority (73%) of the question were successfully answered with FindMeEvidence.

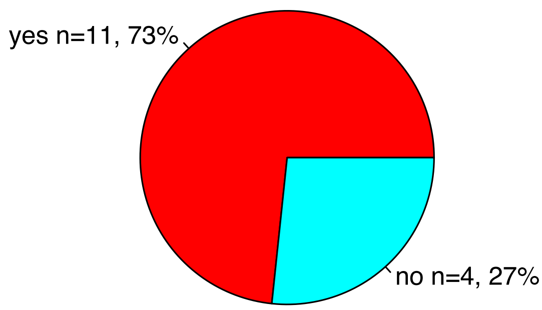


Figure 5.19: Search tasks overall success

5.2.5 Translation button

Nearly all participants (80%) reported to notice the translation button (described in Section 4.1), but only 40% used it. One of the users claimed not to notice the translation button, but he Piwik analysis clearly shows that the user had the translation service activated. This reveals another usability issue. It seems that its not that easy to distinguish the suggested translation from the PubMed autocomplete results.

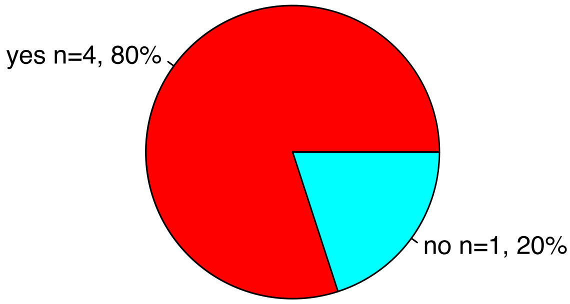


Figure 5.20: Noticeability of the translation button

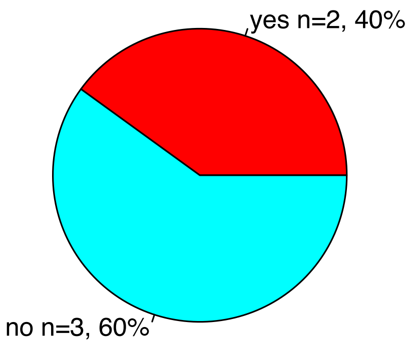


Figure 5.21: Usage of the translation button

The full questionnaire can be viewed in Table A6.

5.2.6 SUS response

Usability was determined using the 10-item SUS. A mean overall global usability score of 84 (N=5) was calculated to determine overall usability. Based on research [AT13], a SUS score above 70 would be considered acceptable and anything below 50 is not acceptable. A 5-point Likert scale was used to indicates the degree of agreement or disagreement. The five points of the scale are labelled, (1) agree strongly, (2) agree somewhat, (3) neutral, (4) disagree somewhat, (5) disagree strongly. For evaluation purposes the 10 SUS were split up in the 5 positive and he 5 negative statements about the system and evaluated separately. The full questionnaire can be viewed in Table A7.

Positive SUS items

Figure 5.22 illustrates the level of agreement to positive statements about FindMeEvidence. Nearly all participants thought the system was easy to use and would imagine that most people would learn to use this system very quickly. More than half of the users found the various functions in the system were well integrated and felt very confident using the system. Only one user would not like to use the system frequently.

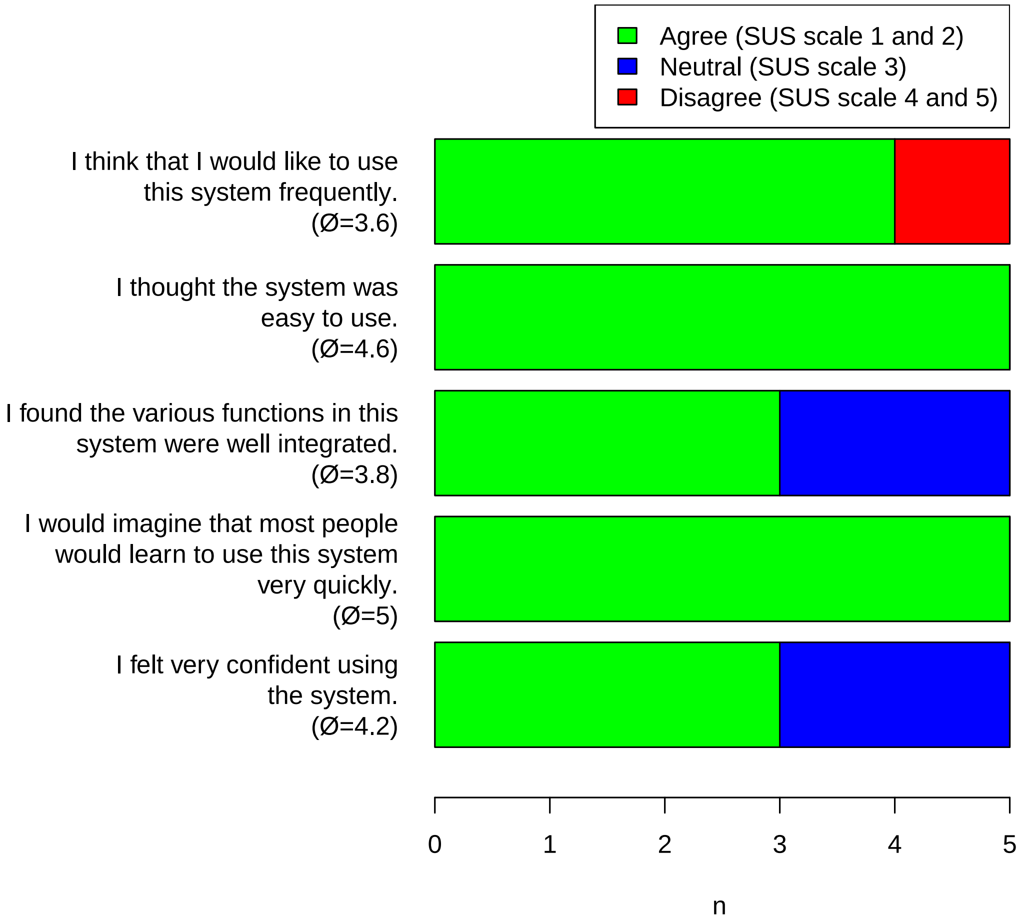


Figure 5.22: Positive SUS items

Negative SUS items

Figure 5.23 illustrates the level of disagreement to negative statements about FindMeEvidence. None of the users found the system unnecessarily complex. The participant strongly agreed that they would need no support of a technical person to be able to use the system and that there was not too much inconsistency in it. More than half of the users do not need to learn a lot of things before they could get going with the system. Just one user found the system very cumbersome to use.

