IMPROVING TEXT ACCESSIBILITY AND UNDERSTANDING OF DOMAIN-SPECIFIC INFORMATION

Teză destinată obţinerii
titlului ştiinţific de doctor inginer
la
Universitatea *Politehnica* Timişoara
în domeniul CALCULATOARE ŞI TEHNOLOGIA INFORMAŢIEI
de către

ing. Vasile Topac

Conducător ştiinţific:	prof.univ.dr.ing Vasile Stoicu-Tivadar
Referenţi ştiinţifici:	prof.univ.dr. ing. Rodica Potolea
	prof.univ.dr. Viorel Negru
	prof.univ.dr.ing. Ştefan Holban

Ziua susţinerii tezei: 07 Feb 2014

Seriile Teze de doctorat ale UPT sunt:

1. Automatică
2. Chimie
3. Energetică
4. Ingineria Chimică
5. Inginerie Civilă
6. Inginerie Electrică
7. Inginerie Electronică şi Telecomunicaţii
8. Inginerie Industrială
9. Inginerie Mecanică
10. Ştiinţa Calculatoarelor
11. Ştiinţa şi Ingineria Materialelor
12. Ingineria sistemelor
13. Inginerie energetică
14. Calculatoare şi tehnologia informaţiei
15. Ingineria materialelor
16. Inginerie şi Management

Universitatea Politehnica Timişoara a iniţiat seriile de mai sus în scopul diseminării expertizei, cunoştinţelor şi rezultatelor cercetărilor întreprinse în cadrul Şcolii doctorale a universităţii. Seriile conţin, potrivit H.B.Ex.S Nr. 14 / 14.07.2006, tezele de doctorat susţinute în universitate începând cu 1 octombrie 2006.

România, 300159 Timişoara, Bd. Republicii 9,
Tel./fax 0256 403823
e-mail: editura@edipol.upt.ro

Acknowledgement

First of all I would like to thank my advisor, prof. dr. ing. Vasile Stoicu-Tivadar. He has helped me a lot during my activity as PhD student in the Department of Automation and Applied Informatics, enjoying together with me the happy moments along the way and supporting me during the difficult periods, especially in some difficult moment when I was ready to give up. For all of this I am thankful to him.

I would like to thank other department members like prof. Lacramioara Stoicu-Tivadar and Dorin Berian, colleagues like Daniel, Norbert, Mihaela, Valentin and all others for their help and collaboration.

Also I am thankful to prof. Valentina Emilia Balas, who was my advisor before starting this PhD study and who has opened my appetite for research; without her, probably I wouldn't have started on this research path.

I would like to thank Luz Rello, a talented researcher on dyslexia from Pompeu Fabra University; she has been a great collaborator and a good friend. Also I would like to thank Eugene Borodin for his advices and mentorship on the Google Student Award and not only.

Special thanks go to my family, including my parents, my parents in law and all others, for all their love, care and support.

The period of this PhD study has been an extraordinary one not only professionally, but also in my personal life. In this time, I got married and I became a father twice. My two daughters, Sofia and Maria had a big role even in my research activity, boosting my motivation and energy to do all this work.

Most of all I want to thank my wife and my love, Cristina. She is the biggest (indirect) contributor to this work, by creating the conditions for doing all this research by means of offering me her love, care, trust and in many times by sacrificing herself. That is why I dedicate this work to her, and to our little daughters.

Last, but not least, I want to thank God, the beginning of all wisdom.

Timişoara, Feb 2014 Vasile Topac

Vasile, Topac
Improving text accessibility and understanding of domain-specific information

Teze de doctorat ale UPT, Seria 14, Nr. 18, Editura Politehnica, 2014, 138 pagini, 50 figuri, 16 tabele.

ISSN: 2069-8216
ISSN-L: 2069-8216
ISBN: 978-606-554-775-9

Cuvinte cheie:
accessibility, terminology, health literacy, language adaptation, NLP, HCI, transcoding

Rezumat:
This thesis explores ways to make text more accessible for end users, taking into consideration the multiple *layers* of text. Most attention is given to the specialized language layer, where lay users encounter issues while accessing domain-specific information. Techniques and tools for accurately identifying and explaining specialized terminology, even if present in derivate form, are presented. The impact of this process is evaluated on medical language and results reflect an improvement of message understanding for lay persons.
Additionally, means to adapt the text at presentation level are presented, giving special attention to users with dyslexia or low vision. The resulting tools were designed with careful attention to availability and usability, making them run anywhere with minimal effort.
Integration of the developed services within a model of universal text accessibility is also presented.

Table of content

List of tables

List of figures

Abbreviations

NLP	– Natural Language Processing
ITS	– Internationalization Tag Set
MT	– Machine Translation
SMT	– Statistical Machine Translation
FHM	– Fuzzy Hash Map
IMIA	– International Medical Interpreters Association
UML	– Unified Modeling Language
WSD	– Word Sense Disambiguation
FP	– False positive
TP	– True positive
FN	– False negative
HTML	– Hyper Text Markup Language
CSS	– Cascade Style Sheets
JS	– Java Script
WCAG	– Web Content Accessibility Guidelines
IR	– Information Retrieval
AMT	– Amazon Mechanical Turk
HIT	– Human Intelligence Task
URL	– Uniform Resource Locator
GUI	– Graphical User Interface
HCI	- Human Computer Interaction
W3C	- World Wide Web Consortium

Abstract

This dissertation explores ways to make text more accessible for end users, taking into consideration the multiple *layers* of text (language, media format and presentation layer). Most attention is given to the specialized language layer, where lay users encounter difficulties while accessing specialized information, the author choosing as case study the medical language. Methods and tools for accurately identifying and explaining specialized terminology, even if present in derivate form, are presented. These methods are based on fuzzy matching techniques combined with some specialized data structures created by the author. Techniques for improving precision of terminology fuzzy matching based on a) semi-supervised methods or unsupervised methods and b) human based validation (like users' feedback or responses from *crowdsourcing* platforms) are also presented. The impact of the process of explaining terminology was evaluated on medical language by performing user studies and results reflect an improvement in terms of message understanding for lay persons.

Additionally, means to adapt the text at presentation level are presented, giving special attention to users with dyslexia or low vision. The adaptations of text at multiple levels and the interactions between levels are presented combined in a universal text accessibility model. Several tools were designed and developed to put in practice the techniques explored in this work. The tools were designed with careful attention to availability and usability, making them run anywhere with minimal effort.

Thesis statement:

Adapting specialized text by identifying and explaining terminology in both canonical and derivate form and providing means to change the text at presentation level can improve text accessibility and understanding for lay users.

Rezumat

Această lucrare explorează căi prin care poate fii crescută accesibilitatea textului pentru utilizatorii de rând, luând în considerare multiple nivele de limitare ale textului (limitări la nivel de limbaj, format media sau la nivelul de prezentare). Cea mai mare atenţie este acordată dificultăţii de înţelegere a textului datorate limbajului de specialitate, alegându-se ca şi domeniu de studiu limbajul medical. Metode şi instrumente pentru recunoaşterea şi explicarea terminologiei medicale, chiar şi atunci când aceasta apare în formă derivată sunt prezentate. Recunoaşterea termenilor în limbajul natural e bazată pe tehnici de tipul fuzzy matching combinate cu structuri de date specializate, dezvoltate de către autor. Tehnici pentru îmbunătăţirea preciziei recunoaşterii aproximative (*fuzzy matching*) bazate atat pe metode semi-supervizate sau autonome cât şi pe metode ce au la bază inteligenţa umană (de exemplu platformele de *crowdsourcing*) sunt prezentate. Aceste metode de adaptare a textelor de specialitate sunt implementate în câteva scenarii de utilizare, rezultând o serie de servicii web. Impactul procesului de adaptare a limbajului medical a fost evaluat prin studii cu utilizatori, rezultatele confirmând utilitatea acestor metode şi a serviciilor asociate.

În plus, mijloace de adaptare a modului de prezentare a textului (aspect) sunt explorate. Atât instrumente pentru adaptarea aspectului textului în general, cât şi instrumente dedicate unor anumite tipuri de utilizatorilor (cum ar fi utilizatori cu disexie sau vedere slabă) sunt prezentate.Toate instrumentele dezvoltate în această lucrare au fost proiectate acordânduse atenţie deosebită unor factori precum accessibilitate, utilizabilitate şi disponibilitate, ele fiind concepute să ruleze oriunde în mediul online, cu efort minimal.

Teza acestei lucrări (ideea de bază):

Adaptarea textelor de specialitate prin identificarea şi explicarea terminologiei (întâlnită atât în formă canonică cât şi în formă derivată), împreună cu punerea la dispoziţie a unor mijloace de a schimba aspectul textului, pot duce la o accesibilitate şi înţelegere mai bună a mesajului pentru utilizatorii de rând.

1 Introduction

Access to information is crucial in our times. It should be provided for all persons, regardless their abilities or disabilities. While textual information distributed in non-digital forms was prone to multiple access limitations, the web has the potential to overpass many of those limitations. Tim-Berns Lee, the inventor of the Internet, said that "*The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect*." Also, the web accessibility section from World Wide Web Consortium (W3C) [1] mentions that the web should work for everybody, "*whatever their hardware, software, language, culture, location, or physical or mental ability*."

However due to the use of advanced or domain-specific languages, inadequate design of web pages or bad design of text presentation, information on the web often has access limitations for certain user categories.

This thesis explores existing tools and presents some novel techniques and tools that allow users to overcome such limitations of existing web pages by offering the capability to adapt the text from the original web page. The adaptation can be done on multiple levels, including language level or presentation and interaction level.

Language level

The language level is the one that is mostly studied in this thesis by looking at aspects like the presence of advanced terminology in domain-specific languages. A common example is the use of medical language that can often be difficult to understand by lay users. This example of adapting medical language is specifically important due to the impact that accurately consuming health information can have for people, empowering them in their own health care. A study on health literacy called "*The European Health Literacy Survey*" [2] that was done in 8 European member states between 2009 and 2012 revealed that in average almost half of the population was lacking the necessary (recommended) level of health literacy.

The adaptation of the language considers identifying specialized terminology in the target text and labeling it with the corresponding explanation.

A study done by the author and presented in this thesis indicates that approximately half of the terminology occurring in medical language (English & Romanian were tested) is not in the dictionary (basic) form. Because of this, much importance was given to accurately recognizing terminology that is the derivate form by the use of fuzzy matching.

While the use of fuzzy matching has highly increased the percentage of recognized terminology it has also opened the door for errors, drastically increasing the false positive rates. In order to cope with this the author explored several fuzzy matching validation solutions, both based on better techniques and on human validation.

The techniques used to increase the precision of fuzzy matching were 1) training the system with text from outside of its scope, and using the resulting matches to create an incorrect matching dictionary and 2) by analyzing the differences at character level between fuzzy matches and creating a matching model that was used to tune the *FuzzyHashMap*.

Human based validation section presents studies about benefits of incorporating human input into the system from sources like 1) the users of the application and 2) workers from crowdsourcing platforms.

Several use cases and tools developed for those use case are then presented. Use cases like explaining medical terminology for increasing message understanding and eventually empowering patients are presented and evaluated.

Domain-specific language and foreign languages

When specialized information is coupled with foreign languages this limitation becomes even bigger. In order to overcome this, the current work explores of annotating terminology in tag sets dedicated to internationalization (ITS).

Presentation level

Another level of adaptation is the presentation and interaction layer. Here aspects like the look of the text (font, color, size...) and the layout are taken into consideration. Ways of adapting text look in order to accommodate the needs of several user categories, like users with dyslexia or users with low vision are explored. According to World Health Organization it is estimated that there are 246 million people with low vision in the world [47]; also, according to Dyslexia International organization, it is estimated that between 5% and 15% of the population has dyslexia [38]. All these numbers indicate the fact that such print disabilities issues are a wide spread problem, and there is a big need for methods and tools designed to overpass such limitations.

The author worked on adapting text at presentation level in collaboration with other researchers (specialized in improving information accessibility for user categories like persons with dyslexia) in order to develop services that provide meaningful text adaptation and customization. The resulting tools can be integrated with the service for language adaptation, in order to facilitate text adaptation at multiple levels for better accessibility.

1.1 Objectives

This thesis pursues several objectives, the major ones being:

- Help **increase the understanding of domain-specific language** for lay persons. This will be done by identifying and explaining terminology in the target text. While the objective is to target a high rate of terminology recognition by the use of error tolerant fuzzy matching techniques, much attention has to be given to the precision of the adaptation in order to

maintain a safe (not misleading) adapted message. There are several additional objectives are derived from this one:
    - To contribute on improving the translation of domain-specific language
    - To design and develop some easy & ready to use data structures that will allow efficient fuzzy search
- **Improve accessibility of** domain-specific **text** and of text in general **by providing means to adapt the look** of text to make it fit the needs of the user
- A general objective of this research is to follow a design model that will lead to accessible, usable and highly available tools.
  The resulting tools should be included in a universal text access concept, which will emphasize the interaction between the tools and the parts that are still missing from the big picture of text access
- The applicative part of this research should be released publically for real world usage and evaluated with real users in order to validate its usefulness and efficiency.

1.2 Motivating example

Consider a student in the library of his school, trying to read a medical article about the symptoms of a certain disease. In order to accomplish his scope he encounters several accessibility limitations:

- He is having problems understanding the message due to the high number of advanced medical terminology
- He is lacking tools that can adapt the language and explain terminology
- If it happens that the article is written in an foreign language, than the limitation becomes bigger

Let's go even further and consider that our student has print disabilities (let's say dyslexia). Now we can add other limitations:

- He cannot read well the text due to the design (bad contrast, small text and difficult font) of the web page
- He cannot install any browser extensions or client applications that would enable him to adapt the look of the web page

One can see there are several levels of text access limitations occurring in this case. One is the limitation on language layer, represented by the difficulty of understanding advanced medical terminology. If the text is written in a foreign language the problem becomes even bigger.

Another limitation is on presentation layer, caused by the difficulty of accessing the text given the existing look.

The last issue is related to the availability of tools specialized in text adaptation, and the lack of alternatives when you are not browsing on your personal computer, like in this case.

This thesis proposes several methods and associated tools that will enable the student to adapt the text at language level (by explaining terminology and optionally improving translation) and at presentation level (by allowing the student to customize the look of the web page). The implementation of the proposed methods and techniques is designed to work directly into the browser, needing no installation or special rights, thus making them highly available on any computer or device, and the ideal alternative on shared computers, like the one from the library.

Services design

When designing the techniques and developing the associated tools presented in this thesis, much importance has been given to the availability and usability.

Availability: In order to make the tool highly available, they were design to work completely online, directly into the browser, making them platform independent, requiring no installation or special rights. This makes these tools easy to use on any platform, whether it is the personal computer of the user or especially when they are using devices from public spaces (like schools, libraries…).

Usability aspects have been given a high importance. Making the user interface of the tools user friendly, or offering alternatives for accessing and controlling the tools were priorities. The resulting tools can be manipulated directly from the URL, using the URL as a user input.

Most of the tools developed by the author and presented in this research are currently available online on a dedicated domain, under the name of *text4all*, at [106]. However, the part responsible with human based validation of approximate matching, although evaluated and showing good results, needs more work in order to be public ready, so it is not present in the current public version.

The available tools can be classified as language level tool, presentation level tools or combined:

a. Language level tools:
   - *text4all terminology interpreter:* takes the address of a web page, or raw text as input and adapts the text, presenting the terminology explained
   - *text4all term analysis*: takes the address of a web page as input and returns statistics of terminology usage in the target web page
   - *text4all ITStagger*: takes as input raw text or HTML and annotates the terminology using Internationalization Tag Set in order to improve translation of specialized language

b. Presentation level or combined levels tools
   - *text4all Customizer*: a service taking as input the address of a web page and enables the user to adjust the look of the web page; mostly useful for users with low vision or other print disabilities
   - *text4all dysWebxia*: a service designed for users with dyslexia, that takes a web page address as input and adapts the look and aspects of the language in order to make the web page more dyslexia friendly

A universal accessibility model for text is proposed at the end. Limitations and techniques and tools to overcome them are distributed per level. Benefits for combining the adaptations on multiple levels are also presented.

It is well known that "accessible design is good design". The tools designed to help users with disabilities are often better for users without disabilities too. So the author hopes that the presented techniques and tools will benefit not only the specific target groups, but also readers in general on the web.

1.3 Thesis outline

In the followings the author presents the outline of the thesis and a short description of each chapter:

1. **Introduction**
   Here the author presents the motivation for addressing text accessibility. The levels of the access limitations are presented, focusing on the ones addressed in this thesis (specialized language and presentation level). The objectives of this thesis are presented and a brief presentation of the techniques and tools developed in this research is done.

2. **Related work**
   Existing work from both specialized language level and presentation level is presented. Also the existing work is localized in a model of universal text accessibility presented in this work. However, specific related work is presented in more details in each chapter before presenting the proposed work.

Understanding of domain-specific language

Proposed techniques

3. **Specialized Language – language study & fuzzy matching**
   A study of Romanian and English medical language is done in order to determine the rate of terminology in canonical form compared to the one in derivate form. The study confirms the need of using fuzzy matching in order to identify terminology in derivate form. *FuzzyHashMap* data structure is presented as a solution for accurately and rapidly identify derivate terminology. Several tests and other use cases of the novel data structure are presented.