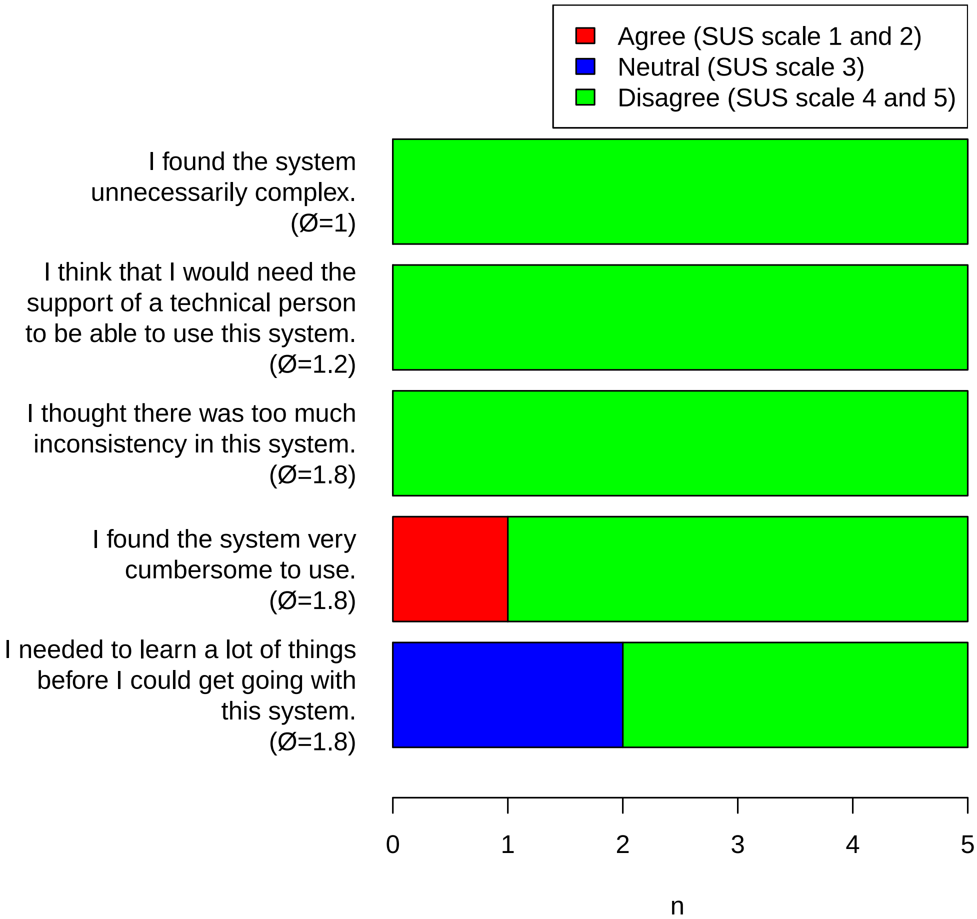


Figure 5.23: Negative SUS items

5.2.7 Final questions

The first question of the SUS: 'I think that I would like to use this system frequently.' was rephrased and asked again with still only one negative result (Figure 5.24).

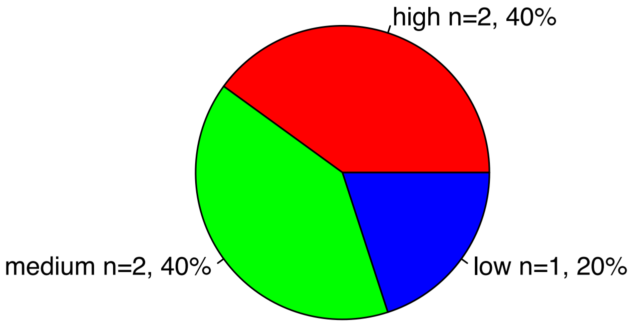


Figure 5.24: Probability to return to FindMeEvidence

The full questionnaire can be viewed in Table A8.

5.2.8 User feedback

Finally the users were also given the chance to provide feedback as well as make suggestions for improvement. The user feedback given in the previous parts and the one in the final can be summarised in the following points:

Users liked

- The search results are up-to-date.
- FindMeEvidence delivers good search results regarding best practice guidelines, case reports etc.
- The possibility to filter search results (PubMed, drug information, etc.) is very handy.
- FindMeEvidence has

  an intuitive UI,

  the possibility to translate search terms

  and is easy to use.
- FindMeEvidence seems well-suited to find studies, but the gold standard is still UpToDate®.

Users did not like / Is in need of improvement

- Users do not figure out how to use the placement of the cursor to get translation support.

- Users cannot clearly distinguish the suggested translation from the PubMed autocomplete results.
- User: 'Google/Wikipedia sometimes returns the same result, but in less than 1 min. If FindMeEvidence links to Wikipedia, it is faster to use Wikipedia directly.'
- Sort by relevance for PubMed does not return the desired result.
- Google is more suitable for complex queries.
- Translation of German terms does not work all the time.

Users suggested

- Provide a suggestion for an alternate search term if the search text contains incorrect spelling and also provide search results for the alternate search term immediately.
- Users would like to filter search results by publication type (e.g., review, meta-analysis or clinical-trial).
- Add direct accessibility to text book information e.g., symptoms.
- Users are missing previews (thumbnails) in the SERP.

5.3 Findings

The following findings were made from analysing the data:

- The usage of the translation button is not self-explanatory.
  - Users did not figure out how to use the placement of the cursor to get translation support.
  - It seems that it is not that easy to distinguish the suggested translation from the PubMed autocomplete results.
  - The dictionary only contains medical terms which sometimes lead to strange translations (e.g., Figure 5.13: symptome -> signs and symptoms of hiv/aids).
- If the search result contains an image, there should be a thumbnail available on the SERP.
- Users are missing basic text book information.

CHAPTER 6

Establishing the system as an open-source project

To allow other developers to contribute and adapt FindMeEvidence to their local needs the codebase is hosted on GitHub the de facto social coding platform for open source projects (Section 6.1). Additionally Docker a platform for packaging and distribution of software is used to easily run a local copy of FindMeEvidence (Section 6.2). Hence FindMeEvidence can run virtually anywhere on anybody's infrastructure. These efforts comply to **Task 6** to ensure the code is in a shape that allows others to contribute to the codebase and adapt it to local needs.

6.1 Software development and distribution on GitHub

FindMeEvidence was originally hosted on Google Developers (https://code.google.com/p/bricoleur-fast-medical-search/). We migrated the source code (with all associated history) and the issue tracker to GitHub (https://github.com/matthias-samwald/find-me-evidence/). GitHub allows interested developers of FindMeEvidence to fork a copy of the repository and immediately begin working on it without affecting the original project. Changes from a forked child repository can be merged back by a Pull Request.

6.2 Packaging and running of FindMeEvidence with Docker

Docker is an open-source project to build, ship, and run any application in isolated environments, called containers. To sandbox Linux processes into very lightweight containers different interfaces to access virtualisation features of the Linux kernel are used. Among the Linux kernel's virtualisation features for Docker are namespaces, cgroups, capabilities, AppArmor profiles, Netfilter, Netlink, SELinux, and so on. Each container is equipped with all the libraries and dependencies to run a certain process. A Docker container is composed of layers and all containers on a host run under the same kernel (Figure 6.1). As the container sits right on top of the operating system no Virtual Machine Monitor (VMM) that occupies about 10 to 15 percent of the resources on a host is needed. The Linux Kernel and Boot Loader (bootfs) is always layer number zero. A container is an instance of an image and the first layer is called a Base Image. It contains the rootfs. The Base Image and additional layers of a Docker container are read only images, except the last image. Docker mounts a read-write file system on top of all the other file system image layers to store the changes made to underneath images. An image ID, which is a 64-character long hexadecimal string, identifies each image. This type of filesystem is referenced to as a copy-on-write model or multilayered union filesystem. Also common portions of the operating system are shared between containers which is one reason Docker is a far lighter solution than a full Virtual Machine (VM). Images are build by reading the instructions from a Dockerfile, a text file that contains all the commands needed to build an image.

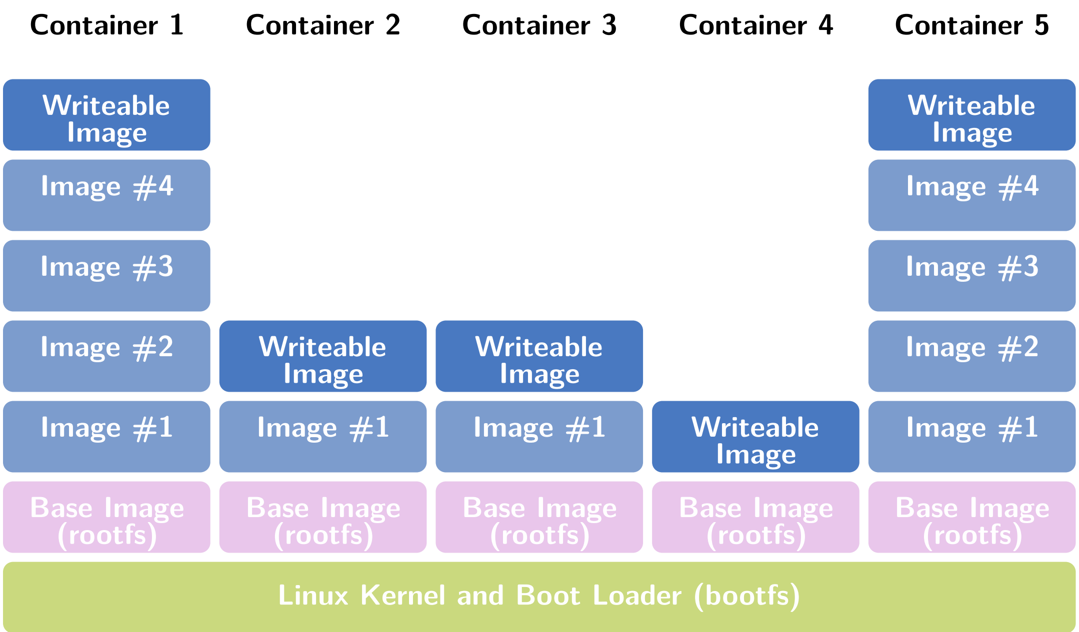


Figure 6.1: Docker architecture

6.2.1 Docker subprojects

Docker consists of 7 featured projects:

1. **Engine:** Docker Engine or ‘Docker’ creates and runs Docker containers.
2. **Registry:** The Docker Registry is an open source registry to store and distribute Docker images.

3. **Kitematic:** Docker's underlying containerization depends on Linux-specific kernel features. Kitematic can be used to run the Docker daemon natively on Windows or a Mac.

4. **Machine:** Docker Machine is used for the provisioning of Docker hosts.

5. **Swarm:** Docker Swarm is a native clustering system for Docker. Docker images can be created and controlled inside a cluster. Swarm decides on which node a container is started and also restarts and stops container.

6. **Compose:** Compose is a command line tool for defining and running multi-container applications with Docker. Everything is defined inside a YAML file (`docker-compose.yml`).

7. **Networking:** It is a multi-platform library for networking containers implemented in the programming language Go. It provides a consistent programming interface and the required network abstractions for applications.

At the time of writing Machine, Swarm, and Compose were still in BETA and not ready for production!

6.2.2 FindMeEvidence at Docker Hub

Docker, Inc host the central place to share, find, and extend Docker images, the Docker Hub: https://hub.docker.com/, which provides a free-to-use, hosted Registry. Docker Hub hosts an image of FindMeEvidence with an empty Apache Solr index. When an operator executes Listing 5 the find-me-evidence image is pulled from the Docker Hub, it

```
$ docker run -d -p 8080:8080 -p 80:80 msamwald/find-me-evidence
```

Listing 5: run msamwald/find-me-evidence

is made available locally and a new find-me-evidence container is started. Documentation and the Dockerfile can be found at the Docker Registry Hub[1]. The best practices for writing Dockerfiles recommend to run only a single process in a single container and use the linking system of Docker to link multiple containers together. Connection information is sent from one container to another one. Despite this recommendation we created a self-contained FindMeEvidence container that runs two processes, an Apache HTTP Server and Apache Solr. This makes it quite easy to deploy and further develop FindMeEvidence by just running one container. The UI of FindMeEvidence is available at http://localhost:80 (Figure 6.2) and the Apache Solr Admin UI can be accessed here: http://localhost:8080/solr (Figure 6.3).

[1] https://registry.hub.docker.com/u/msamwald/find-me-evidence/

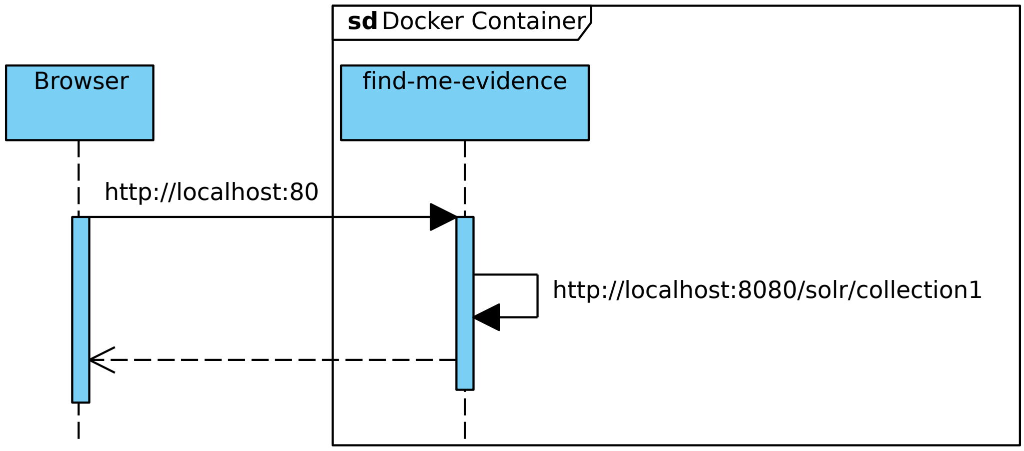


Figure 6.2: Docker test & development environment (Apache HTTP Server)

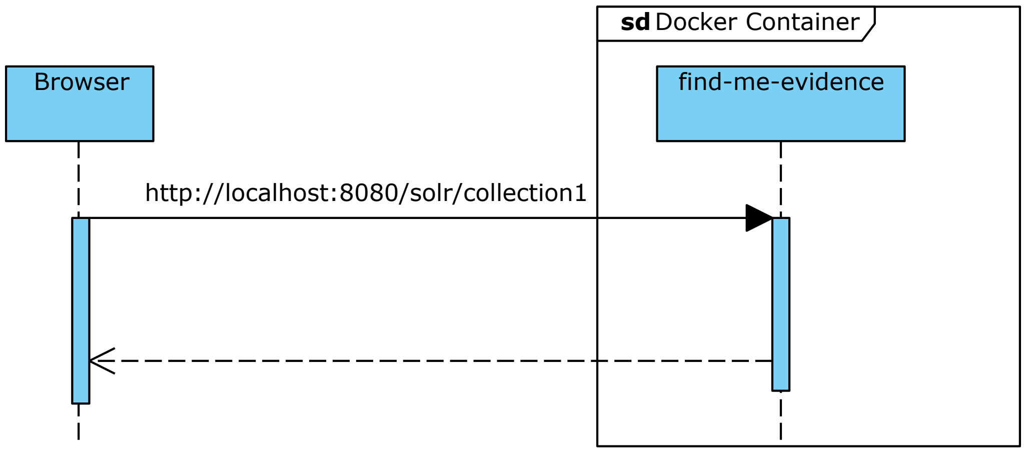


Figure 6.3: Docker test & development environment (Apache Solr)

We additionally offer two Dockerfiles on GitHub[2] to run FindMeEvidence in a production environment. This solution is shown in Figure 6.4. One container is used to host the Apache HTTP Server process (fme_apache) and the other one for Apache Solr (fme_solr). First a new fme_solr container has to be started (Listing 6). The Dockerfile[3] of fme_solr uses the EXPOSE instruction to inform Docker that the container will listen on the network port 8080 at runtime. Next fme_apache is started and Docker linking creates a secure tunnel between fme_apache and fme_solr over the exposed port 8080 (Listing 7).