4. **Specialized Language – increasing fuzzy matching precision**
   The drawback of using fuzzy string matching is that, being error tolerant, it opens the door for incorrect matches (false positives). This chapter presents several methods for increasing fuzzy matching precision by the use automated and human based revision methods. The automated methods consist of the use of a) incorrect matching repository and b) creating a matching model for the specialized language and deriving and using a custom hashing pattern for the *FuzzyHashMap*. The human based revision is based on a) the users of the application and b) workers from crowdsourcing platforms.

Use cases and applications of proposed techniques

5. Specialized language accessibility - use cases and applications
This chapter presents several use cases and applications for identifying and adapting terminology in specialized text. Tools like terminology interpreter (used for explaining terminology in existing web pages in order to support patient empowerment), ITS term tagger (used for annotating terminology using ITS2.0 standard in order to improve localization and translation) and other tools and cases are presented. The impact of using *terminology interpreter* for patient empowerment is analyzed with several user studies, having encouraging results.

Accessibility of text at presentation level

6. Text Accessibility - Presentation level
The limitations for accessing text due to text presentation aspects like the look of the text, including text size, colors, fonts and others are explored. Tools allowing customization of text from existing web pages are presented, with several predefined templates for different user categories, like users with low vision. A tool developed by the author in collaboration with the author of DysWebxia model, designed to enhance readability for persons with dyslexia is presented.

7. Tools design considerations
Several aspects related to the design of the tools are presented. First the location and interactions of the resulted tools into the universal text accessibility concept is presented. Benefits of implementing the entire model of universal text accessibility are explored. Modalities of interacting with the tools are listed and special attention is given to interacting with the tools via the URLs is given.

8. Conclusions
A short review of the things presented in this thesis is presented. The main contributions of this work are listed. The resulted tools are shortly revised, focusing on their contribution.

2 Related work

This section explores and summarizes related work in the area of text accessibility tools, focusing on text in the web. Firs the author looks over the existing standards related to text accessibility and work related to standards compliance. Then work related to text adaptation at language level is explored, with special attention to medical language. Related work on text adaptation at presentation level is then summarized. Similar accessibility tools exploiting high availability and usability are presented. In the end the author presents the differences between the presented techniques and tools and the ones proposed by the author.

2.1 Text accessibility standards

There are several standards for information accessibility on the web, most popular being Web Content Accessibility Guidelines (WCAG) [3]. WCAG covers many aspects of text accessibility, the ones of interest for this work being the one related to recommended properties of the look of the text and the ones related to difficult language or difficult words (parts of Principle 1. <*Perceivable*> and Principle 3. <*Understandable*>). Apart from being the most known web accessibility standard, in the last years more governmental agencies have given laws imposing the compliance to WCAG standard for all public (state) web sites. For instance Australia has a campaign called "Web Accessibility *National Transition* Strategy" [4] imposing all governmental agencies web sites to follow standardized accessibility guidelines. This received support from the research and academia, like Vivienne [5] work on assisting this. Also the European Commission proposed Action 64, the Commission's *eGovernment* Action Plan 2011-2015 that calls for the development of services designed around user needs and ensuring inclusiveness and accessibility.

The techniques proposed in this research try to adapt existing web pages and to improve the compliance to WCAG standards in areas like color contrast, difficult words and others. Another standard (although it's not defined as a standard) is the Internationalization Tag Set (ITS) [6], used for improving translation and localization of web content.

2.2 Language level adaptation tools

Existing work on language level adaptation relate to both limitation due to foreign languages and limitation due to specialized languages. Relating to foreign language limitations there is a lot of research and production ready tools implementing various types of machine translation types. Most popular statistical machine translation tools are Google Translate [7], Microsoft Translator [8] and translation tools from SDL [9]. Other research projects like Duolingo [10] are doing translation based on *gamification* (by exploiting the answers from users of a specially designed game).

Translation & specialized language: This research does not focus on foreign language translation, it only integrates a similar service (Google Translate), and tries to improve translation for specialized language by tagging terminology in a dedicated annotation standard called International Tag Set (ITS).

There are only few services that are annotating terminology in text using ITS in order to improve translation, such a service being the one developed by Tilde, called TAWS [11] which can perform terminology annotation. The difference between TAWS and the service proposed in this work is that the second one supports advanced fuzzy matching for terminology, while the first exploits a statistical approach.

Specialized language adaptation: An early tool dedicated to adapting specialized language in the medical domain, facilitating medical language translation and mediating between doctors and patients is the MedSLT project presented in [12] and [13]. MedSTL is multilingual spoken language translation system tailored for medical domains, the multilingual aspect being presented in [14] and [15]. The system is designed to help in situations where no common language between the diagnosing doctor and the patient exists. The project considers narrow parts of the medical language, and uses controlled terminology dictionaries to perform the translation. In the latest versions a mobile version of the project has been developed [16], enhancing the usability and availability of the tool.

Classic machine translation tools can be used for translating medical content too, but these tools are very dependent on the training data. The text resulting from this process was evaluated in a research to be mostly incomprehensible [17].

Several dedicated tools addressed the issue of specialized language access. One research [18] proposed a framework to inform the design of an "interpretive layer" to "mediate" between lay (illness model) and professional (disease model) perspectives. A similar solution is a NLP project, simplifying medical text by replacing difficult terms with synonyms and/or reducing sentence size, having god results in terms of readability increase [19].

Another tool is identifying and explaining terminology from reports of electronic health records [20].

Consumer Health Vocabularies (CHVs) are a popular solution for increasing understanding to medical information for lay users by providing consumer/lay-friendly alternatives to the advanced medical terminology. Such approaches are presented by Zeng in [21] and such a vocabulary is available at [22].

This research proposes the adaptation of specialized language by recognition and labeling of advanced terminology. There are several differences between the related work and the one presented in this work. Main differences are the use of fuzzy matching for recognizing terminology and the increase availability of the resulted service by implementing it as a web mediator, working directly in the browser. This fills few gaps from the related work, by recognizing terminology even if present in a derivate mode compared to the term present in the dictionary and by making the service highly available and usable. Other benefits and also limitations are presented in the sections dedicated to language adaptation in this work.

2.3 Presentation level adaptation tool

There are an impressive number of tools and techniques for adapting the look of the text, most of them being related to text customization. They can be differentiated by the place where they are situated and operate on the text. In order to achieve text customization, users have a wide range of technical choices. Considering the location where the adaptation is done, it can be classified on three categories:

- **Client side**: This can be done by using custom browser options and settings, custom style sheets (formally called User Styled Sheets or USS) set in the browser, or dedicated browser extensions; most major browsers have support for adding USS.
- **Server side**: web content authors can add customization capabilities directly into the web document (most common is text size adjusting or color schemes)
- **Mediators**: There are several mediator projects (also called *transcoding* projects) that can adapt the look of existing web documents. Their main advantage is that they usually need no installation, and are platform independent.

Figure 2.1 illustrates this distribution of tools and services for text adaptation based on their location.

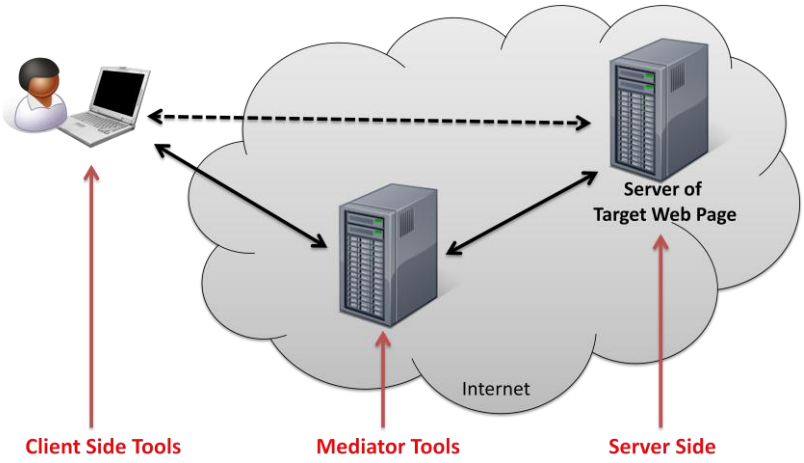

Figure 2.1 Overview of text adaptation type based on the location

Web mediator tools & web page *transcoding*

This thesis focuses on crating mediator tools, because they have the advantage of high availability. There are several web mediators, also called *transcoding* tools, which are designed to improve accessibility of existing web pages. Some enable text customization, other do text transformation, like text to speech or changing the look of the text. The tool considered as one of the first *transcoding* service is the proxy

presented in [27], designed for enhancing accessibility for blind users. Other examples of *transcoding* tools are: "Access Proxy", a tool designed by Brown and Robinson [23] for low vision users; BETSIE tool developed by Myers [24] from BBC; Web Accessibility Service [25] developed by Hanson and Richards (2003) from IBM; WebAnywhere [26], online text magnifier and screen reader developed by Bigham, Prince and Ladner (2008).

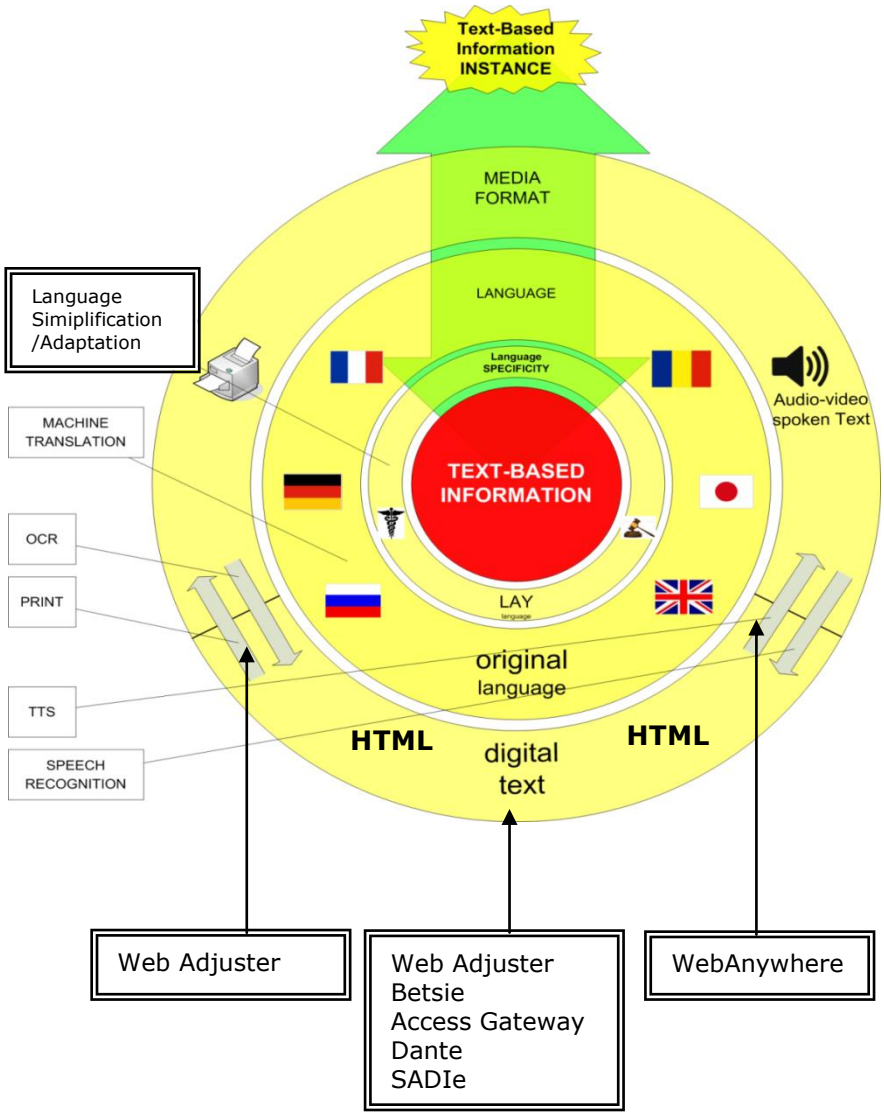

Figure 2.2. Technologies and tools distributed on access layers

A project performing semantic annotation of web pages in order to improve the experience of blind users, called Dante, is presented in [116]. The project supports multiple annotation types and relies on the concept of traveling within a web site

similar to the way blind people travel in the real world, having travel objects as references, being based on a travel framework [117]. A similar project, using a hybrid approach, combining semantic annotation with rule-based annotation is *SADIe* [118]. This *transcoding* project is also designed to help visually impaired users better browse the web. Both projects (*Dante* and *SADIe*) were done in the Web Ergonomic lab from University of Manchester.

Existing projects distributed per access limitations layers of text are presented in figure 2.2. Most of the presented mediator instruments target web site *transcoding* for visually impaired users.

The *transcoding* instruments resulting from this research address the issue of improving message understanding by explaining and semantically enriching original web pages (by annotating terminology in specific standards). Also, *transcoding* instruments that adjust web pages at presentation level are designed for users with dyslexia or low vision.

The tools presented here, and other tools related to this work are further taken in discussion and compared to the tools developed by the author throughout the thesis, in the sections where the proposed techniques and tools are presented.

3 Specialized Language Level – Language Study & Fuzzy Matching

3.1 Introduction

This research gives special attention to the difficulty of accessing and understanding specialized languages. The main problem with accessing specialized languages is the presence of advanced terminology, which is often hard to use and understand by lay readers. The author explores ways of identifying terminology in both canonical and derived form, annotate and attach the explanations. Special attention is given to reducing the false negative rate (*recall*) when identifying terminology that is not in the canonical form, by using fuzzy string matching. Before exploring the work done by the author, few notions like *canonical form* of a word, *derived form*, *fuzzy string matching* and *edit distance* have to be presented.

The *canonical form* is also called as standard form, dictionary form or citation form, and represents the basic form of a term, being usually the way it appears in dictionaries. It is also called as *lemma*, while the process of taking a word to its canonical form is called *lemmatization*. The *derived* or *inflected form* is the term present in a modified form due to the grammar construction of the context. Derived form is usually encountered in natural language.

Fuzzy string matching, also called approximate string matching, is the technique of finding strings that match a pattern approximately, rather than exactly. The closeness of a match is measured in terms of the number of primitive operations necessary to convert the string into an exact match. This number is called the *edit distance* between the string and the pattern.

First a study is presented, exploring the percentages of terminology found in canonical form compared to the one in derivate form in English and Romanian language, medical domain. The study results motivate the use of fuzzy matching techniques, in order to identify terms in derivate form.

A novel data structure designed to allow fast fuzzy string matching is presented. The data structure, named *FuzzyHashMap,* combines the performance of *HashMaps* (popular data structure from Java programming language, presented later in more details in this chapter) with the flexibility and tolerance of fuzzy string matching.

All methods presented here are designed to work on any languages and any domains, all language dependencies being related to external resources like the used dictionaries.

3.2 Study on incidence rate of canonical vs. derivate form of terminology in natural language

In this section the author studies the distribution of terminology appearing in canonical form compared to the one appearing in derivate form in natural language. The study is done in order to measure the need of using approximate matching techniques when identifying terminology. The study was done on Romanian and English language, looking for terminology within the medical sublanguage. *Sublanguage* is subunit of a language, representing the language specific to a domain. While Romanian represents a language, medical Romanian text represents a sublanguage of Romanian language. Throughout this work the notion of *sublanguage* will be used to express a domain specific language. Since the tools from this work, that are meant to identify and explain medical terminology, are designed to find and explain terminology within a web page or raw text, this study was done on medical language from several dedicated web pages.

In order to perform the study a dedicated tool was designed. The tool, called *text4all term analysis*, takes as input the address of a web page and several settings like: language, domain and the maximum edit distance allowed when using fuzzy matching. The tools uses behind the scenes dedicated dictionaries and advanced fuzzy matching techniques, but they are not going to be detailed here, since this section focuses on the terminology study, the techniques behind the tool being detailed in a dedicated section in chapter 5. However, some information about the tool is provided, in order to help the readers understand the study process. Figure 3.1 presents an overview of the tool; here one can see the first section handling input data (in the left), the internal structure and the results section (in the right), listing details about the number of terms encountered and their distribution between exact and fuzzy match. The results section also lists all the matching pair and their frequencies.

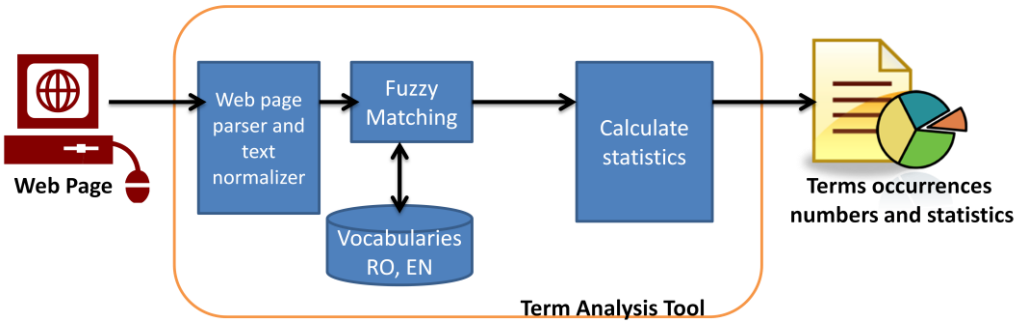

Figure 3.1 Overview of text4all term analysis tool

The analysis process starts by entering the URL of the target web page and selecting other input settings (language and domain). Also the maximum edit distance for fuzzy matching is set in this phase, which for all these studies was selected to be 2. Then the analysis process is started, and the results area is presented, containing

information about numbers of words and terms, and listing all exact and fuzzy matches together with their frequency of appearance.