[2]https://github.com/matthias-samwald/find-me-evidence/tree/master/docker/

[3]https://github.com/matthias-samwald/find-me-evidence/tree/master/docker/fme_solr/Dockerfile

```
$ docker run --restart=always \
  -v /home/path_to_index/solr4.10.4/solr:/opt/solr/example/solr -d \
  --cap-add SYS_PTRACE --security-opt=apparmor:unconfined \
  --name solr_instance \
  fme_solr
```

Listing 6: run fme_solr

```
$ docker run --restart=always \
  -d -p 80:80 --link solr_instance:solr \
  fme_apache
```

Listing 7: run fme_apache

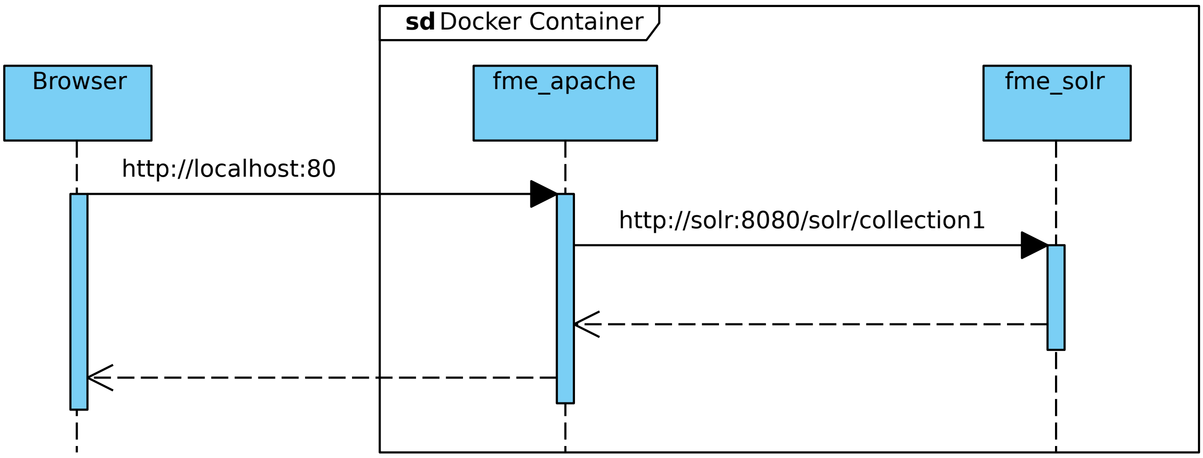


Figure 6.4: Docker production environment

Docker updates the `/etc/hosts` file on fme_apache accordingly to map the hostname solr to the IP address of fme_solr. Hence in the production environment only port 80 is forwarded to the Browser and only fme_apache can access Apache Solr over port 8080.

6.2.3 FindMeEvidence multi-container environment with Docker Compose

The Docker Compose YAML file in Listing 8 builds and runs fme_solr & fme_apache by calling `$ docker-compose up -d`. Exactly the same that is done in Listing 6 and Listing 7 explicitly.

```
fmesolr:
 build: fme_solr/
 volumes:
  - /home/path_to_index/solr4.10.4/solr:/opt/solr/example/solr
 cap_add:
  - SYS_PTRACE
 security_opt:
  - apparmor:unconfined
fmeapache:
 build: fme_apache/
 ports:
  - "80:80"
 links:
  - fmesolr:solr
```

Listing 8: docker-compose.yml

CHAPTER 7

Critical reflection

How to deal with synonyms containing multiple words is the topic of Section 7.1. Section 7.2 shortly discusses the biggest competitors of FindMeEvidence. and Section 7.3 deals with the introduction of improved for spelling correction and translation.

7.1 Better multi-word synonym handling

FindMeEvidence uses DBpedia to create a synonym dictionary based on Wikipedia page redirects. The RDF property `http://dbpedia.org/ontology/wikiPageRedirects` is used within a SPARQL query together with `rdfs:label` to get a list of synonyms (Listing 9). Based on the idea of Soldaini *et al.* [SYYT+15] FindMeEvidence could keep only those redirect terms which lead to a Wikipedia page describing a medical symptom, drug or disease.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX resource: <http://dbpedia.org/resource/>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

SELECT ?synonym_label
WHERE {?synonym dbpedia-owl:wikiPageRedirects resource:Blood_phobia .
       ?synonym rdfs:label ?synonym_label}
```

Listing 9: SPARQL query (synonym labels for wikipedia article *Blood phobia*)

The results from the SPARQL query are used to create A synonym file with explicit mapping. The arrow (=>) is used to separate the terms. Any token sequence on the left-hand side of => is replaced with all alternatives on the right-hand side (e.g. `Haemophobia => Haemophobia, Blood phobia`). FindMeEvidence injects synonyms during query processing. According to the Apache Solr docs it is recommended to expand the synonyms when indexing. The main 3 reasons for it are:

1. Counter-intuitive IDF weighting:

   The IDF boosts rare synonyms during query-time expansion. Hence documents that match the rare synonyms appear to high in the search result which may be counter intuitive to the user. Index-time expansion on the other hand will result in the same IDF for all documents.

2. No phrase search for multi-word synonyms:

   For phrase searches the QueryParser passes the entire string 'Blood phobia condition' to the analyzer. The SynonymFilter applied the rule 'Haemophobia => Haemophobia, Blood phobia' (Figure 7.1) but two terms ('Blood phobia') cannot occupy the same position. Hence there is no way to indicate that a multi-word synonym occupies the same position as a term (Figure 7.2). A workaround for this problem is the reverse mapping 'Blood phobia => Haemophobia' at index-time and query-time.

3. The QueryParser breaks up the input on white spaces before giving it to the Analyzer and therefore no multi-word synonym matches.

ST	text	Haemophobia		condition	
	raw_bytes	[48 61 65 6d 6f 70 68 6f 62 69 61]		[63 6f 6e 64 69 74 69 6f 6e]	
	start	0		12	
	end	11		21	
	positionLength	1		1	
	type	<ALPHANUM>		<ALPHANUM>	
	position	1		2	
SF	text	haemophobia	blood	condition	phobia
	raw_bytes	[68 61 65 6d 6f 70 68 6f 62 69 61]	[62 6c 6f 6f 64]	[63 6f 6e 64 69 74 69 6f 6e]	[70 68 6f 62 69 61]
	start	0	0	12	12
	end	11	11	21	21
	positionLength	1	1	1	1
	type	SYNONYM	SYNONYM	<ALPHANUM>	SYNONYM
	position	1	1	2	2

Figure 7.1: TokenStream with multi-word synonyms applied (screenshot)

On the other hand synonym injection at index-time is less flexible. The index gets larger and changes in the synonym mapping require a complete re-index. If FindMeEvidence wants to keep synonym injection during query processing it could simply discard all multi-word synonyms. Still, the counter-intuitive IDF weighting remains.

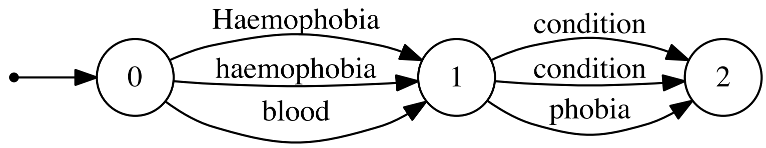


Figure 7.2: TokenStream with multi-word synonyms applied (graph)

7.2 Competition with Google and UpToDate®

The biggest freely available competitors of FindMeEvidence are Google and other major search engines (e.g., Bing, Yahoo, Yandex, and Baidu). Users participating in the online evaluation literally reported that they get the same result from Google but faster. However, they do not report that the results from FindMeEvidence as worse than results from Google. Still, the advantage in speed reported for Google for some queries should be further addressed in future development of FindMeEvidence. Among closed, commercial medical web resources participants considered UpToDate® as the gold standard. What Google and UpToDate® cannot offer is the deployment of a local, customised installation of the software tailored to the needs of specific medical institution.

7.3 Algorithms for Natural Language Processing (NLP) tasks

FindMeEvidence can benefit from the introduction of spelling correction algorithms and an improved translation service. The evaluation has clearly shown that users are missing suggestions for an alternative search term if the search text contains incorrect spelling. The translation service of FindMeEvidence is more like an autocomplete service than a translation service (Section 4.1). Hence established algorithms from NLP have to be applied.

CHAPTER 8

Summary and future work

8.1 Summary

Based on FindMeEvidence 1.0 a version 1.1 was developed, the source code made available on GitHub, deployed on http://findmeevidence.org/, and the index updated which now contains 980,452 documents (statistics calculated on August 3rd, 2015). For the new version of FindMeEvidence the following features were developed:

- Translation support for German and Spanish during query entry was implemented. It is based on a dictionary is build from WikiProject Medicine & Pharmacology article titles and preferred MeSH terms.
- An Apache Solr boost function for PubMed results, based on the release date and number of reverse citations, was developed. As the E-utilities offer no efficient way to harvest the number of reverse citations, this functionality is not made available in the current version of FindMeEvidence.
- The algorithm for finding key assertions in the abstracts of PubMed articles has been improved.
- FindMeEvidence 1.1 supports the display of all important normative trustworthiness criteria.
- OA of PMC articles is signalled to the user via the PMC OA symbol.
- Assessment data from the Wikipedia Release Version Tools is used to show warnings for articles with bad quality.
- Links to the DOI and PubReader™ are now available in the results preview.
- Several smaller enhancements and bug fixes were carried out.

To establish FindMeEvidence as an open-source project the codebase was migrated from Google Developers to GitHub. FindMeEvidence is now also available on Docker to run a local copy virtually anywhere on anybody's infrastructure.

An online evaluation of FindMeEvidence was conducted from May-July 2015. The overall SUS rating of 84 (N=5) testifies FindMeEvidence an above average rating. Also the majority (73%) of the question asked by the participants during the usability test were successfully answered with FindMeEvidence. Nevertheless we made some valuable observations regarding the translation button and missing features in FindMeEvidence.

8.2 Future work

8.2.1 Further DBpedia integration

DBpedia can be used to semantically enrich FindMeEvidence. Two simple SPARQL queries (Listing 10 and 11) reveal 5,186 instances of the class `Disease` and 5,505 of the class `Drug` in DBpedia Version 2014. DBpedia's labels and short abstracts in 30 different languages; links to images; links to external web pages; Wikipedia categories, and Yet Another Great Ontology (YAGO) categories offer plenty of semantic additional information.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

SELECT count(?disease)
WHERE {?disease rdf:type dbpedia-owl:Disease}
```

Listing 10: SPARQL query (diseases in DBpedia)