The results list was then revised by the author in order to filter out incorrect matches, and correct the terms distribution number, in order to have a real image of derived terminology incidence. An example of such a revised list of results is presented in **Appendix A.**

3.2.1 Study on Romanian medical language

Here the text from several medical web pages or portals in Romanian language was considered for the study. Articles from web sites presenting 1) interviews with doctors (medlive.hotnews.ro), 2) information for pregnancy (www.infomaterna.ro) and 3) general presentation of several diseases (www.sfatulmedicului.ro) were studied. Nine web pages were studied, counting all together more than 98000 words. Table 3.1 presents details of the studied text, including total number of words and terminology occurrence in both canonical and derivate form.

Web Page	Total number of words	Unique **Canonical** terms occurrence	Repeating **Canonical** terms occurrence	Unique **Fuzzy** terms occurrence	Repeating **Fuzzy** terms occurrence
http://medlive.hotnews.ro/tumorile-suprafetei-oculare-dr-florentina-chitac-pashalidi-medic-primar-oftalmolog-discuta-online-cu-cititorii-miercuri-de-la-11-00.html	11444	31	71	38	63
http://medlive.hotnews.ro/video-interviu-foto-dr-victor-radu-medic-primar-chirurgie-generala-a-realizat-o-interventie-laparoscopica-in-premiera-la-un-pacient-cu-hernie-inghinala-avantajul-este-agresiunea-mai-mica-asupra-p.html	8706	26	78	17	30
http://medlive.hotnews.ro/cum-se-trateaza-corect-varicele-dr-halpern-rafael-medic-primar-chirurgie-cardiovasculara-discuta-online-cu-cititorii-joi-de-la-ora-12-00.html	11694	30	74	24	46
http://medlive.hotnews.ro/studiu-40-din-totalul-lombalgiilor-cronice-cu-hernie-de-disc-sunt-cauzate-de-un-germen.html	7926	26	64	17	28
http://www.infomaterna.ro/Disfunctiile-tiroidiene-in-sarcin-227/182/articol.html	10881	16	44	39	115
http://www.infomaterna.ro/Epiziotomia---rutina-sau-necesitate/137/articol.html	6594	4	6	9	11
http://www.sfatulmedicului.ro/cancer	15250	16	30	33	48
http://www.sfatulmedicului.ro/depresia	13243	7	19	22	32
http://www.sfatulmedicului.ro/hepatita	12618	9	34	26	45
TOTAL	**98356**	**165**	**420**	**225**	**418**

Table 3.1 Distribution of medical terminology in canonical and fuzzy form in Romanian medical web pages

Next the author analyses and compares the distribution of canonical and derived terminology in percentages.

Web Page	**% of canonical terms**	**% of fuzzy terms**
http://medlive.hotnews.ro/tumorile-suprafetei-oculare-dr-florentina-chitac-pashalidi-medic-primar-oftalmolog-discuta-online-cu-cititorii-miercuri-de-la-11-00.html	52.9	47.1
http://medlive.hotnews.ro/video-interviu-foto-dr-victor-radu-medic-primar-chirurgie-generala-a-realizat-o-interventie-laparoscopica-in-premiera-la-un-pacient-cu-hernie-inghinala-avantajul-este-agresiunea-mai-mica-asupra-p.html	72.2	27.8
http://medlive.hotnews.ro/cum-se-trateaza-corect-varicele-dr-halpern-rafael-medic-primar-chirurgie-cardiovasculara-discuta-online-cu-cititorii-joi-de-la-ora-12-00.html	61.6	38.4
http://medlive.hotnews.ro/studiu-40-din-totalul-lombalgiilor-cronice-cu-hernie-de-disc-sunt-cauzate-de-un-germen.html	69.5	30.5
http://www.infomaterna.ro/Disfunctiile-tiroidiene-in-sarcin-227/182/articol.html	27.6	72.4
http://www.infomaterna.ro/Epiziotomia---rutina-sau-necesitate/137/articol.html	38.4	61.6
http://www.sfatulmedicului.ro/cancer	37.2	62.8
http://www.sfatulmedicului.ro/depresia	43	57
http://www.sfatulmedicului.ro/hepatita	35.29	64.71
AVERAGE	**48.6%**	**51.4%**

Table 3.2 Distribution of percentages of canonical vs. fuzzy terms, considering all terms occurrences.

From table 3.2 one can see that the rate of canonical terms is a little smaller than the rate of term in derivate form (fuzzy matches), 49% of the terms being exact matches, while 51% fuzzy matches.

From table 3.3 one can see that the rate of canonical terms is significantly smaller than the rate of term in derivate form (fuzzy matches), 40% of the terms being exact matches, while almost 60% fuzzy matches.

Figure 3.2 summarizes all these data, and shows a big image over the distribution of terminology in Romanian medical language in both canonical form (Exact match) and derivate/inflected form (Fuzzy match).

Web Page	% of canonical terms	% of fuzzy terms
http://medlive.hotnews.ro/tumorile-suprafetei-oculare-dr-florentina-chitac-pashalidi-medic-primar-oftalmolog-discuta-online-cu-cititorii-miercuri-de-la-11-00.html	44.9%	55.1%
http://medlive.hotnews.ro/video-interviu-foto-dr-victor-radu-medic-primar-chirurgie-generala-a-realizat-o-interventie-laparoscopica-in-premiera-la-un-pacient-cu-hernie-inghinala-avantajul-este-agresiunea-mai-mica-asupra-p.html	60.5%	39.5%
http://medlive.hotnews.ro/cum-se-trateaza-corect-varicele-dr-halpern-rafael-medic-primar-chirurgie-cardiovasculara-discuta-online-cu-cititorii-joi-de-la-ora-12-00.html	55.6%	44.4%
http://medlive.hotnews.ro/studiu-40-din-totalul-lombalgiilor-cronice-cu-hernie-de-disc-sunt-cauzate-de-un-germen.html	60.5%	39.5%
http://www.infomaterna.ro/Disfunctiile-tiroidiene-in-sarcin-227/182/articol.html	29.1%	70.9%
http://www.infomaterna.ro/Epiziotomia---rutina-sau-necesitate/137/articol.html	30.8%	69.2%
http://www.sfatulmedicului.ro/cancer	32.7%	67.3%
http://www.sfatulmedicului.ro/depresia	24.1%	75.9%
http://www.sfatulmedicului.ro/hepatita	25.7%	74.3%
AVERAGE	**40.4%**	**59.6%**

Table 3.3 Distribution of percentages of canonical vs. fuzzy terms, considering first (one) occurrence of each term.

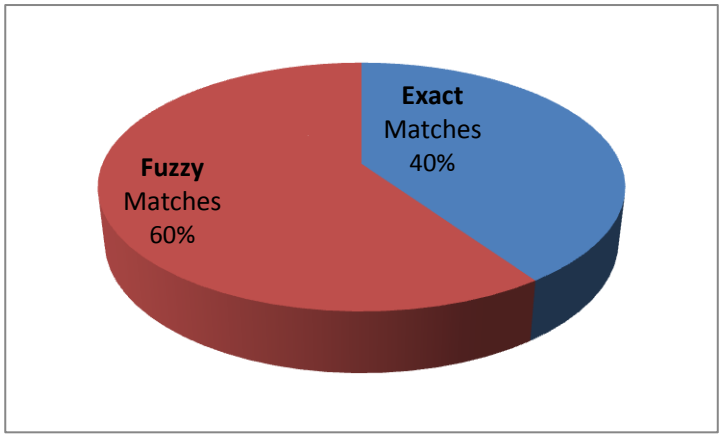

Figure 3.2 Fuzzy Matches (derivate term) vs. Exact matches (canonical term) for Medical Romanian Language

This study reveals that the incidence of fuzzy matches of medical terminology in Romanian medical language from websites is bigger than the incidence of exact matches. Given the fact that medical language can vary in complexity depending on the nature and subject of the medical text, the author does not claim that this distribution rate of canonical/derived terms can be generally applied in medical Romanian language; a more extended study should be realized in order to validate this assumption. However this study confirms the need of using fuzzy matching when identifying medical terminology on medical texts from web pages dedicated for end users (lay persons).

3.2.2 Study on English medical language

In this study the author analyzed again 9 web pages containing medical content in English. Most of web pages contained informative messages on different diseases, symptoms or treatments being taken from web sites or portals like netdoctor.co.uk, medicinenet.com and webmed.com. Table 3.4 presents the details of the data analyzed for English medical language. In table 3.4, unique occurrences represent the number of different terms appearing at least once in the text. The repeating occurrences represent all occurrences of medical terms, including multiple occurrences of the same term. From table 3.4 one can see that the number of terms in derivate terms is approximately half of the number of canonical terms.

Web Page	Total number of words	**Canonic** terms unique occurrence	**Canonic** terms repeating occurrence	**Fuzzy** terms unique occurrence	**Fuzzy** terms repeating occurrence
http://www.netdoctor.co.uk/diseases/facts/allergyfood.htm	9680	22	57	7	61
http://www.netdoctor.co.uk/diseases/facts/asthma.htm	11225	27	112	9	25
http://www.netdoctor.co.uk/diseases/facts/lungcancer.htm	9367	17	39	9	16
http://www.medicinenet.com/script/main/art.asp?articlekey=174117	11376	13	17	7	10
http://www.medicinenet.com/liver_disease/article.htm	16846	23	46	9	23
http://www.medicinenet.com/norovirus_infection/page2.htm	14581	16	25	8	28
http://www.webmd.com/hypertension-high-blood-pressure/guide/understanding-high-blood-pressure-basics	40340	25	175	11	44
http://www.webmd.com/skin-problems-and-treatments/psoriasis/news/20130913/treatment-options-expand-for-psoriasis-patients	27379	11	19	11	28
http://www.webmd.com/add-adhd/childhood-adhd/understanding-adhd-treatment	28629	16	23	6	22
TOTAL	**169423**	**170**	**513**	**77**	**257**

Table 3.4 Occurrences of canonical and fuzzy medical terminology in English text from medical web pages

In order to get a better image over the rate of canonical vs. derivate terms, table 3.5 and 3.6 illustrate the percentage of canonical term vs. derivate terms from the total number of terms identified.

Web Page	**Canonical terms**	**Fuzzy terms**
http://www.netdoctor.co.uk/diseases/facts/allergyfood.htm	75.9	24.1
http://www.netdoctor.co.uk/diseases/facts/asthma.htm	75.0	25.0
http://www.netdoctor.co.uk/diseases/facts/lungcancer.htm	65.4	34.6
http://www.medicinenet.com/script/main/art.asp?articlekey=174117	65.0	35.0
http://www.medicinenet.com/liver_disease/article.htm	71.9	28.1
http://www.medicinenet.com/norovirus_infection/page2.htm	66.7	33.3
http://www.webmd.com/hypertension-high-blood-pressure/guide/understanding-high-blood-pressure-basics	69.4	30.6
http://www.webmd.com/skin-problems-and-treatments/psoriasis/news/20130913/treatment-options-expand-for-psoriasis-patients	50.0	50.0
http://www.webmd.com/add-adhd/childhood-adhd/understanding-adhd-treatment	72.7	27.3
AVERAGE	**68.0%**	**32.0%**

Table 3.5 Distribution in % of unique occurrences of canonical and fuzzy terminology

From table Y2 one can see that the rate of canonical terms is bigger than the rate of term in derivate form (fuzzy matches), 68% of the terms being exact matches, while 32% fuzzy matches.

Web Page	**Canonical terms**	**Fuzzy terms**
http://www.netdoctor.co.uk/diseases/facts/allergyfood.htm	48.3	51.7
http://www.netdoctor.co.uk/diseases/facts/asthma.htm	81.8	18.2
http://www.netdoctor.co.uk/diseases/facts/lungcancer.htm	70.9	29.1
http://www.medicinenet.com/script/main/art.asp?articlekey=174117	63.0	37.0
http://www.medicinenet.com/liver_disease/article.htm	66.7	33.3
http://www.medicinenet.com/norovirus_infection/page2.htm	47.2	52.8
http://www.webmd.com/hypertension-high-blood-pressure/guide/understanding-high-blood-pressure-basics	79.9	20.1
http://www.webmd.com/skin-problems-and-treatments/psoriasis/news/20130913/treatment-options-expand-for-psoriasis-patients	40.4	59.6
http://www.webmd.com/add-adhd/childhood-adhd/understanding-adhd-treatment	51.1	48.9
AVERAGE	**61.0%**	**39.0%**

Table 3.6 Distribution in % of repeating occurrences of canonical and fuzzy terminology

From table 3.6 one can see that the rate of canonical terms is significantly bigger than the rate of term in derivate form (fuzzy matches), 61% of the terms being exact matches, while almost 39% fuzzy matches.

Figure 3.3 illustrates the rate of canonical terminology (exact match) occurrence compared to the one of derived terminology (fuzzy match) occurrences.

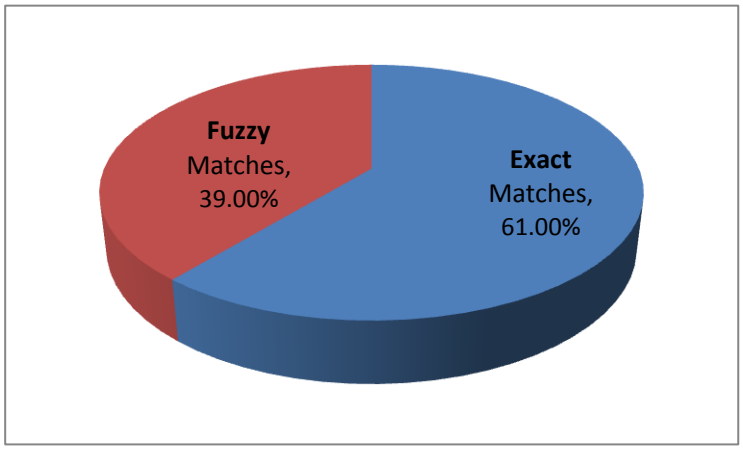

Figure 3.3 Fuzzy Matches (derivate term) vs. Exact matches (canonical term) for English Medical Language

From this study the author concluded that the incidence of fuzzy matches of medical terminology in English medical websites is much smaller than the incidence of exact matches. However, the number of fuzzy matches is big enough to motivate the need of using fuzzy matching when identifying medical terminology.

Conclusion

From the analysis of the form or medical terminology occurring in Romanian and English medical web pages several conclusions were driven:

- Given the high percentage of terminology appearing in derivate form, **the use of approximate matching recognition is recommended** (more for Romanian language, but also for English).
- The incidence of medical terminology appearing in canonical form compared to the one appearing in derivate form can be very different from language to language
- The incidence of medical terminology appearing in canonical form compared to the one appearing in derivate form is much bigger in Romanian language compared to the one in English language.
- In Romanian websites the incidence of medical terminology in canonical from is approximately the same with the rate of terminology appearing in derivate form.

These conclusions may bring useful information for future research, by providing a clear image of the form in which medical terminology appears in natural language. This information may be useful for standards makers in the field of terminology annotation and markup; it may be useful for content authoring tools, annotating tools, spell checkers and other terminology related tools.

3.3 Reducing false negative rate with fuzzy matching

Given the high incidence of terminology occurring in derivate form, as found in the study from the previous section, an important aspect was taken into consideration. By only identifying and explaining terminology occurring in canonical form, so by using exact matching only, approximately half of the terminology occurrence would be lost, approximate matches being false negatives. In order to reduce the false negatives rate (also called *recall*) this work uses fuzzy matching techniques. Because the techniques used here are implemented in tools adapting web pages, it was very important to design techniques that comply to a very low latency, so that the resulting tool would preserve a decent experience for the end user. For this the author created an extension to the high performance HashMap data structure, called FuzzyHashMap. This extension adds fuzzy search capabilities to regular HashMap data structure from Java language.

3.3.1 FuzzyHashMap

Hash maps are data structures widely used in modern programming languages like Java for their simplicity and efficiency. When fuzzy string search is needed (like in natural language processing) finding an approximate key match in a regular Java HashMap [28] is a non trivial task. It usually requires the brute force method of iterating trough the set of keys and use of string metrics methods. Although this approach works it is time consuming and loses the hashing advantage of the hash map. Another option is to use a different data structure like TreeMap, which is faster, but also have limitations on fuzzy string search. Here we present FuzzyHashMap [29], an extension to the regular Java HashMap data structure allowing highly efficient fuzzy string key search. Based on object oriented principles this extended hash map uses a custom key that enables different types of pre-hashing functions and different types of dynamic programming algorithms for approximate string matching. Customizable algorithms and settings bring flexibility to this new data structure, making it adaptable to each specific use case. This data structure is the ideal structure for data like natural language. Since its creation it has been also used in the field of bioinformatics, for genome data [30].