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

SELECT count(?drug)
WHERE {?drug rdf:type dbpedia-owl:Drug}
```

Listing 11: SPARQL query (drugs in DBpedia)

8.2.2 Dataset specific local FindMeEvidece installations

The FindMeEvidence 1.1 index contains documents from a clinically relevant subset of PubMed, a clinically relevant subset of Wikipedia, Merck Manual Professional Edition, Medscape, National Guideline Clearinghouse, BestBETs, and ATTRACT. We had to remove NICE Clinical Knowledge Summaries from the original image as the site is currently only available from Great Britain. There are certain use cases of FindMeEvidence with a customised local installation where only a subset of the index is needed. We are hoping to see forks of FindMeEvidence for customised local installations on GitHub soon. Another advantage of a local implementation is that it is much harder for a third party to compromise the search history.

8.2.3 Revision of the translation service

The online evaluation of FindMeEvidence has unveiled some usability issues with the translation service. As demonstrated in Figure 5.14a the correct translation is suggested and in combination with the PubMed autocomplete services (Figure 5.14b) it returns exactly what the user was looking for. Hence a better combination of these two services is necessary.

8.2.4 Extension of the Extended DisMax Query Parser

To implement a better multi-word synonym handling(Section 7.1) in FindMeEvidence an extension of the Extended DisMax Query Parser has to be written. An extension of the Extended DisMax Query Parser that splits queries into a 'normal' query and a 'synonym' query is available from The Health On the Net (HON) Foundation on GitHub: https://github.com/healthonnet/hon-lucene-synonyms/. It allows synonym expansion during query-time with no side effects.

Bibliography

[ACd+08] B. Thomas Adler, Krishnendu Chatterjee, Luca de Alfaro, Marco Faella, Ian Pye, and Vishwanath Raman. Assigning trust to wikipedia content. In *Proceedings of the 4th International Symposium on Wikis*, WikiSym '08, pages 26:1–26:12, New York, NY, USA, 2008. ACM.

[AdA07] B. Thomas Adler and Luca de Alfaro. A content-driven reputation system for the wikipedia. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 261–270, New York, NY, USA, 2007. ACM.

[AS12] Maik Anderka and Benno Stein. A breakdown of quality flaws in wikipedia. In *Proceedings of the 2Nd Joint WICOW/AIRWeb Workshop on Web Quality*, WebQuality '12, pages 11–18, New York, NY, USA, 2012. ACM.

[AT13] William Albert and Thomas Tullis. *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics.* Newnes, May 2013.

[BG09a] Jöran Beel and Bela Gipp. Google scholar's ranking algorithm: The impact of articles' age (an empirical study). pages 160–164. IEEE, April 2009.

[BG09b] Jöran Beel and Bela Gipp. Google scholar's ranking algorithm: The impact of citation counts (an empirical study). pages 439–446. IEEE, April 2009.

[BLK+09] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia - a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, September 2009.

[Blu08] Joshua E. Blumenstock. Size matters: Word count as a measure of quality. In *on Wikipedia. Computing*, pages 2007–2008, 2008.

[DG08] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113, January 2008.

[DGH+14] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 601–610, New York, NY, USA, 2014. ACM.

[DGM+15] Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. Knowledge-based trust: Estimating the trustworthiness of web sources. *CoRR*, abs/1502.03519, 2015.

[DNHK09] J Dettori, D Norvell, B Hanson, and B Kopjar. Judging an article's level-of-evidence from its abstract: Can it be done successfully? *Journal of Bone & Joint Surgery, British Volume*, 91(SUPP I):17–18, 2009.

[ERS98] David Eichmann, Miguel E. Ruiz, and Padmini Srinivasan. Cross-language information retrieval with the UMLS metathesaurus. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 72–80, New York, NY, USA, 1998. ACM.

[Fil09] Elena Filatova. Multilingual wikipedia, summarization, and information trustworthiness. *SIGIR workshop on information access in a multilingual world*, 2009.

[GCBC11] Arnaud Gaudinat, Sarah Cruchet, Celia Boyer, and Pravir Chrawdhry. Enriching the trustworthiness of health-related web pages. *Health informatics journal*, 17(2):116–126, 2011.

[GCR+12] Nicolas Griffon, Wiem Chebil, Laetitia Rollin, Gaetan Kerdelhue, Benoit Thirion, Jean-François Gehanno, and Stéfan Jacques Darmoni. Performance evaluation of unified medical language system®'s synonyms expansion to query PubMed. *BMC medical informatics and decision making*, 12:12, 2012.

[HdWBB14] James M. Heilman, Jacob F. de Wolff, Graham M. Beards, and Brian J. Basden. Dengue fever: a wikipedia clinical review. *Open Medicine*, 8(4):105–115, February 2014.

[HLS+07] Meiqun Hu, Ee-Peng Lim, Aixin Sun, Hady Wirawan Lauw, and Ba-Quy Vuong. Measuring article quality in wikipedia: Models and evaluation. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 243–252, New York, NY, USA, 2007. ACM.

[KGS+13] Marlene Kritz, Manfred Gschwandtner, Veronika Stefanov, Allan Hanbury, and Matthias Samwald. Utilization and perceived problems of online medical resources and search tools among different groups of european physicians. *Journal of Medical Internet Research*, 15(6):e122, June 2013.

[Kuc13] Rafał Kuc: *Apache Solr 4 cookbook.* Packt Pub., Birmingham, UK, 2013.

[Mas08] Ken Masters. For what purpose and reasons do doctors use the internet: a systematic review. *International Journal of Medical Informatics*, 77(1):4–16, January 2008.

[Mas14] James Maskalyk. Modern medicine comes online: how putting wikipedia articles through a medical journal's traditional process can get free, reliable information into as many hands as possible. *Open Medicine*, 8(4):116–119, February 2014.

[MBHG07] Emma Meats, Jon Brassey, Carl Heneghan, and Paul Glasziou. Using the turning research into practice (TRIP) database: how do clinicians really search? *Journal of the Medical Library Association: JMLA*, 95(2):156–163, April 2007.

[ME02] Victor M Montori and Jon O Ebbert. Trip database. *Evidence Based Medicine*, 7(4):104, 2002.

[Mer68] Robert K. Merton. The matthew effect in science the reward and communication systems of science are considered. *Science*, 159(3810):56–63, January 1968.

[MZS+06] Deborah L. McGuinness, Honglei Zeng, Paulo Pinheiro Da Silva, Li Ding, Dhyanesh Narayanan, and Mayukh Bhaowal. Investigation into trust for collaborative information repositories: A wikipedia case study. In *In Proceedings of the Workshop on Models of Trust for the Web*, pages 3–131, 2006.

[NNW+12] Eva Nourbakhsh, Rebecca Nugent, Helen Wang, Cihan Cevik, and Kenneth Nugent. Medical literature searches: a comparison of PubMed and google scholar. *Health Information and Libraries Journal*, 29(3):214–222, September 2012.

[Pat14] Jarutas Pattanaphanchai. *Trustworthiness of web information evaluation framework.* phd, University of Southampton, July 2014.

[PBWB06] V. K. Podichetty, J. Booher, M. Whitfield, and R. S. Biscup. Assessment of internet use and effects among healthcare professionals: a cross sectional survey. *Postgraduate Medical Journal*, 82(966):274–279, April 2006.

[POH13] Jarutas Pattanaphanchai, Kieron O'Hara, and Wendy Hall. Trustworthiness criteria for supporting users to assess the credibility of web information. In *Proceedings of the 22nd International Conference on World Wide Web Companion*, WWW '13 Companion, pages 1123–1130. International World Wide Web Conferences Steering Committee, 2013.

[PT14] Trey Grainger Potter and Timothy. *Solr in Action*. Manning Publications, March 2014.

[RRT13] Tony Russell-Rose and Tyler Tate. Chapter 5 - formulating the query. In Lou Rosenfeld, editor, *Designing the Search Experience*, pages 99–128. Morgan Kaufmann, 2013.

[Rub08] Jeffrey Rubin. *Handbook of usability testing : how to plan, design, and conduct effective tests*. John Wiley & Sons, 2008.

[SH03] Ariel S. Schwartz and Marti A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text, 2003.

[SH14] Matthias Samwald and Allan Hanbury. An open-source, mobile-friendly search engine for public medical knowledge. *Studies in Health Technology and Informatics*, 205:358–362, 2014.

[SSP+14] Matthias Samwald, Veronika Stefanov, Georg Petz, Claus-Dieter Volko, and Allan Hanbury. Improving the quality and findability of open medical information on the web: Medical wikidata and the FindMeEvidence search engine. In *Proc. Medicine 2.0*, 2014.

[SYH03] Herbert Van de Sompel, Jeffrey A. Young, and Thomas B. Hickey. Using the OAI-PMH ... differently. *D-Lib Magazine*, 9(7-8), 2003.

[SYYT+15] Luca Soldaini, Andrew Yates, Elad Yom-Tov, Ophir Frieder, and Nazli Goharian. Enhancing web search in the medical domain via query clarification. *Information Retrieval Journal*, pages 1–25, July 2015.

[TN06] Hangwi Tang and Jennifer Hwee Kwoon Ng. Googling for a diagnosis–use of google as a diagnostic aid: internet based study. *BMJ (Clinical research ed.)*, 333(7579):1143–1145, December 2006.

[VLS+08] Ba-Quy Vuong, Ee-Peng Lim, Aixin Sun, Minh-Tam Le, Hady Wirawan Lauw, and Kuiyu Chang. On ranking controversies in wikipedia: Models and evaluation. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 171–182, New York, NY, USA, 2008. ACM.

[WCG05] Johanna I. Westbrook, Enrico W. Coiera, and A. Sophie Gosling. Do online information retrieval systems help experienced clinicians answer clinical questions? *Journal of the American Medical Informatics Association: JAMIA*, 12(3):315–321, June 2005.

[You10] Paula Younger. Internet-based information-seeking behaviour amongst doctors and nurses: a short review of the literature. *Health Information and Libraries Journal*, 27(1):2–10, March 2010.

[ZAD+06] Honglei Zeng, Maher A. Alhossaini, Li Ding, Richard Fikes, and Deborah L. McGuinness. Computing trust from revision history. In *Proceedings of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services*, PST '06, pages 8:1–8:1, New York, NY, USA, 2006. ACM.

Glossary

Apache Solr (http://lucene.apache.org/solr/) Apache Solr is an open source enterprise search platform from the Apache Lucene project. xi, xxi, 5, 11, 19–21, 31, 32, 55–57, 60, 63

ATTRACT (http://www.attract.wales.nhs.uk/) ATTRACT, a department of the National Public Health Service (NPHS) in Wales, provides evidence based summaries to clinical queries as a service. Doctors in the NHS send their clinical queries to ATTRACT. They search the evidence, appraise and summarise onto a side of A4 and make them available on the website. xiii, 6, 64

BestBETs (http://www.bestbets.org/) BestBETs gives access to current evidence on a wide range of clinical topics. BETs were developed in the Emergency Department of Manchester Royal Infirmary, UK, to provide rapid evidence-based answers to real-life clinical questions, using a systematic approach to reviewing the literature. xiii, 6, 64

bootfs The bootfs contains the boot loader and the kernel. 54

DBpedia (http://dbpedia.org/) DBpedia is a crowd-sourced community effort to extract structured information from Wikipedia [BLK+09]. This structured information can be queried through a SPARQL endpoint. xxi, 32, 59, 64

Docker (https://www.docker.com/) Docker is an open-source project to build, ship, and run any application in isolated environments, called containers. xi, xiii, 53, 64

Extended DisMax Query Parser The Extended DisMax Query Parser (eDisMax) is a combination of two other query parsers, the Lucene query parser and Disjunction Max (DisMax) query parser. It queries multiple fields with different boosts, based on the significance of each field. 65

GitHub (https://github.com/) GitHub is a web-based Git repository hosting service, which offers all of the distributed revision control and source code management (SCM) functionality of Git as well as adding its own features. xiii, xvi, 53, 56, 63–65

Google Developers (https://developers.google.com/) Google Developers (previously Google Code) is Google's site for software development tools, application programming interfaces (APIs), and technical resources. 53, 64

Google Forms (https://docs.google.com/forms/) Google Forms is a tool that allows collecting information from users via a personalized survey or quiz. The information is then collected and automatically connected to a spreadsheet with the same name. 12

Google Scholar (http://scholar.google.com/) Google Scholar is a freely accessible web search engine that indexes the full text or metadata of scholarly literature across an array of publishing formats and disciplines.. 21, 38

IDF The inverse document frequency (IDF) weights down common terms that are very frequent in documents and weights up rare terms. 60

jQuery Mobile (http://jquerymobile.com/) jQuery Mobile is a touch-optimised HTML5-based user interface system currently being developed by the jQuery project team. 11, 32

Knowledge Vault The Knowledge Vault is a knowledge base created by Google [DGH+14] where all the facts are accumulated. 8

MediaWiki API (https://www.mediawiki.org/wiki/API:Langlinks) The MediaWiki API is a Web service that provides convenient access to Wikipeda features, data, and meta-data over HTTP. 19

Medscape (http://emedicine.medscape.com/home/) Medscape is a web resource that offers daily medical news and a drug & disease database. The content is available free of charge. xiii, 6, 64

Merck Manual Professional Edition (http://www.merckmanuals.com/professional/) The Merck Manual Professional Edition is a medical reference book produced by pharmaceutical company Merck & Co. that provides health care practitioners and students with practical explanations of what to do to diagnose and treat conditions in all of the major medical and surgical specialties. The full text is available free online. xiii, 6, 64

n-grams An n-gram is a contiguous sequence of n items from a given sequence of text or speech. 20

National Guideline Clearinghouse (http://www.guideline.gov/) National Guideline Clearinghouse is a public resource for evidence-based clinical practice guidelines. xiii, 6, 64

Netbeans (https://netbeans.org/) NetBeans is a software development platform written in Java. 11

Piwik (http://piwik.org/) Piwik is a free and open source web analytics platform. It tracks online visits to one or more websites and displays reports on these visits for analysis. 13, 31, 40, 48

PMC (http://www.ncbi.nlm.nih.gov/pmc/) PubMed Central (PMC) is a free digital repository that archives publicly accessible full-text scholarly articles that have been published within the biomedical and life sciences journal literature. 24, 30, 63

precision Precision is the probability that the search result deals with the same concepts as the query. 2, 21

PubMed (http://www.ncbi.nlm.nih.gov/pubmed/) PubMed is a free search engine accessing primarily the MEDLINE database of references and abstracts on life sciences and biomedical topics. In addition to MEDLINE, PubMed provides access to PMC citations. xi, xiii, 2, 5, 6, 8, 17, 21–24, 26, 30–32, 38, 43, 48, 51, 52, 63–65

PubReader™ (http://www.ncbi.nlm.nih.gov/pmc/about/pubreader/) The PubReader view is an alternative web presentation that offers another, more reader-friendly way to read literature in PMC. xi, 27, 30, 63

recall Recall is the probability that a relevant document is retrieved in a search. 6, 43

rootfs The rootfs includes the typical linux directory structure. 54

SPARQL SPARQL (a recursive acronym for SPARQL Protocol and RDF Query Language) is a syntactically-SQL-like language for querying RDF graphs via pattern matching. 59, 64, 73

The Health On the Net (HON) (http://www.healthonnet.org/) The Health On the Net (HON) Foundation is an organisation that certifies those health-related websites that meet specific reliability standards. 8, 65

TRIP database (https://www.tripdatabase.com/) The TRIP (Turning Research into Practice) database [ME02] is a free clinical search engine that offers healthcare professionals a list of online resources useful for evidence-based practice. 6

UID Each Entrez database refers to the data records within it by an integer ID called a UID. Examples of UIDs are GI numbers for Nucleotide and Protein, PMIDs for PubMed, or MMDB-IDs for Structure. 22

UpToDate® (http://www.uptodate.com/home/) UpToDate® is an evidence-based clinical decision support resource that requires a subscription fee. xvi, 1, 38, 51, 61

virtual appliance A virtual appliance is a pre-configured virtual machine image ready to run on a hypervisor. 13

VirtualBox (https://www.virtualbox.org/) VirtualBox is a hypervisor for x86 computers from Oracle Corporation. 13

WikiProject Medicine & Pharmacology (https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Medicine/ & https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Pharmacology/) A WikiProject is a group of contributors that help to coordinate and organise creating and improving articles. The WikiProject Medicine team consists of people interested in medical and health content on Wikipedia. WikiProject Pharmacology deals with articles about pharmacology and science of medications and other pharmacology-related topics. 5, 8, 19, 28, 30, 32, 63