When dealing with linguistic text storing, for text collection in the form of dictionaries, the best choice for storing such data is the HashMap [28] data structure. They have the advantage of storing (in memory) a big amount of data and offering almost instant access to the value mapped on the searched key.

The problem with linguistic data, especially in natural language processing, is dealing with uncertain and uncontrolled information. A method of dealing with this kind of data is using error-tolerant methods like fuzzy string matching. There is a lot of research done in this area; one can even consider this a dedicated research field, with indexing subcategories as listed in [32]. A few examples of usage of this kind

of technique are name matching [33], spellchecking and "on-the-fly" type-ahead suggesting systems [34] and most important textual information search in string collections, using a big variety of methods, some described in [35]. In many cases when dealing with text, the back-end data storage solutions are databases. Fuzzy usage in database information manipulation is being worked on for a while, FSQL (FuzzySQL) or SQLf getting closer to standardization [36]. Not the same thing is true when coming to in memory fuzzy data structures. This paper focuses on in memory data structures, usually used in small or medium applications, where speed and fuzziness is needed. Similar concepts, ranging from suffix trees [37] that were implemented from a lot of time to more recent variable length gram indexing [39], that are related to this concept, could be implemented in OOP.

Below the initial version of the fuzzy data structure that was published in [29] is presented, followed by a presentation of a revised version and a comparison between initial and revised version. The current data structure presented here, and possible future releases are available as open source at the *FuzzyHashMap* project web address [40].

3.3.1.1 Methods

The main advantage of HashMaps is high speed (almost instant) when searching elements by key. This speed is given by the hash based search mechanism instead of iteration trough the set of stored data. Furthermore the implementation provides constant-time performance for the basic operations (get and put), assuming the hash function disperses the elements properly. Iteration over collection views requires time proportional to the size of the HashMap (the number of buckets contained).

There are two main methods for Java objects that have an important impact over the HashMap performance and functionality: *hashCode()* and *equals().* HashCode method is implementing the hash function that generates a numerical code corresponding to a certain key. The hash code will be used to identify the location where the entry will be stored in the map. A bad hash function can lead to bad performance due to a) high time consumption of hashing, or b) duplicate hash code for two different keys. The last case is called a *collision.* When a collision happens, *equals* method is used to compare the value already stored on a specific location, with the new one. If the keys are different, a new location, linked to the existing entry, will be allocated. The ideal Java HashMap has no collisions.

FuzzyHashMap (FHM) adds fuzzy enabled functionality to the default operations of standard Java HashMap. In Fig. 3.4 the UML Class Diagram of the FHM implementation is presented. One can see that FHM class extends the Java HashMap class and aggregates FuzzyKey class. FuzzyHashMap class has multiple constructors, to allow customization of algorithms used for hashing or string comparison.

The most important methods in FHM class are *getFuzzy* and *containsFuzzyKey*, allowing fuzzy query. *FuzzyKey* class is used to create fuzzy enabled keys to be used in FHM class. *FuzzyKey* class uses algorithms from the associated StringMetrics class to perform string metrics operations.

3.3.1.2 FuzzyHashMap Functionality

Further the internal functionality of the FHM is described. Contrarily to regular HashMap, in FuzzyHashMap collisions are being intentionally created and controlled (clustered), so that each collision will happen between words from the same "group". The "group" of the words is being dictated by the pre-hashing function, and is usually associated with syntactic similarity.

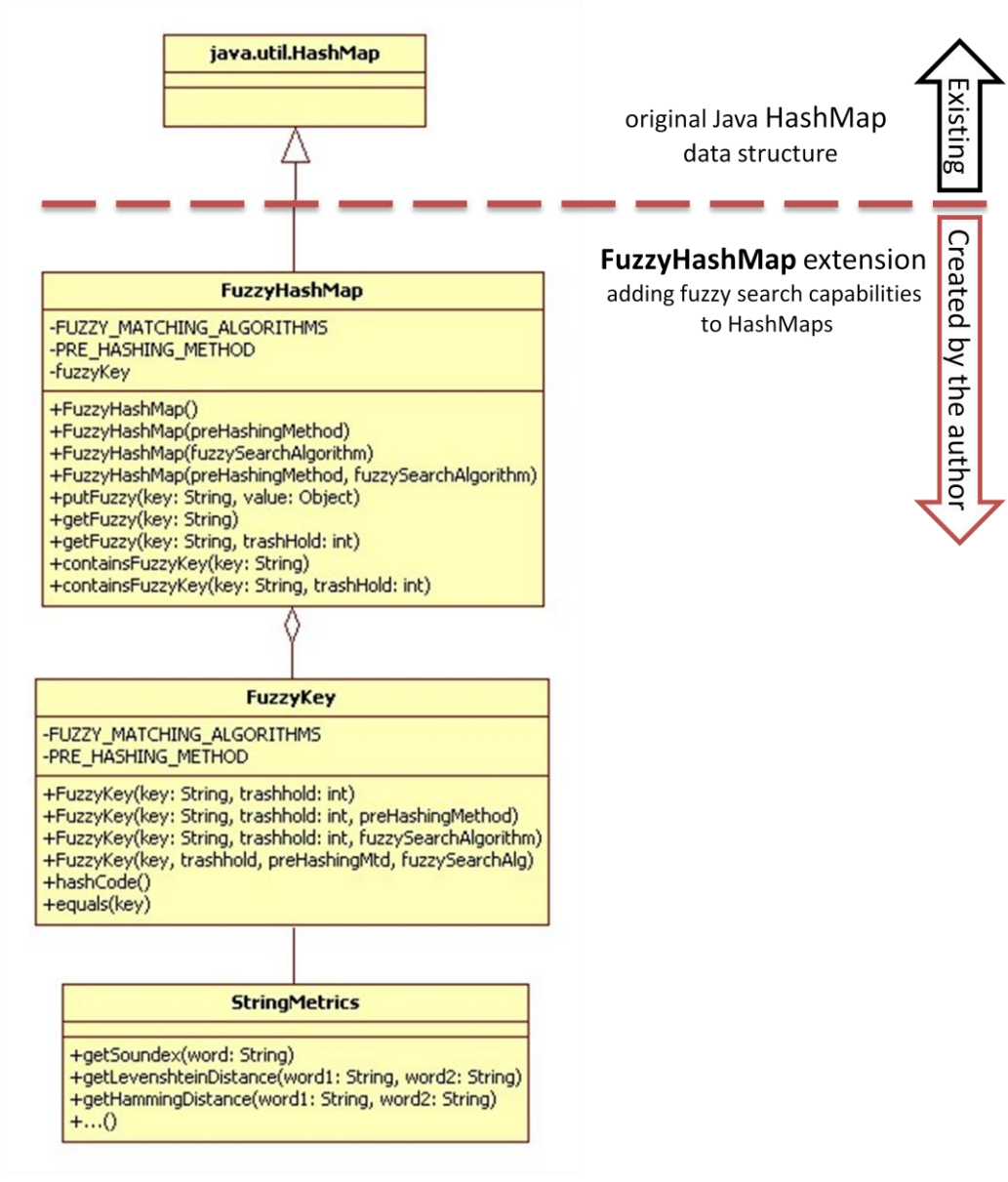

Figure 3.4 FuzzyHashMap UML Class Diagram and contribution delimitation

To achieve an almost native fuzzy search, FHM use fuzzy keys. So for each string key added or searched in the FHM, an instance of *FuzzyKey* class is created. *FuzzyKey* class has a string key as member, and most important, it overrides the default Object methods that are used in HashMap:

```
hashCode()
equals(Object o).
```

These methods (functions) are called when adding and when querying the data in the HashMap. Below, the implementation of the overridden *hashcode* and *equals* methods from *FuzzyKey* is being presented.

a. The first method called is ***hashCode()*.** The overriding *hashCode()* method performs a pre-hashing function that is reducing the given key, followed by a call on the default String *hashCode* method. Here is a pseudo code example of the overriding hashCode function:

```
{
  String preHashedKey = performPreHashing(key);
  int hashCode = preHashedKey.hashCode();
  return hashCode;
}
```

Several pre-hashing functions were implemented in the initial version. The most efficient ones are: *Substring*, *Soundex.*

When using *substring*(0, 4) as pre-hashing function, a (collision) group that contains words starting with the same four letters are created; here is an example of how this substring function works:

```
substring("Washington",0,4) = "Wash"
```

Using substring as pre-hashing function will make the FuzzyHashMap have a similar behavior to a TreeMap, but the inner structure is different, the entries are not being ordered and it allows more complex pre-hashing functions that might not be suitable for a tree or would have less performance.

The other type of pre-hashing function is *soundex* algorithm [41], which will group words by English phonetic similarity. Soundex algorithm associates an alphanumerical code to a given word, the resulting code having the fist letter equal to the word, and then three numbers calculated according to the letters in the word. An important note is that *soundex* resulting code has always the size four. Here is an example of soundex code:

```
soundex("Washington") = "W252"
```

Performing pre-hashing and calculating the hash code is done whenever an item is added or searched in the HashMap.

b. The method ***equals()*** is called in case of collisions, whenever a query operation is done and for adding (put) operations.

Put operation (adding entries): In case of adding operation, the *equals* method is used for collision handling. If the hash code value for the new item represents a

location already used, then the *equals* method is used to check whether the existing item is equal to the new one. If positive, than the new item will be ignored, else, the new item will be saved at a location linked to the existing item location. In FHM this situating will occur very often, due to the pre-hashing function. A very important note is that in FHM, for adding new entries the equals method will not use fuzziness. This is done to avoid losing similar elements when they are added in the hash map.

Get operation (searching entries): When searching items, after the hash code was calculated, the entry stored at the resulted hash code value is checked for similarity by using *equals* method. Notice that this time, equals method will allow fuzziness, in order to obtain approximate matches. The accepted similarity level can be controlled by a threshold value stored in the FuzzyKey object. If no similar match is found, then all items linked to the item corresponding to the hash code value are checked for similarity.

The default approximate matching algorithm is Levenshtein Distance [42]. This algorithm calculates the edit distance between two words. The edit distance represents the number of modifications (insert / update / delete) needed to make the words equal. The matching fuzziness is flexible and is controlled by making *equals* method use a custom maximum distance threshold value. By default this value is 2 and it was decided empirically after several tests. Another approximate matching algorithm used is Hamming [43] distance, which can compare equal length string only, and represents the number of positions where the corresponding symbols are different.

To conclude, in this section the author shows how fuzzy capabilities can be added to a HashMap data structure by relaxing the semantics of equality used in determining the uniqueness of a key. This is done by manipulating the two-step mechanism required to search a map with a key (*hashcode* and *equals*). Instead of having a hashing function that tries to avoid possible collisions, the fuzzy approach deliberately encourages controlled collisions in a hashMap, by allowing only part of the search key to be used by a hashing function and permitting a degree of variability when checking key equality. For all these the author extended HashMap with *FuzzyHashMap* class, created a *FuzzyKey* class that is a wrapper of the default String class and aggregated several String Metrics functioanlities.

3.3.1.3 Examples of FuzzyHashMape use cases

To illustrate the way FHM works and areas where it can be used two examples are presented.

Example 1. Justice Terminology Fuzzy Dictionary*:* We want to identify, in plain text, law specific terms. The terms have to be identified even though they are not in the canonical form. For this we will use a FHM to build a law specific terminology dictionary. We will use that terminology dictionary to recognize specialized law terms in the given text.

The first step is creating and populating the terminology dictionary. The FHM used will have the default settings. Fig. 3.5 presents the process of populating the FHM with law terminology data. For this case the *substring(0,4)* pre-hashing function was chosen. Let's consider adding term "*action*" with the associated interpretation into the FHM. The pre-hashing function returns "*acti*", and the hashing function has

calculated value 12 for this string. Since there is no other entry stored on this location, this entry is saved here.

Now let's jump to the "*violation*" dictionary entry. Notice that in this case the pre-hashing function returned "*viol*" the same as it returned for the "*violence*" entry. The hashing function will calculate the same value for both entries, *215*. In this case we have a collision, so the hash map will call *equals* method to check whether "*violation*" is already saved in the map. After performing this check, the entries are considered different, so a new location is being allocated for the new entry, which will be linked to the location of "*violence*" entry.

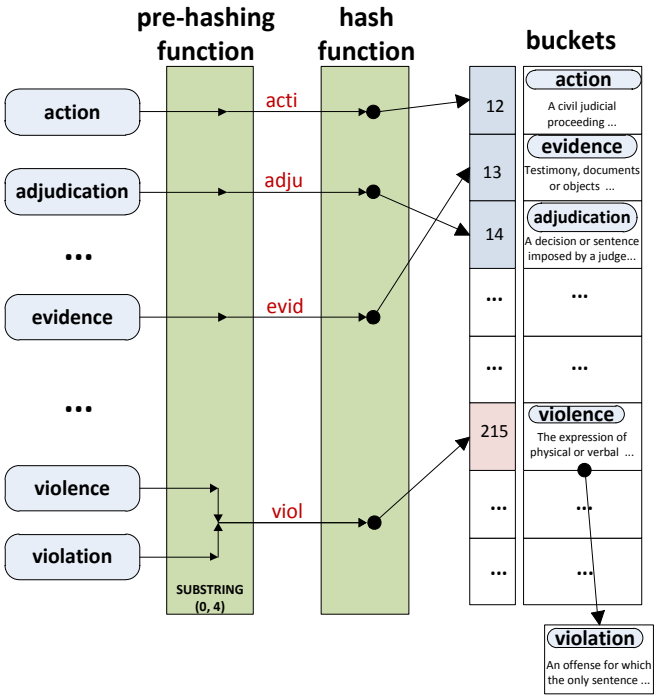

Figure 3.5 Populating FuzzyHashMap with Law Terminology data

Let's put the fuzzy dictionary to work now. So we consider we are parsing the following phrase:

"the *judge* has the option of either *adjudicating* you as guilty or.."

Each word is checked against the dictionary. When arriving to "*adjudicating*" term, as presented in Fig. 3.6, the dictionary will search by firstly pre-hashing the term. The hash code for the resulted string "*adju*" is computed, and it points to the location 14, where "*adjudication*" entry is stored. The *equals* method is called to check the term against the entry at position 14. *Equals* method now uses approximate matching. The Levenshtein distance (which is the default approximate matching algorithm in FHM) between "*adjudication*" and "*adjudicating*" is 2. While by default FHM has threshold value 2, the *equals* method will return *true.* So the

word *adjudicating* has been associated to the term *adjudication* from the dictionary.

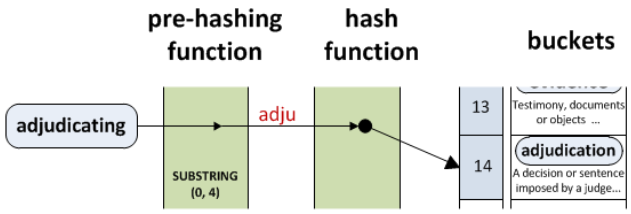

Figure 3.6 Fuzzy searching for word "*adjudicating*"

In conclusion, the FHM enables finding terms that are not in their canonical form, in a very efficient way. To make this possible, this is error tolerant, so it may do mistakes, but a good threshold and algorithm tuning improves the performance of the FHM.

Example 2. Phone book: The phone book is storing contacts (name of a person and the phone number). The example is very similar to the precedent one, juts that here the pre-hashing algorithm used is *soundex* algorithm. Manipulating data in the phone book is presented in Fig. 3.7.

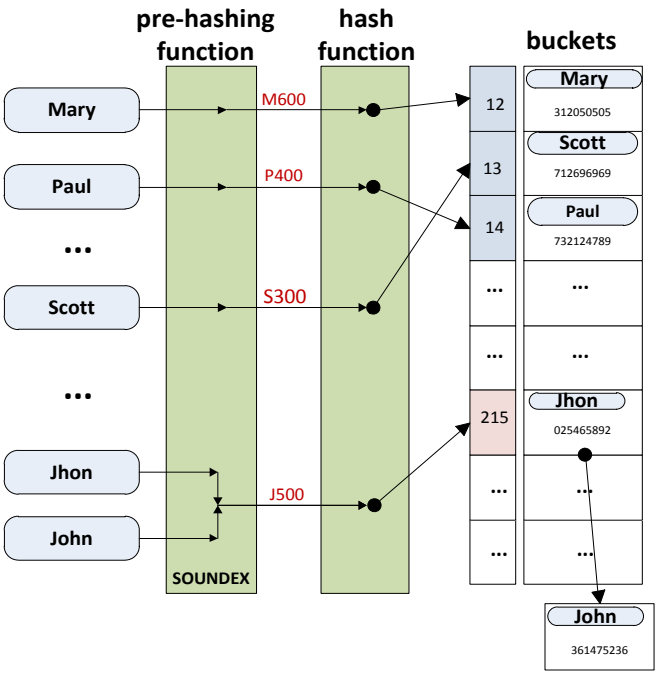

Figure 3.7 Fuzzy Phone Book

The advantage of using soundex algorithm is that it helps on having a better words clustering, according to the way they sound. So this fuzzy phone book could be used to enable users find contacts even if they are misspelling the name of a contact

person, or when using speech recognition, which is error prone, to select a contact person. The detailed procedure of adding and searching entries in the FHM will not be presented because it is similar to the previous example.