YAML YAML is a recursive acronym for 'YAML Ain't Markup Language' and is a human-readable data serialisation language. 57

Yandex.Translate API (http://api.yandex.com/translate/) The Yandex.Translate API is a Web service for online translation service of a russian internet search company. 19, 32

Acronyms

Ajax asynchronous JavaScript + XML. 30, 32

API Application Programming Interface. 28, 30

CC Creative Commons. 24

DOI Digital Object Identifier. xi, 30, 63

E-utilities Entrez Programming Utilities. 22, 30, 63

IDE Integrated Development Environment. 11

KBT Google's Knowledge-Based Trust. 8

MeSH Medical Subject Headings. 6, 19, 63

NCBI National Center for Biotechnology Information. 30

NLP Natural Language Processing. 61

OA Open Access. xi, 24, 26, 63

OAI-PMH Open Archives Initiative Protocol for Metadata Harvesting. 24, 26

QA Question Answering. 2

SERP Search Engine Results Page. 6, 7, 21, 44, 45, 52

SQL Structured Query Language. 11

SUS Standard Usability Scale. xi, 13, 48–50, 64

UMLS Unified Medical Language System. 6

VM Virtual Machine. 54

VMM Virtual Machine Monitor. 54

XPath XML Path Language. 11, 23

YAGO Yet Another Great Ontology. 64

Appendices

Orientation e-mail

Herzlich Willkommen!

Liebe Testperson,

Wissenschaftliche Studien haben gezeigt, dass mehr als die Hälfte der Ärzte angibt, durch Informationen im Netz in der Wahl der Behandlung beeinflusst worden zu sein. Eine kürzlich von uns durchgeführten Umfrage unter 500 europäischen Ärzten ergab, dass dabei viele auf klassische, frei verfügbare Webressourcen wie Google und Wikipedia zurückgreifen.
Da diese Ressourcen aber auch viele irrelevante Informationen und teilweise auch gefährliche Fehlinformationen liefern, wäre es von großem Nutzen für das Gesundheitswesen, wenn für die medizinische Praxis optimierte Suchmaschinen frei im Netz verfügbar wären. Das Ziel des Projektes "FindMeEvidence" ist die Schaffung eines solchen, frei zugänglichen Systems.
Diese Studie dient zur Untersuchung der Benutzbarkeit des Programms FindMeEvidence. Unser Ziel ist herauszufinden wie gut eine Benutzerin oder ein Benutzer mit dieser Software zurecht kommt. Im Idealfall hat er oder sie dieses Programm zuvor noch nie verwendet.
Wir möchten betonen, dass bei diesem Test nicht Ihre Fähigkeiten getestet werden, sondern die Software auf dem Prüfstand steht. Sie können dabei keine Fehler machen. Falls Probleme auftreten, liegt es an dem Programm.

Testablauf:
Nach einem kurzen Screening Fragebogen werden Fragen zur Computernutzung gestellt. Anschließend haben wir 3 Suchaufgaben vorbereitet. Hierfür überlegen sie sich bitte jeweils eine Suchabfrage für "FindMeEvidence", führen diese auf http://findmeevidence.org/ durch und beantworten zu jeder Suchaufgabe daraufhin 6 Fragen.
Abschließend werden noch Fragen zu ihrer Erfahrung mit "FindMeEvidence" gestellt.
Sie können jederzeit mit einem Klick auf http://goo.gl/forms/0aixTXIsyj beginnen.

Vielen Dank im Vorhinein für die Unterstützung!

User test questionnaire

Tables A1 to A8 contain the full user test questionnaire.

Question	Response
Ausbildung	❍ Arzt in Ausbildung ❍ Facharzt - Selbstständig ❍ Praktischer Arzt - Selbstständig ❍ Spitalsarzt - Angestellt in Spital, Klinik, Rehabzentrum etc. ❍ Universität - Medizinische Forschung und Lehre ❍ kein Mediziner
Geschlecht	❍ männlich ❍ weiblich
Alter	❍ 18 bis 25 ❍ 26 bis 32 ❍ 33 bis 40 ❍ 41 bis 50 ❍ Über 50
Berufserfahrung als Mediziner in Jahren	❍ 0 bis 5 ❍ 6 bis 10 ❍ 11 bis 15 ❍ 15 oder mehr

Table A1: Demographics

Question	Response
Verwenden Sie im Zuge ihrer Arbeit als MedizinerIn frei verfügbare Webresourcen wie Google und Wikipedia?	❍ ja ❍ nein
Welche Geräte verwenden Sie um nach medizinischer Information zu suchen?	❍ Smartphone ❍ Tablet ❍ Desktop Computer / Laptop
In welchen Quellen haben Sie bereits nach medizinischer Information gesucht?	❒ Google ❒ UpToDate ❒ Wikipedia ❒ Pubmed ❒ Google Scholar ❒ Medizinische Foren ❒ FindMeEvidence ❒ Sonstiges:
Wählen Sie den Gerätetyp aus, der ihrem für die folgenden Suchaufgaben verwendeten Gerät am ehesten entspricht	❍ Desktopbrowser ❍ Smartphone ❍ Tablet

Table A2: Search preferences

Question	Response
In dieser Aufgabe sollen Sie FindMeEvidence verwenden, um Informationen zu Nebenwirkungen von einem pharmazeutischem Wirkstoff ihrer Wahl zu finden.	Wählen Sie einen pharmazeutischen Wirkstoff aus, den Sie für diese Aufgabe verwenden möchten, und tippen Sie dessen Bezeichnung hier ein.
Konnten Sie ein zufriedenstellendes Ergebnis finden?	❍ ja ❍ nein
Falls ihre Suche zu einem zufriefdenstellenden Ergebnis führte, welche Webseite (aus der Trefferliste) ist ihrer Meinung nach das beste Ergebnis?	Web-Addresse
Falls ihre Suche zu einem zufriedenstellenden Ergebnis führte:	Welche Suchanfrage führte zum Erfolg?
Falls ihre Suche zu keinem zufriedenstellenden Ergebnis führte:	Wo gab es ihrer Meinung nach Probleme?
Geschätzte Zeit, die Sie für die Suche verwendet haben	❍ Weniger als 1 Minute ❍ 1 - 5 Minuten ❍ 6 - 10 Minuten ❍ Länger als 10 Minuten

Table A3: Search task 1 - Suche nach Nebenwirkungen von einem beliebigen Wirkstoff

Question	Response
In dieser Aufgabe sollen Sie FindMeEvidence verwenden, um Informationen zu Symptomen einer beliebigen Erkrankung ihrer Wahl zu finden.	Wählen Sie eine beliebige Erkrankug aus, und tippen Sie deren Bezeichnung hier ein.
Konnten Sie ein zufriedenstellendes Ergebnis finden?	❍ ja ❍ nein
Falls ihre Suche zu einem zufriefdenstellenden Ergebnis führte, welche Webseite (aus der Trefferliste) ist ihrer Meinung nach das beste Ergebnis?	Web-Addresse
Falls ihre Suche zu einem zufriedenstellenden Ergebnis führte:	Welche Suchanfrage führte zum Erfolg?
Falls ihre Suche zu keinem zufriedenstellenden Ergebnis führte:	Wo gab es ihrer Meinung nach Probleme?
Geschätzte Zeit, die Sie für die Suche verwendet haben	❍ Weniger als 1 Minute ❍ 1 - 5 Minuten ❍ 6 - 10 Minuten ❍ Länger als 10 Minuten

Table A4: Search task 2 - Suche nach Symptomen einer beliebigen Erkrankung

Question	Response
Führen Sie eine Suche nach einem medizinischen Thema durch, die Sie sonst vielleicht über eine herkömmliche Suchmaschine (z.B. google.at) gemacht hätten.	Bitte Beschreiben Sie die Fragestellung in ein paar Worten.
Konnten Sie ein zufriedenstellendes Ergebnis finden?	❍ ja ❍ nein
Falls ihre Suche zu einem zufriefdenstellenden Ergebnis führte, welche Webseite (aus der Trefferliste) ist ihrer Meinung nach das beste Ergebnis?	Web-Addresse
Falls ihre Suche zu einem zufriedenstellenden Ergebnis führte:	Welche Suchanfrage führte zum Erfolg?
Falls ihre Suche zu keinem zufriedenstellenden Ergebnis führte:	Wo gab es ihrer Meinung nach Probleme?
Geschätzte Zeit, die Sie für die Suche verwendet haben	❍ Weniger als 1 Minute ❍ 1 - 5 Minuten ❍ 6 - 10 Minuten ❍ Länger als 10 Minuten

Table A5: Search task 3 - Typische Suche aus dem medizinischen Alltag

Question	Response
Sind Ihnen diese beiden Buttons ("suggest german to english" und/oder "suggest spanish to english") aufgefallen?	❍ ja ❍ nein
Haben Sie diese auch verwendet?	❍ ja ❍ nein

Table A6: Übersetzungsfunktion

Question		Response	
Ich denke, dass ich das System häufig verwenden würde.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich fand das System unnötigerweise komplex.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich denke, das System war einfach zu benutzen.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich denke, dass ich die Unterstützung einer technisch erfahrenen Person benötige, um das System verwenden zu können.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich halte die verschiedenen Funktionen des Systems für gut integriert.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Für mich wirkte das System zu inkonsistent.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich kann mir vorstellen, dass man die Benutzung des Systems sehr schnell erlernen kann.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich fand das System sehr mühsam zu benutzen.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich fühlte mich bei der Nutzung des Systems sehr sicher.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu
Ich müsste mir noch einige Dinge aneignen, bevor ich mit dem System zurecht kommen würde.	stimme überhaupt nicht zu	❍❍❍❍❍	stimme vollkommen zu

Table A7: SUS

Question	Response
Wenn Sie in Zukunft ein Suchmaschine zur Entscheidungsunterstützung in der medizinischen Routine benötigen, wie hoch ist die Wahrscheinlichkeit, dass Sie FindMeEvidence verwenden werden?	❍ hoch ❍ mittel ❍ niedrig
Was hat Ihnen gut an FindMeEvidence gefallen? Was hat Ihnen nicht gefallen? Haben Sie vielleicht Verbesserungsvorschläge?	

Table A8: Final Questions