Other examples and recent real world usage: Additionally to the described examples, some existing use cases for FHM data structure are listed next:

- *Natural language processing*: Here the fuzzy words recognition can help in tasks like *lemmatization*. This use case is explored in this thesis in chapter 5 by looking at the terminology annotation tools developed by the author.

- Bioinformatics: At the time of writing this thesis, a customized version of the FHM has already been used in projects like genome anchoring [30], [31] by specialized bioinformatics researchers.

- Information Retrieval and Search Engines: a project using FHM for a blog search engine is presented at [113]

Also, the author indicates that the FHM could be used in areas like:

- Spelling correction

- *"On-the-fly" type-ahead search:* fuzzy HashMaps could also be an alternative to the current work done in this area for improving database query performance [44].

The fact that the *FuzzyHashMap* was already used in fields like natural language processing, bioinformatics and development of custom search engines, and the possibility of using it in other areas listed before makes this data structure an important (ready to use) instrument and proves its usefulness.

3.3.1.4 Speed and accuracy test of FHM

While the FHM was used in the research work described in this thesis, the need for several changes and options surfaced. This section presents tests that were done with the initial implementation of the FHM, at the time the FHM was developed. At the time of writing this thesis, the FHM has more options, and the default settings were changed, so these test results are not necessarily accurate for the current version, but for the original one. Extended accuracy tests were done with the updated FHM versions, and are presented in the next chapter.

Here, two main test types were done to measure the performance of the FHM, one for accuracy and another one for time consumption.

a. Accuracy test: Testing the accuracy of fuzzy matching is very dependent to the *text used in the test*, the *size of the map* and the *settings* of the FHM, so the results can change significantly form one case to another. For this type of test, the settings used for the FHM were:

1. Substring(0,4) pre-hashing function
2. Levenshtein fuzzy matching algorithm
3. Distance threshold value 2

The test strategy was to use the FHM as a medical terminology dictionary. The FHM medical dictionary was populated with 1030 English medical terms.

The author developed a prototype application, integrating the FHM, which takes text as input, normalizes it and tries to identify medical terms in the text by searching fuzzy matches in the FHM for each word in the text. Later this application was extended to a mature terminology annotation tool, which is being presented in chapter 5.

Using this prototype application a terminology recognition test was done on text from the *American Family Physicians Journal*. Here are some test results: for a text having a total number of 568 words, 43 words have been identified as medical terms. From these 43 words 9 were incorrect matches. This means an approximate 80% accuracy for the fuzzy matching. In another test done on text from an *eMedicine* web site, containing 2730 words, 260 were recognized, from which, 7 were incorrect matches. This means 97% accuracy.

As mentioned before, these values can change for different test conditions. In our case the biggest improvement can be done by populating the map with more medical terms. This issue of improving approximate matching precision is explored in details in chapter 4.

b. Speed test: The other type of test was done to analyze the speed of the FHM. The speed is compared to HashMap speed, where iteration (complete scan of the map iterator) was used when exact matches were not found. For these tests each map was populated with 30000 words (random strings). Than 1000 words were searched in the maps, in two tests:

a. In this test all the words identified as matches were approximate matches only. Here (Fig. 3.8) we see the performance of HashMap is bad and not increasing even if all the words are matched.

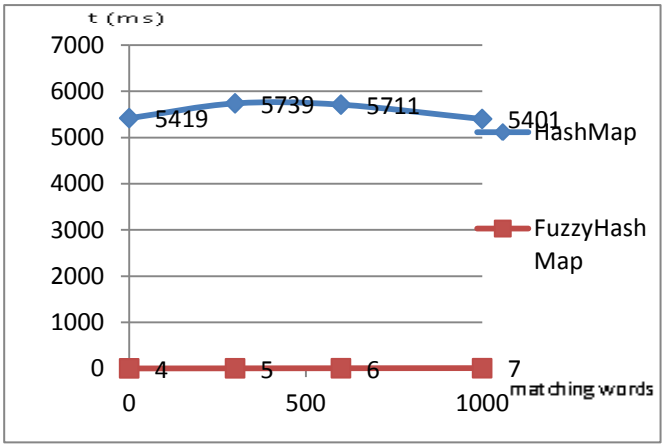

Figure 3.8 Testing Maps for Fuzzy Match only

b. In this test 50% of the words were exact matches and 50% fuzzy matches.

For this case, in Fig. 3.9 one can see the performance of HashMap is increasing proportionally with the number of matched words. Once again we can notice that FHM time consumption is almost constant for this test too.

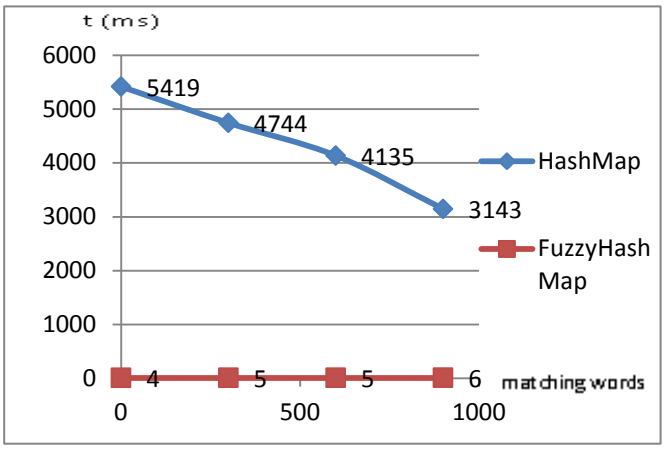

Figure 3.9 Testing Maps for Exact & Fuzzy Match

From all these tests the following conclusions arise: regular HashMap has better performance than FHM when all the searched words are exact matched, but it has very bad performance when the number of exact matched words decreases, and brute force was used.

3.3.1.5 Other FHM settings for reducing false negatives

Pre-hasing keys only by *first* n letters (first 4 in our case) lead to false negative results when the first letters of the inflected term (the term as found in the text) were different than the first ones of the canonical term. In order to avoid this and to reduce the false negative rates, the pre-hasing was changed to consider all fixed *n*-gramsof the original term (empirically we chose n = 5 as default, 5-grams). In our case *N-grams* represent all sequence of n consecutive letters from a given word. Using this method for pre-hasing increases the approximate matching of the FHM, but also has a side effect on the time performance.

3.3.1.6 Conclusions and a comparison with related work

As mentioned in the introduction of this section, there is a big amount of research done in the area of approximate string matching and we can consider this a well established research domain. Since the main objective of this thesis is text accessibility, it was not in the intention of the author to dedicate much effort for bringing contributions in the area of approximate string matching, but it was the need to have an easy and ready to use data structure that has fuzzy search capability.
The contribution that this research brings is that it implements exiting principles from the area of fuzzy string matching in the area of object oriented programming (OOP), more particularly on the existing Java HashMap data structure. To the author

best knowledge, dictionary specific data structures for Java programming language that allow efficient fuzzy search and also preserves a) simplicity of usage b) original O(1) performance when fixed search is done and c) very good performance for approximate search were not available. FHM was released as open source, and recently received attention from bioinformatics researchers in projects like [30] and [31].

4 Specialized language level – increasing fuzzy matching precision

The downside of using fuzzy matching techniques is that they are error tolerant, letting room for false positive matches. This chapter explores ways of improving the precision of the fuzzy matching. Matching precision is related to the rate of false positives. A *false positive* is a matching that was considered correct by the matching system, but in reality it is an incorrect match.

4.1 False positive types

The author gives special consideration to reducing false positives rates (increasing precision). Both automated methods and human based validation methods for increasing precision are proposed and tested. Human based validation is explored with persons like the end users of the application or workers from dedicated crowdsourcing platforms.

There are two major types of false positive, false positives on exact matches and false positives on fuzzy matches, the last one being of more interest for the author.

4.1.1 False positive on exact matches (word sense ambiguity).

False positive can occur even when only exact matching is used (no fuzzy matching), and both the word in the text and the associated term look exactly the same. This happens to words that can be used in multiple areas, and can have different meaning in different contexts. In natural language processing this problem is known as *Word-sense disambiguation* (WSD), being one of the oldest problems in computational linguistics (first mentioned in 1949 in [45]). For example the term "*immunity*" can be used both in medical language and justice/diplomacy language. In order to reduce the ambiguity in this case, solutions that are taking context (surrounding words) into consideration are suitable. Several such solutions exist, both supervised (most efficient) and unsupervised, most of them using surrounding words. Some are based on collocation; other classical solution is using language models based on N-grams, like the open n-grams collection released by Google from millions of digitized books [105], available at [46].

This type of false positive is highly influenced by the level of specificity of the terms from the dictionary used, and on the target text. Also this type of ambiguities appears when a word is used in different context and different discourses/ documents. Gale, Church and Yarowsky have shown in a research [48] that "the sense of a target word is highly consistent within any given document". So given the fact that the current research is designed to be applied on a specific language and

sublanguage (for example medical terminology is designed to be used on medical texts and not general ones, and so on), the chance of ambiguity is very low.

Considering this, the author did not focus on solving this kind of false positive, but gave a higher priority to the other major type of false positive, based on fuzzy matching which is being described below. In order to reduce the number of false positive on exact matches, the author chose to go on the simpler solutions of using more advanced/specific terminology, which is less likely to fall into this problem.

4.1.2 False positive on approximate matches

This type of false positive is a consequence of incorrect fuzzy matching, where words similar in syntax but different semantically are considered a match. This is the type of false positive that is more frequent when using fuzzy matching, and is the problem that is given most attention in this work

False positive based on incorrect fuzzy matching were also classified by the author in two distinct categories; false positive on incorrect fuzzy matching:

- With **words outside the sublanguage** (most frequent)
  Considering the medical sublanguage, this category represents words that are not medical related that are mapped on medical terms. For example consider word "*community*" mapped on word "*immunity*"
- With **words inside the sublanguage**
  Considering the medical sublanguage, this category represents medical terms incorrectly mapped on different medical terms. For example consider term "*hepatitis*" mapped on term "*keratitis*".

The author further describes in dedicated sections both categories of false positives based on incorrect fuzzy matching (with words inside and outside of sublanguage), providing details on the solutions found to overcome this problem.

4.2 Reducing false positives on words outside of sublanguage: Incorrect matching dictionary & training

Using fuzzy string matching has a big impact on reducing false negative rate, but because it is error tolerant it increases the rate of false positives too (mapping terms on words that are not related).

In order to decrease the false positive rates, an *incorrect matching repository* can be created. The repository is created by *training* the system with data that is outside of the target sublanguage (in this case text that is not medical related). This way, most words that are similar in syntax with term from sublanguage but different in meaning can be identified. Of course it is possible to encounter terms from target sublanguage in a non related text too, that is why at the end of this process, the resulting incorrect matches list needs a human revision.

For this *training* the author developed a module that takes as income an URL of a webpage, extracts the text from the given page, searches for all fuzzy matches

(exact matches are not taken into consideration) and saves them into an XML file. Identified fuzzy matches are saved together with their frequency of appearance and optionally with the context (phrase) surrounding them.

For Romanian language the author used various sources of text that are not related to medical language in order to find incorrect matches on medical terms. The system was trained with texts like:

- the Romanian constitution
- Romanian novels
- geographical and historical texts
- sport articles
- politics articles
- technical articles
- religious articles

This way the author could identify words that are outside of medical language, but when using fuzzy matching, they are mapped on medical terms.
After training the system with texts counting all together more than 200.000 words, a list of 248 incorrect matches was created, with occurrence frequencies ranging from 37 to 1.

Because even in texts outside of medical language it is possible to encounter medical terms too, a human (manual) revision is needed to make sure only incorrect matches are saved. After revising the list resulted from training, 45 mappings were actually considered correct matches. So in the end 203 out of 248 matches (82%) were saved as incorrect matches. Below is a sample of top ranked incorrect matches saved in the XML file used for *Incorrect Matches from non-medical text to Romanian medical terms*:

```
<match word="lumea" term="lumen" freq="37" />
<match word="teologie" term="sexologie" freq="33" />
<match word="color" term="colon" freq="23" />
<match word="media" term="medic" freq="20" />
<match word="zaharia" term="zaharina" freq="19" />
<match word="postate" term="prostata" freq="18" />
<match word="întors" term="entorsa" freq="17" />
<match word="centrala" term="ventral" freq="16" />
<match word="internet" term="internist" freq="16" />
<match word="medie" term="medic" freq="16" />
<match word="scorul" term="scorbut" freq="13" />
<match word="plastica" term="plastida" freq="10" />
<match word="vestul" term="vestibul" freq="9" />
<match word="centrale" term="ventral" freq="8" />
<match word="program" term="prognat" freq="8" />

...
```

The same process was done for English language, using non medical texts to train the system from similar sources (USA & UK constitution, historical, geographical, sports, politics, religious articles). After the manual revision a list of 290 incorrect matches was saved.

Below is a sample of top ranked incorrect matches saved in the XML file used for *Incorrect Matches from non-medical text to English medical terms*:

```
<match word="constitution" term="reconstitution" freq="46"/>
<match word="relations" term="ablations" freq="29"/>
<match word="position" term="deposition" freq="28"/>
<match word="community" term="immunity" freq="27"/>
<match word="positions" term="depositions" freq="21"/>
<match word="opposition" term="deposition" freq="19"/>
<match word="meeting" term="feeting" freq="19"/>
<match word="funding" term="fundis" freq="18"/>
<match word="Community" term="immunity" freq="18"/>
<match word="action" term="traction" freq="17"/>
<match word="entered" term="enteral" freq="16"/>
<match word="places" term="placebo" freq="15"/>

...
```

After creating the incorrect matches repository, it is integrated into the terminology interpreting service. Any approximate matching identified by the service is checked against this incorrect matching repository, and if found here, the match is considered invalid (is not considered a medical term). This integration was done directly in the *FuzzyHashMap*, when searching for a term the result is checked using this repository, and if the match is incorrect, the search continues without the need to initiate a new one. This way, we can improve accuracy and also preserver the speed.
A small test has been done to compare the precision of terminology recognition with and without the use of the *incorrect matching repository*. The evaluation was done by the authors on a web page [49] containing 2616 words. Both the initial implementation (without using incorrect matching repository), and the revised implementation (using validation with *incorrect matching repository*) were used to adapt the web page.
The initial implementation had a rate of 9 false positives within matches, while the implementation using incorrect matching dictionary had a rate of 2 false positives within matches. This shows that using incorrect matching dictionary has a big impact on reducing the rate of false positives.

4.3 Reducing false positives on words within sublanguage: matching model and hashing pattern

While in the previous section the case of incorrect matching of non medical related words on medical terms was presented in this section the author is looking on the other kind of incorrect matching, inside the sublanguage, when a medical term is mapped on another (non related) medical term, just because of their syntactic similarity. Such an example for Romanian medical language is the mapping "pediatrie" - "geriatrie".
In this section the author analyzed the syntax and internal structure of matching words, looking at the position of different (non-matching) characters and trying to find patterns in both correct and incorrect matches.

Studies were performed using text within and outside the target sublanguage. However, more importance has been given to text within the target sublanguage, since for the other category a solution has already been found (incorrect matching repository).
The study, done in multiple steps, tries to identify in-word matching patterns (at the character level), and if any pattern can be found, it is used to create a matching model. Then from the matching model a hashing pattern can be derived, that will be used to improve the hashing function used by the *FuzzyHashMap*.

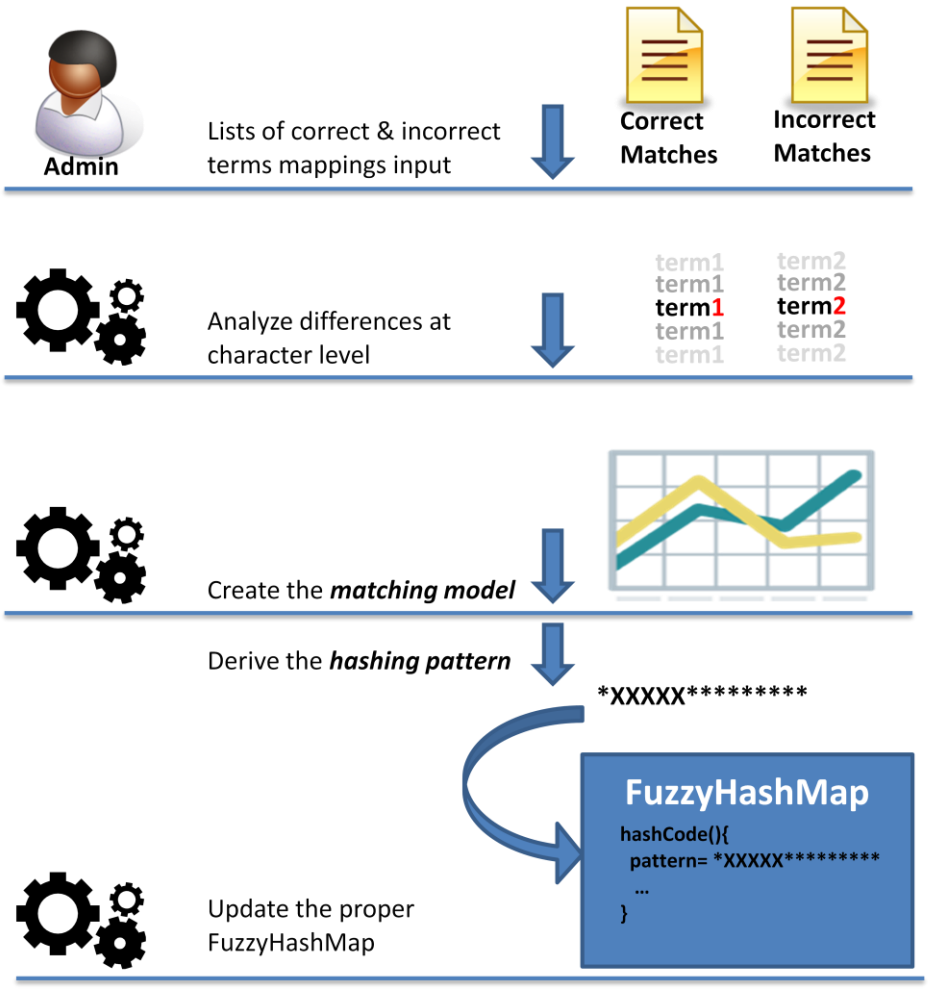

Figure 4.1 The process of updating the *FuzzyHashMap* with the sublanguage specific hashing pattern. The process shows parts that need human input (first step) and parts that are automated.

Here is the list of the steps proposed by the author in the process of obtaining the custom hash pattern and using it for the FHM:

Step 1. Search for patterns in incorrect matches
1.1.1 **Incorrect matches within sublanguage** (most important)
1.1.2 Incorrect matches outside sublanguage (optional)
Step 2. Search for **patterns in correct matches** (within sublanguage)
Step 3. Combine findings, create a **sublanguage specific matching model** and generate a new **hashing pattern** from the model

In this phase only terms-word matches that have the same length have been considered for analysis. The analysis was done on Romanian and English texts. Each of them is being detailed bellow.
Figure 4.1 presents an overview of this process, highlighting the steps that need human input and steps that are automated. In the current implementation the part of deriving the hashing pattern from the matching model is still assisted by the author, in order to assure the optimal hash pattern.

4.3.1 Methods explained on Romanian Medical Sublanguage

For analyzing correct and incorrect matches from texts within Romanian medical sublanguage the author used the results (list of correct and incorrect matches) from the *Study on Romanian medical websites.* The pre-hashing function used in the mentioned study was all 5-grams of the word, and the maximum edit distance allowed was 2. The three steps for obtaining the custom hash pattern and using it for is FHM for increased precision are detailed next.

Step 1.1 Find patterns in incorrect matches within Romanian medical sublanguage: For this the author selected all the incorrect matches from the *Study on Romanian medical websites* done with *text4all Term Analyzer* tool [50] described in use cases and applications chapter. Characters on equivalent positions from both words part of an incorrect match were compared to each other, and the result was stored in numerical form in an array. For each position where the characters did not match value "1" has been assigned, while for the other "0". This is illustrated in the example bellow:

```
pediatrie
geriatrie
101000000

carotide
parotida
10000000

biologie
miologie
10000000

flebologie
flebotomie
0000010100

genital
genitor
0000011
```

After all incorrect matches were analyzed, the resulting arrays were summed into a single array. The result in this case was:

```
14 3 5 1 2 3 12 2 0 0 0 0 0 0 0 0 0 0 0 0
```

In order to have a better image over the results, figure 4.2 is presenting a chart with the results.

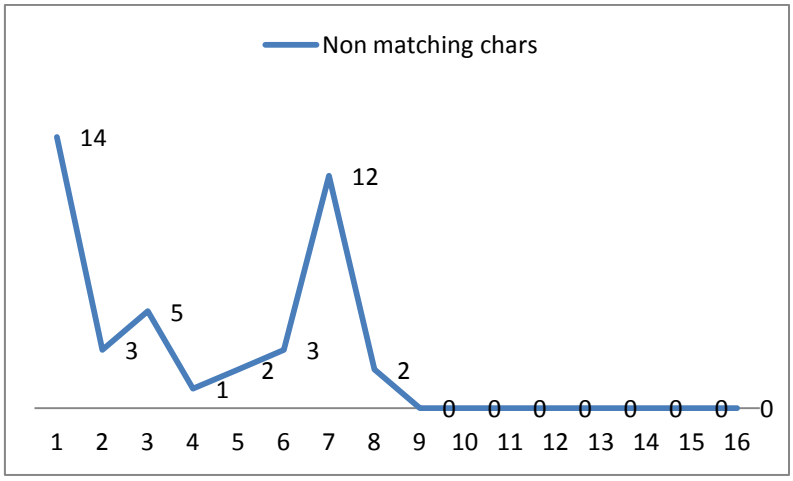

Figure 4.2 Positions of non-matching chars in bad term-term mapping (when using text within the domain/sublanguage)

From the graphic one can see that most frequent differences in incorrect matching words were in the first positions (1, 2, 3) and around the 7th position.

Step 1.2 Find patterns in incorrect matches on words outside of medical sublanguage: In this analysis the author looked over the details on matching between medical terms and unrelated words. The same process as in step 1.1 has been followed.

```
lumea
lumen
00001

color
colon
00001

...
```

Again, after summarizing all results the following array resulted:

```
23 5 12 6 34 34 22 3 6 1 0 0 0 0 0 0 0 0 0 0
```

The array is illustrated in Fig. 4.3.

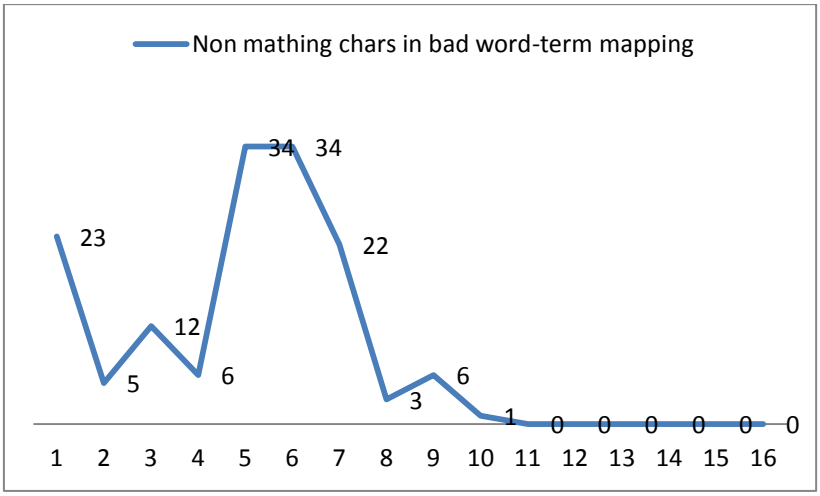

Figure 4.3 Positions of non-matching chars in bad word-term mapping (when using text outside of the domain/sublanguage).

One can see there is a similarity between the non-matching char positions from bad term-term mapping (within domain) to bad word-term mappings (when the word is outside of domain language). However, because for this case of bad mappings the author already proposed as a solution using the incorrect matching repository, the matching model found for this case will not be further used. Step 1.2 is an optional phase, and it is done just to illustrate the similarity with results from step 1.1.

Step 2. Find patterns from correct matches within Romanian medical sublanguage: Here the author selected all **correct matches** again from the *Study on Romanian medical websites* done with text4all Term Analyzer tool [50].

```
paralizia
paralizie
000000001

tomografii
tomografie
0000000001

conjunctivala
conjunctivita
0000000000110

tromboze
trombоza
00000001

biopsia
biopsie
0000001
```

The same process has been followed here, computing the differences array. Few mapping examples are presented in the precedent text area.

After all incorrect matches were analyzed, the resulting arrays were summed into a single array. The result in this case was:

```
0 0 0 0 0 6 10 14 15 17 6 4 3 1 0 0 0 0 0 0
```

Figure 4.4 illustrates the model of the correct matching. One can see here that most correct mappings have the different chars located at the end of the word.

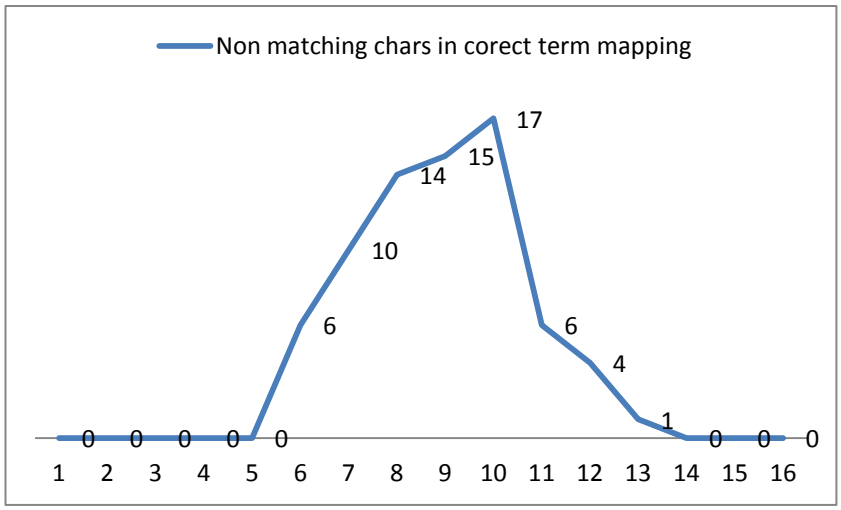

Figure 4.4 Positions of non-matching chars in correct term mapping (when using text within the domain/sublanguage).

Step 3. Combine findings, create a sublanguage specific matching model and generate a hashing pattern: In this phase we combine results from step 1.1 and step 2 in order to have a comparison between the model of differences between correct and incorrect matches within the sublanguage. Figure 4.5 presents these combined results.

One can notice that the differences occurring in the first positions clearly indicate an incorrect match (see the high number of different chars on positions 1, 2, 3 in the incorrect matching dataset). Differences at positions at the end of the word (starting with position 7) can represent both correct and incorrect matches.

In order to create a good hashing function, that will increase matching accuracy, several aspects are taken into consideration:

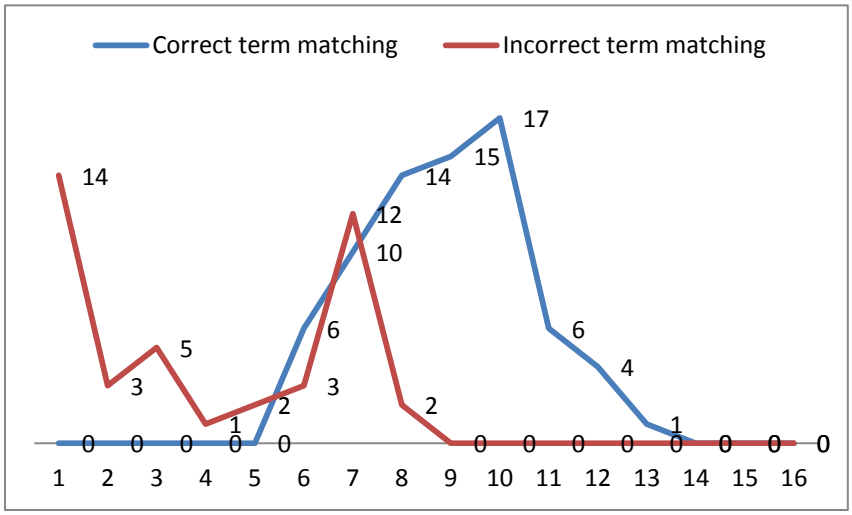

Figure 4.5 Number of differences at each char position (from 1 to 16) in incorrect term matching (red) vs. correct term matching (blue).

- Positions (with high number of non matching chars) that indicate mostly **incorrect matches have to be considered for hashing**, making these matches be invalidated right from the hashing phase
- positions that indicate mostly **correct matches have to be excluded from hashing**, so that such matches can be considered for further similarity comparison
- positions presenting an ambiguity (that reflect both incorrect and correct matches) are recommended to be **excluded from the hashing**, in order to avoid losing correct matches; In this phase, correct matches have priority, while incorrect matches are left to be identified in later phases (in string similarity comparison or in other post search validation).
- If all positions from the model are in the case of an ambiguity (number of incorrect matches is relatively close to the number of correct matches) **no hashing pattern will be derived**, and the original approach of using all 5-grams is recommended.
From the matching model illustrated in figure 3.14 some hashing patterns were decided. The pattern is presented by series of X and * (example: XXXX***) where X represents a position taken into consideration for hashing, while * is ignored in hashing.
Based on the correct matching model, all positions with no differences can be considered for hashing, obtaining the pattern:

XXXXX*********XXX

Since the last characters represent empty spaces (matching words were shorter, those positions remaining empty) they can be ignored; so the pattern becomes:

XXXXX*********

Although the hashing pattern was derived by the author after analyzing the matching models, in order to automate this process, so that it can be iteratively and frequently run by the system in order to improve matching performance, a formula

for deriving the pattern is needed. Below the author proposes a formula for automatically deriving the hash pattern (X or *) from the matching model. The current proposal only takes into account the correct matching model, since correct matching is being given higher priority. More work on enhancing this formula is planned as future work.

$$h(i) = \begin{cases} X, & v(CMi) \geq Avg(CM1 \dots CMn) \\ *, & v(CMi) < Avg(CM1 \dots CMn) \end{cases}$$

where:

$v(CMi)$ = value on position i, in the Correct Matching model

$Avg(CM1 \dots CMn)$ = average value in the Correct Matching model

The hashing pattern derived from the model of Romanian medical sublanguage is: XXXXX******** (first 5 characters will be used for hashing)

4.3.2 English Medical Sublanguage case

For analyzing correct and incorrect matches from texts within English medical sublanguage the author used the results (list of correct and incorrect matches) from the *Study on English medical websites.* Similar to the Romanian medical language study, the pre-hashing function used in the mentioned study was all 5-grams of the word, and the maximum edit distance allowed was 2.

The author will not detail again all the steps of the analysis, since this was already done for the Romanian medical language case. Here step 1 and step 2 will be omitted and the combined results from these steps will be presented (specific to step 3), the matching model will be studied and the hashing pattern will be derived.

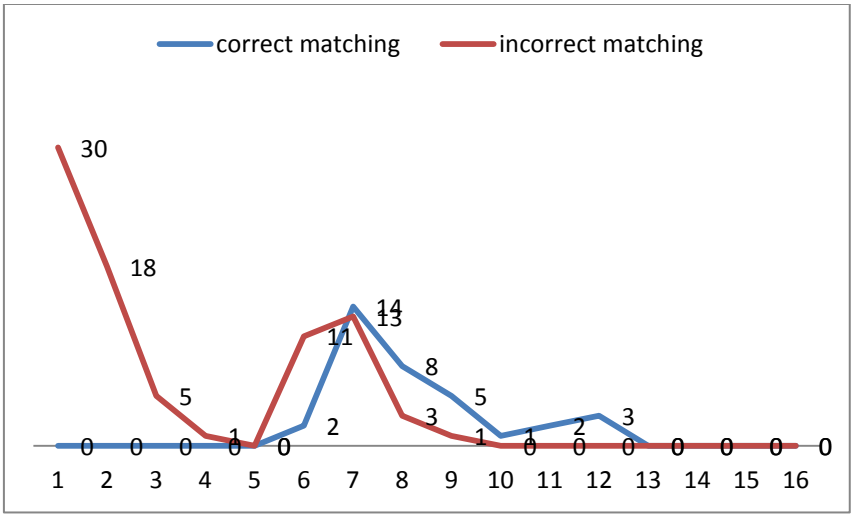

Figure 4.6 Number of differences at each char position (from 1 to 16) in incorrect term matching (red) vs. correct term matching (blue) for English medical language

After analyzing the differences at character level in word pairs representing incorrect mappings inside the medical domain and correct mappings, the result have been summed up in arrays (same way as in the Romanian medical case). The results were put together for comparison in a single chart that is presented in Figure 4.6.

Similar to the Romanian medical case, one can notice that the differences occurring in the first positions clearly indicate an incorrect match (see the high number of different chars on positions 1, 2, 3 in the incorrect matches dataset).

Differences at positions at the end of the word (starting with position 6) can represent both correct and incorrect matches.

The matching model for English medical sublanguage has proven to be very similar to the model of Romanian medical sublanguage. Due to this the hashing pattern derived from the English medical sublanguage matching model will be identical to the one used for Romanian medical text:

XXXXX********

The author considers this similarity of matching model just a coincidence and does not extrapolate the use of the same pattern as a general hashing pattern for other languages/sublanguages. It might be likely that other languages/sublanguages will have similar matching model because this model is specific to the use of suffixes. However, in order to achieve a good fuzzy matching precision on other types of texts it is recommended to follow the steps described here and to derive the hashing pattern from the obtained matching model.

4.4 Metrics and methodology for evaluating fuzzy matching efficiency

In order to evaluate the efficiency of the fuzzy matching system using the custom hashing pattern and compare it to the initial implementation the author run tests with both versions on three web pages containing Romanian medical terminology (different than the pages used in the terminology analysis).
Several classic information retrieval metrics were considered for evaluation: rate of false positive (fp), rate of true positive (tp), rate of false negative (fn), Precision (P), Recall (R), F1 score (F1). Last three metrics, Precision, Recall and F1 Score are popular metrics in the fields of information retrieval, pattern recognition and statistics. *Precision* and *recall* are among the first metrics to be used when evaluating the performance of an information retrieval system, being first mentioned as metrics in IR in 1966 by Cleverdon in [115].

Precision (also called positive predictive value) is the fraction of retrieved instances (in our case terms) that are relevant.

It is defined as:

$$Precision = \frac{tp}{tp + fp}$$

where:

fp = number of false positive
tp = number of true positive

Recall (also known as sensitivity) is the fraction of relevant instances (in our case terms) that are retrieved. Mathematically it is expressed as:

$$Recall = \frac{tp}{tp + fn}$$

where:

tp = true positive
fn = false negative

The relationship between precision and recall is illustrated in figure 4.7.

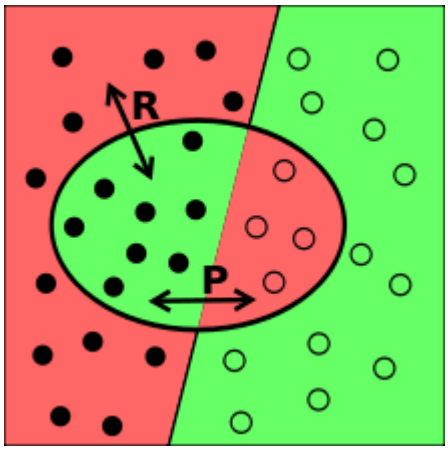

Figure 4.7 Relationships between Precision and Recall [51].

In Figure 3.16 the relevant items (correct matching terms) are to the left of the straight line while the retrieved items (all matching terms) are within the oval. The red regions represent errors. On the left these are the relevant items not retrieved (false negatives), while on the right they are the retrieved items that are not relevant (false positives).
A measure that combines precision and recall is the harmonic mean of precision and recall, the traditional ***F-measure*** or ***F-score***. The general formula uses a positive real variant β and it "measures the effectiveness of retrieval with respect to a user who attaches β times as much importance to recall as precision [52].

$$\boldsymbol{F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}}$$

The most popular *F score* of *F Measure* is the balanced F_1 score (having β=1) that is very used in statistics to evaluate tests accuracy. The author used this score to evaluate the accuracy of the terminology matching tests:

$$\boldsymbol{F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}}$$

4.4.1.1 Fuzzy matching evaluation for Romanian medical sublanguage

In order to measure the matching capabilities of the new implementations (using custom hashing pattern) compared to the initial one the author performed tests using both implementations on some web pages containing Romanian medical terminology (different from the ones from the matching analysis). Three web pages were used to perform the tests, formally called in the tables presenting the results as Web Page 1, 2 and 3. Here are the references to the actual web pages:

Web Page 1: http://medlive.hotnews.ro/video-interviu-dr-claudia-ciobanu-specialist-dermatolog-expunerea-la-razele-uv-este-principala-cauza-a-cancerului-de-piele-excizia-chirurgicala-a-alunitei-are-sanse-de-vindecare-de-peste-99-a-mela.html
Web Page 2: http://medlive.hotnews.ro/tratamentul-modern-al-cancerului-mamar-dr-radu-ionescu-medic-primar-chirurgie-plastica-si-dr-gabriel-gogescu-medic-primar-chirurgie-oncologica-discuta-online-cu-cititorii-joi-de-la-ora-13-00.html
Web Page 3 http://medlive.hotnews.ro/cat-de-periculoasa-este-bacteria-helicobacter-pylori-dr-alina-bondoc-medic-specialist-gastroenterolog-discuta-online-cu-cititorii-miercuri-de-la-ora-14-00.html

All of them represent interviews and discussions with doctors (about themes like skin cancer, breast cancer or stomach related problems) and were chose because the author considered this kind of web pages being of more interest for end users.

In Table 4.1 results from the tests performed with the fuzzy matching settings based on all subwords (5-grams), the initial implementation, are presented. Here one can see that the precision obtained using these fuzzy matching settings is 0.80 which corresponds to 80% precision.

When talking about recall, that takes into consideration false negatives too, a clarification has to be made. There can be two types of false negatives:

- false negative based on Fuzzy Matching: this represent terms that are present in the dictionary, but due to their derived form they could not be mapped on the dictionary term.
- False negative based on dictionary: this represent terms that were not identified because they were missing from the used dictionary, so the system could not identify them as terms.

When calculating recall the author only considers false negatives based on fuzzy matching, because this is an evaluation of matching accuracy. The false negatives based on dictionary can be solved by adding those terms into the used dictionary.

	Web page 1	Web page 2	Web page 3	Total
False positive	7	10	9	**26**
True Positive	42	30	43	**115**
False Negative (*Matching based*)	2	0	0	**2**
False Negative (*Dictionary based*)	29	11	20	**60**
Precision	**0.85**	**0.75**	**0.82**	**0.80**
Recall (Matching based)	**0.95**	**1**	**1**	**0.98**
F_1 Score	**0.897**	**0.857**	**0.901**	**0.88**

Table 4.1 Tests results using 1st implementation (based on all 5-grams indexing)

The recall for these tests, using false negatives based on matching, was 0.98 (98%). In Table 4.2 results from the tests performed with the fuzzy matching settings based on the custom hashing pattern, the new implementation, are presented.

	Web page 1	Web page 2	Web page 3	Total
False positive	1	1	2	**4**
True Positive	44	31	45	**120**
False Negative (*Matching based*)	2	0	0	**2**
False Negative (*Dictionary based*)	29	11	20	**60**
Precision	**0.97**	**0.96**	**0.95**	**0.96**
Recall (Matching based)	**0.95**	**1**	**1**	**0.98**
F_1 Score	**0.96**	**0.98**	**0.97**	**0.97**

Table 4.2 Tests results using 2nd implementation (Based on matching model and custom hashing pattern)

Next, figure 4.8 illustrates a comparison of results between the two tests, taking into consideration Precision, Recall and F1 Score. One can see that Test 2 (using custom hashing pattern) had a much better precision than test 1.

Recall however, remains the same in both tests. This was expected, given the fact that in this case, the custom hashing pattern used in Test 2 is actually a subset of the subwords hashing done in Test 1. It is also possible to find examples where the recall can be even smaller in Test 2 compared to Test 1, but based on the matching model, this is not likely to happen, and should be rare occurrence.

Summing up all metrics in the F1 score, the 2nd test, using custom hashing pattern, proves to be the best option, being much better than the initial implementation using all subwords (5-grams).

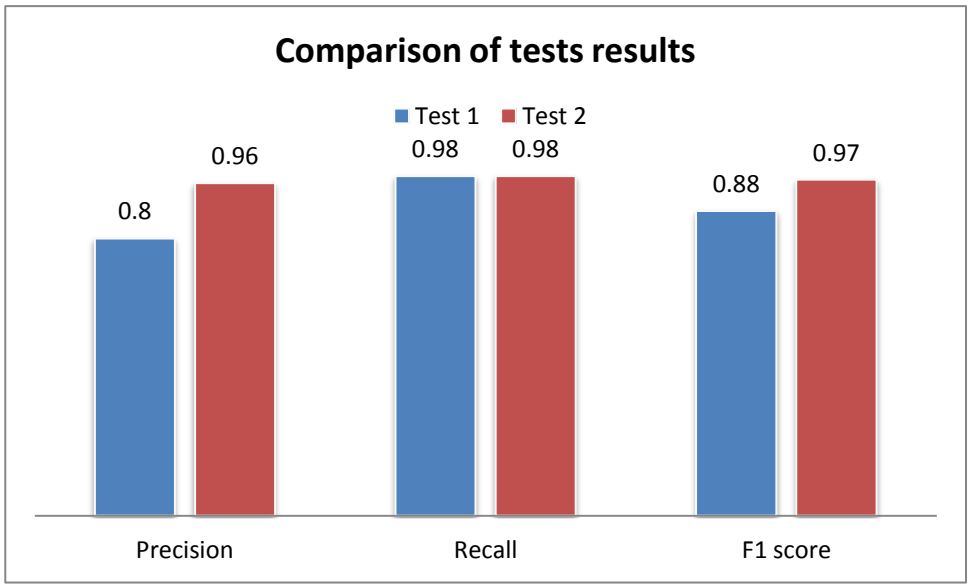

Figure 4.8 Comparison of matching accuracy results for both tests done on Romanian medical language. Test 1 represents test done with *all subwords* matching settings while Test 2 represents test done with *custom hashing pattern*

4.5 Human revision

In previous sections the author explored ways of increasing terminology fuzzy matching precision by implementing several ways to automatically identify false positives, using various automated processes or NLP and IR techniques.

Even if such techniques are highly efficient, they cannot guarantee full precision. In areas like natural language interpretation, machines and algorithms cannot achieve the performance of humans. This is similar to other areas like image interpretation, handwriting transcription where humans generally have better results.

This section explores ways of increasing terminology recognition precision using human input. Even if the target language is specialized language, the author investigates the performance of human revision done by non experts. For example,

for medical language, input from lay users (non medical stuff) is gathered, analyzed and summarized. Also, ways of managing all these aspects programmatically and automatically deriving the response upon the matching precision are presented.

The author distinguishes two types of human input:

- Feedback from application users (getting feedback from applications users using dedicated forms, automatically manage responses and derive decisions)
- *Crowdsourcing* based validation (paying workers from dedicated workforce platforms <in this case Amazon Mechanical Turk [53]> to answer a question)

Both cases are being illustrated in an overview of the system in figure 4.9 and presented in more details bellow, showing analysis done, conclusions driven and current state of implementation for automating the revision process.

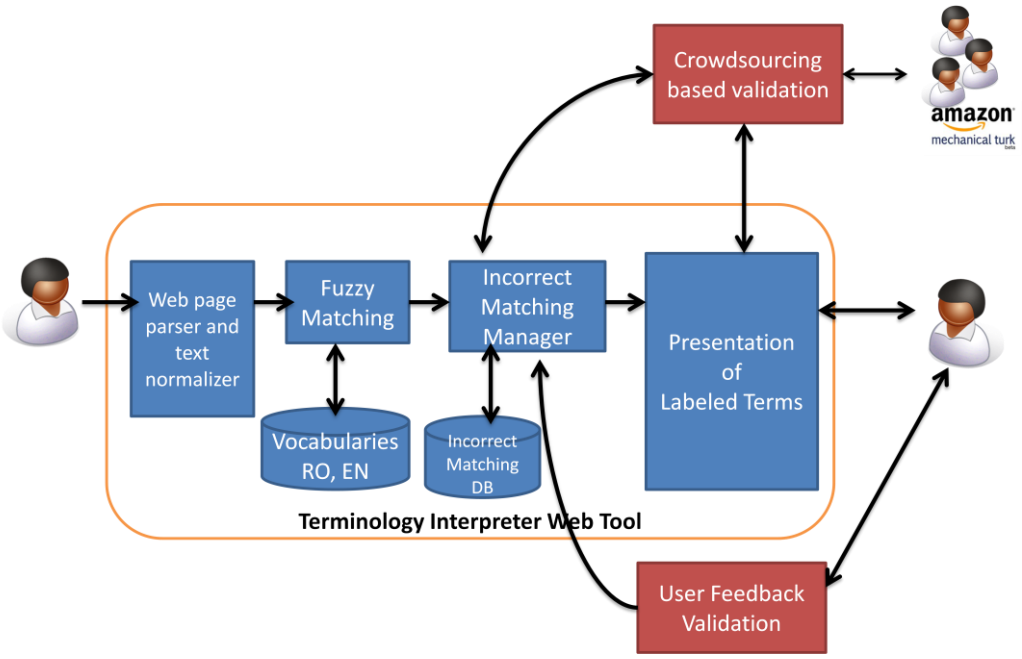

Figure 4.9 Overview of the terminology interpreter implementation focusing two human validation modules based on: a) user feedback and b) crowdsourcing

Methodology and Quality control Accepting feedback from users and automatically use integrate the results into the system (in dedicated data stores like incorrect matching repository) can be useful for dynamically improving the precision of the system, making it better and better while more feedback is being incorporated.

However, accepting feedback without some quality control over the received answers is dangerous, letting room for errors or malicious answers. Incorrect feedback can be received intentionally from spammers or users trying to full the

system or unintentionally due to the lack of knowledge or bad understanding of the question.
Two methods have been used for feedback validation:

- Trap questions
- Matching feedback from multiple users

4.5.1 User feedback validation

4.5.1.1 By trap questions

Trap questions is the first quality control method considered for user feedback validation. This method consists of combining questions with unknown answers with questions with known answers (also called trap questions, or control questions), without providing means for users to identify which is the trap question. Answer to the trap question is used to validate the responses.

The author used forms containing three questions, one with unknown answer and two trap questions with known answer. If the user answered the two control questions correct, the third (unknown) answer would be validated. In order to test this theory, a study has been performed, asking users to answer a form with three questions. The details of the study and the results are detailed bellow.

Figure 4.10 Questioner with medical term matching containing 3 pairs of terms, together with their context

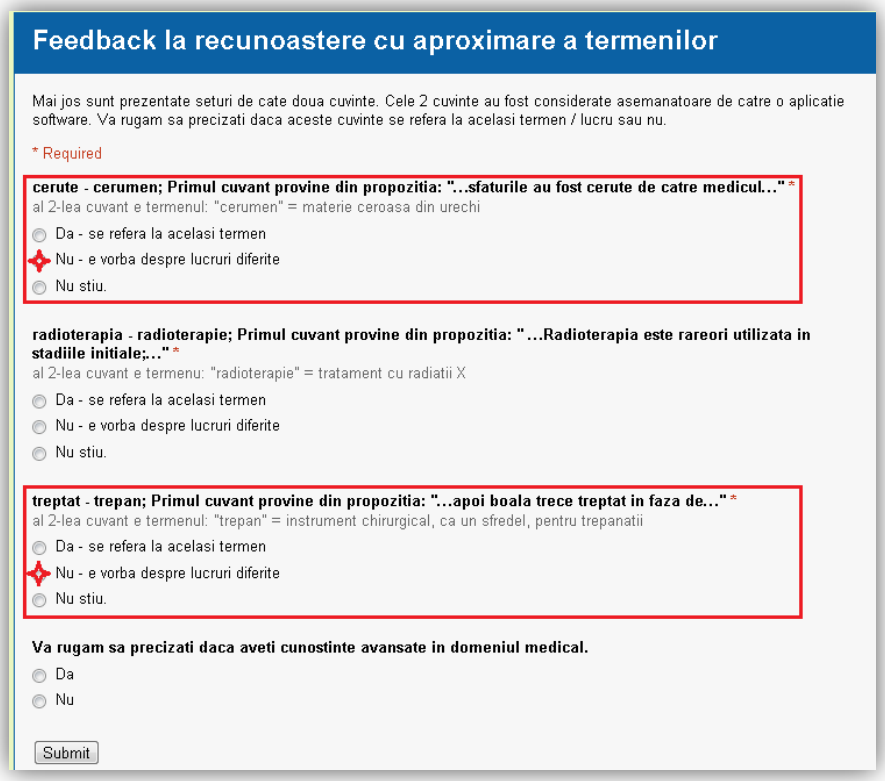

Figure 4.11 Questioner with medical term matching containing 3 pairs of terms, together with their context. The two trap questions are highlighted, the middle question being the one with unknown answer.

4.5.1.2 Trap Questions Study

In this study the author analyzed the efficiency of using trap questions to validate answers related to terminology matching and word sense similarity.

The study was performed using three forms, each with three questions, which were integrated into a web page that randomly selected one form to be shown to the user. Google Docs (now Google Drive) [54] has been used for the three forms; it was chosen because t offers lots of tools for response analysis and filtering. The webpage integrating and randomly showing one form was distributed on the internet using social networks (mainly *Facebook*) by the friends or connections of the author.

A total of 45 persons have answered the form. The only metadata asked was if the person answering the questions has strong medical knowledge. This was used to analyze the difference between lay users and users with medical background.

The question considered to be with unknown answer (further called *target question*) was the middle one (2nd) while the first and the last were considered trap

questions. Figure 4.10 shows how the questioner looked like while Figure 4.11 indicate the trap questions.

Table 4.3 bellow illustrates the distributions of answers for target questions and for the trap questions.

	Target question	Trap question
Correct answers	39	81
Incorrect answers	4	7
"**Don't know**" answers	2	2
Total answers	45	90

Table 4.3 Number of answers for the target and trap questions

The answers having "Don't know" response on the target question were ignored. The other answers were analyzed and 4 responses for the target answer were considered incorrect. **By applying the trap question filter, non on the four incorrect questions was removed** (in all four cases trap questions were correctly answered)**.** Trap questions only removed correct answers on the target answer. This proves that **this method, used in the conditions detailed above is inefficient for identifying incorrect answers on the target question**.

However, several interesting trends have been noticed from the study:

- Some of the forms having incorrect responses on the trap questions had the same response on all three questions (yes-yes-yes or no-no-no). This is a suspicious case, and for such cases this method might prove useful. In order to make this efficient, it is important to include trap questions with both incorrect and correct answers.

- Incorrect term matching, between one medical term and a non medical word, tend to be easier to identify than correct matches: we noticed that the rate of correct answers for incorrect term matching is 96.7% while the rate of correct answers for correct term matching was 93.4%. It is exactly the same for „Don't know" answers.

Conclusion

User feedback validation using trap questions has shown bad performance in this use case. It might be useful for filtering out suspicious users answering with no-no-no or yes-yes-yes responses, but for this the trap questions should be easy to answer, which from our tests means they should be questions related to mappings between a medical term and a non medical word. However, if used, this method should be accompanied by other methods for user feedback validation.

4.5.2 Validation by multiple user agreement

Another method of validating the responses from users are by tiring to find agreement for the same questions by matching responses from multiple users. Early usage of similar techniques can be found in the ESP Game [55], a computer game

that performed image labeling by matching responses from two users. Also it was designed to be entertaining, in order to motivate users to play the game. The project was very successful, having impressive results. Another project was trying to translate text on the web by looking for agreement in user responses in a dedicated game called Duolingo [10].

The author proposes a mechanism for matching responses from multiple users in order to validate them. The mechanism is not designed as a game, but relies on the user volunteer feedback. In order to test the feasibility of such validation mechanism, the author used the responses from the previous trap questions study (the 45 answered forms that were initially designed to be used with trap questions). Instead of considering multiple questions from which some are trap questions, multiple answers for a single question are considered. By analyzing the agreement between the answers from multiple users for a single question an answer is derived, being considered validated, or in case of disagreement between responses no final answer may be derived.

Looking at the nature of the questions in the trap questions study we can distribute the questions by their nature, and also show the number of responses and the agreement among them:

- Incorrect matching between a medical term and a non medical word; there were 4 such questions for the following mappings (in Romanian): dispret-dispnee, cerute-cerumen, treptat-trepan, program-prognat;
- Correct matching between two medical terms; there were 2 such mappings: oncolog-oncologie, radioterapia-radioterapie, benign-benigne
- Mappings between two medical terms that are related, but not necessarily indicate the same thing, thus the response is arguable; there was one such mapping: cerebral – cerebel

Mapping	Nr. of responses	NO response (incorrect mapping)	YES response (correct mapping)	"Don't know" response
dispret-dispnee	17	15	2	0
cerute-cerumen	15	14	0	1
treptat-trepan	15	15	0	0
program-prognat	13	12	0	1
Benign-benigne	13	1	10	2
oncolog-oncologie	17	2	15	0
radioterapia-radioterapie	15	1	14	0
cerebral – cerebel	30	24	6	0

Table 4.4 Distribution of responses from non-experts for each mapping from the study

Table 4.4 indicates the number of responses and their distributions for each mapping. Again, one can see there is a high agreement on incorrect mappings between non-medical words and medical terms.On medical terms that are related, but don't necessarily express the same thing, there is less agreement between answers.

The next question would be how many users should agree on an answer in order to validate it. The author decided not to concentrate on investigating this, and to rely on the results presented on similar studies, like the study done by Snow on crowdsourcing and NLP [56]. From such studies the author chose to check the agreement on 3 responses, and to derive and validate the answer for a question.

Response agreement - validation rules: The following rules were chosen from the studies mentioned above and from empirical analysis of the data from the trap questions study:

- If all 3 responses agree and they are not "Don't know" responses (so they are Yes or No), the common response is validated, and the mapping is treated and saved according to the answer (correct or incorrect match).
- If only 2 responses agree, and they are not "Don't know", the mapping is considered for another test (asking other 3 users)
- In all responses are different, or 2 or all responses are "don't know", the mapping is considered an incorrect map. Even if this may be introduce the possibility to consider correct mappings as incorrect ones, the data shows that this kind of heterogeneous disagreement appears when there is ambiguity about the mapping (like the case: "cerebral-cerebel"). This hypothesis is also sustained by the results expressed in the next section, from responses gathered via Amazon MTurk crowdsourcing platform. The author considers that is better and safer to consider as incorrect match cases with ambiguity.

MAPPING	Answer Expert 1	Answer Expert 2	Answer Expert 3
dispret-dispnee	No	No	No
cerute-cerumen	No	No	No
treptat-trepan	No	No	No
program-prognat	No	No	No
benign-benigne	Yes	Yes	Yes
cerebral - cerebel	No	No	No
oncolog - oncologie	***No***	***Yes***	***Yes***
radioterapia-radioterapie	Yes	Yes	Yes

Table 4.5 Responses from experts for each mapping from the study

The author also solicited responses from 3 experts in medical area (all are graduate students of medical faculty, activating in the medical sector, one in academic area, another one in a hospital and the last one in a pharmacy). The answers from the three experts are presented in Table 4.5.

One can notice that there is a case (*oncolog – oncologie* mapping) where there is disagreement between experts too. This aspect will be further discussed in the next section (crowdsourcing based validation).

By comparing the responses from non-experts with the ones from experts we can notice a high correlation between results. This indicates the fact that non-experts can achieve results similar to experts in validating word-term sense matching. Other aspects, like what type of mappings are easier to answer with precision, are presented at the end of the section describing crowdsourcing based validation.

Implementation

Integration of user feedback validation in the main project is under development. In the current design, the user will be asked to answer an answer about a word-term mapping via a dedicated window. An example of the screen asking for feedback is presented in figure 4.12. For future development it is considered to integrate this capability directly into the page displaying the adapted text, right on the word being labeled.

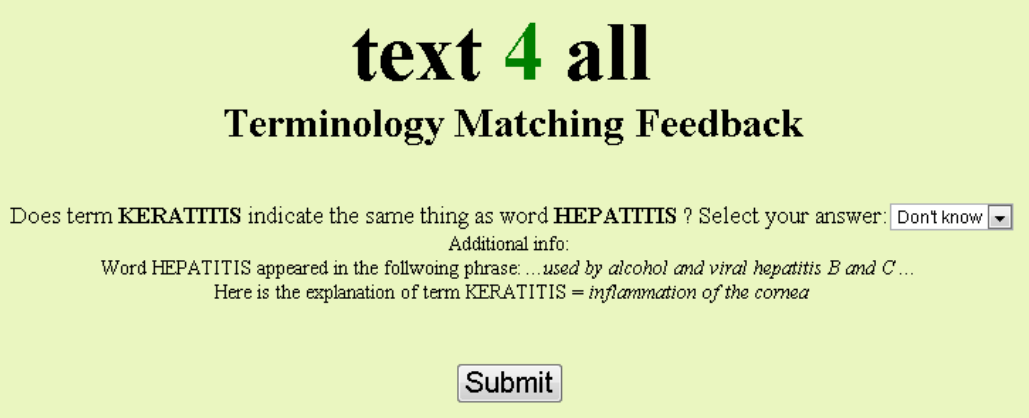

Figure 4.12 Screen capture of matching user feedback (current version)

In order to distinguish answers between different users, and avoid having the same question answered multiple times by the same user, the first usage of the service will save into the browser of the user a cookie (persistent variables saved into the browser) an unique user id, which will be passed on along with the answer at validation phase. This still doesn't completely eliminate the chance of having the same user answer multiple times the same question (let's say the user operates on multiple machines/browsers), but on a large usage of the system, such cases would be negligible.

4.5.3 Crowdsourcing based validation

Another option for getting human validation on word-term mapping is to use a crowdsourcing platform like Amazon Mechanical Turk (AMT) [53]. AMT is an online labor market where workers are paid small amounts of money to complete small tasks, named Human Intelligence Task (HIT). The design of the system is as follows: one is required to have an Amazon account to either submit tasks (HITs) <requesters> or to work on the submitted tasks <workers>. In this specific use case the requester is our system. These Amazon accounts are anonymous, but are referenced by a unique Amazon ID. A requester can create a group of HITs, each of which is in this case a form contacting one *word - medical term* mapping, and some context data. The requester can also specify how many workers should perform the task, which is the lifetime of the task and how much the worker will be paid for it.

AMT allows a requester to restrict which workers are allowed to perform a task by requiring that all workers have a particular set of qualifications, such as sufficient accuracy on a small test set or a minimum percentage of previously accepted

submissions. After the work is done, the requester can approve the answers (or can set to be auto approved) or it can reject it.

One of the main advantages that AMT has is that it offers a programmable interface (API) that allows you to submit HITs automatically and to process the response in the same way.

Related Work
Snow and all [56] shown in a research where several NLP processes were performed by non-expert human workers that the results were very accurate, being close to the ones of the experts. Also it has been observed that a small number of non-experts is needed to achieve the results of an expert. Results from two NLP process highly relevant for this research are being detailed below.

- Word Similarity – here one cans see in Fig. 4.13 the correlation between expert (dashed line) and non-experts.

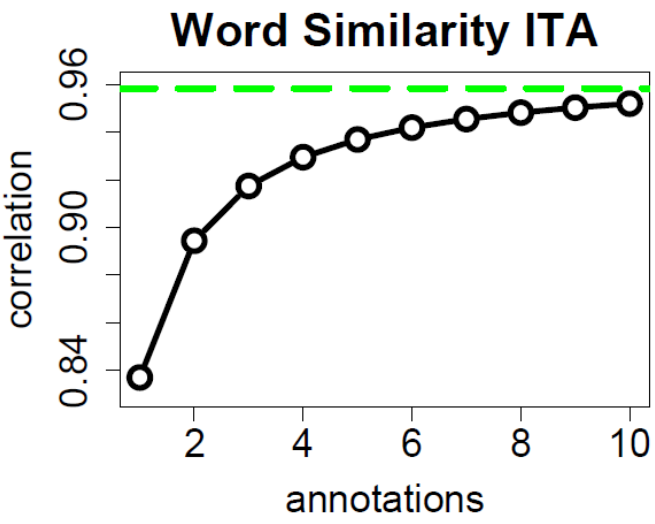

Figure 4.13 ITA for word similarity experiment [56]

- Words Sense Disambiguation. This is an interesting case where the accuracy of non-experts, is always bigger than the expert accuracy. Actually in this experiment, a mistake in the expert annotation has been discovered. The accuracy distributed on the number of annotators is presented in figure 4.14

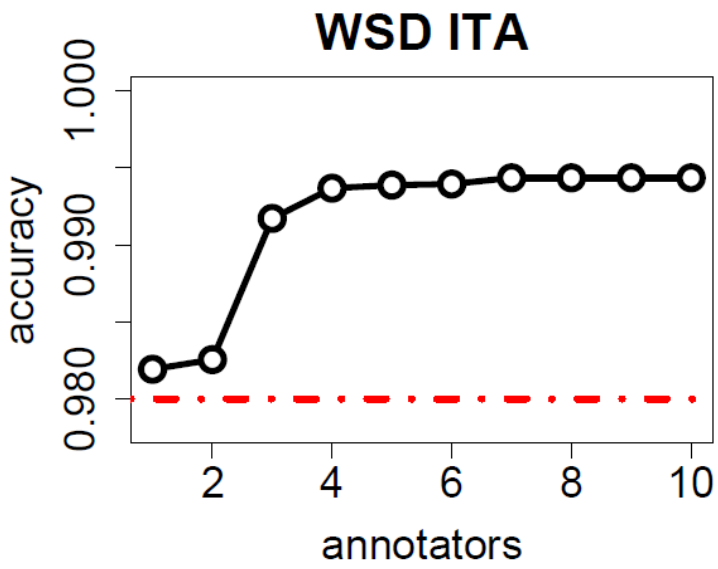

Figure 4.14 Inter Annotator agreements for Word Sense Disambiguation experiment [56]

From these study results one can see that non-experts can have a performance similar of the one of experts. Also this study helped the author decide to use 3 as the minimum number of answers from non-experts for a single question so that the accuracy is good enough. This helps preserve a good balance between <time, cost> and <result accuracy>.

Resolving word sense definition by crowdsourcing has been proven to be efficient by Anna Rumshiscky in [57]**.** Anna proposed the "construction of both an empirically-derived sense inventory and the corresponding sense-annotated corpus." For this the study relied on "the intuitions of non-expert native speakers to create an expert quality resource, and natively lends itself to supplementing such a resource with additional information about the structure and reliability of the produced sense inventories." The study has shown good results, and proposed several quality control strategies (like using multiple annotators, comparing the annotation to a gold standard, checking the annotation over the majority vote, using prototype-quality control step), some of them being used in this research too.

The same author, together with others have mentioned in a more recent study about word sense inventory creation by non-experts [58] that "the set of experiments we conducted suggest that there has been a significant change in the crowdsourcing marketplace since its early days just a few short years ago." This is an important note, and it emphasizes that using such NLP processes on crowdsourcing platforms should be reanalyzed from time to time since the behavior can change. They also mention that the "preliminary results indicate that it is possible to create a high-quality word sense inventories using the sense crowdsourcing methodology outlined above without significant modification to the HIT structure and cost. In a largescale
lexical development effort, however, even a small adjustment to the per-HIT pay would be costly."

Also, considering the importance of a good design for the HIT layout the authors mention "There is also a great deal of flexibility in HIT structure for the MTurk marketplace. Determining the exact HIT layout that is best for this task is more of an art than a science, requiring a balance between user-friendliness, cost considerations, and the cognitive limitations of MTurkers."

HIT design

In order to obtain answers from workers, a form has been designed, containing a single question related to a word – term mapping, and some extra data, to provide clues about the words. Similar to the user feedback process, the mapping was accompanied by the context where the word was found (the phrase that contained the word) and by the definition of the candidate term match. Figure X. illustrates the design and details of a HIT.

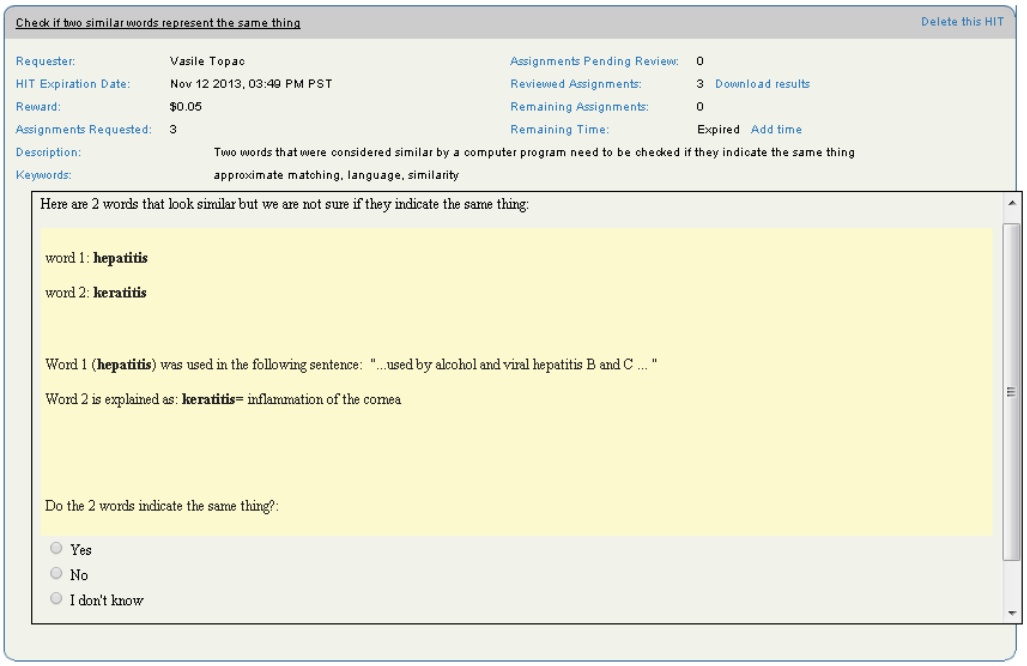

Figure 4.15 Design of a HIT with a form asking about the "hepatitis-keratitis" mapping

As seen in figure 4.15, the amount of money paid for a HIT was 0.05$, and each HIT had 3 assignments (should be responded by 3 users).

A test using five HITs, each one containing one mapping was performed. For each HIT were allocated 3 assignments (3 responses were expected). A number of 15 responses were received (3 pet HIT), all were approved.

HITs and answers
Bellow the distribution of responses received for each HIT (asking about a word-term mapping) is presented:

Mapping	Yes (correct match)	Don't know	No (incorrect match)
hepatitis – keratitis	0	0	3
tomato – stomata	0	0	3
ingesting – ingestion	3	0	0
bronchoscope – bronchoscopy	1	1	1
allergic – allergen	3	0	0

Table 4.6 Answers from **non-expert** workers on the HIT asking about mapping correctness (Yes answer meaning correct mapping).

Table 4.6 shows the mappings that were used in the test, and the distribution of answers. The mappings were chosen to cover all matching types: two medical terms incorrectly matched, medical term with non-medical word match, and few medical to medical term mappings, some correct and some with arguable response.

If using the same response agreement validation rules, as the one mentioned in the previous section (for user feedback validation) 4 mappings out of the 5 would be validated and saved in the repository and 1 would be considered as incorrect (allergic – allergen).

The same questions were then presented to 3 experts (same experts as the one mentioned in the previous section) in order to have a reference for comparison. Table 4.7 presents the responses from the experts.

Mapping	Answer Expert 1	Answer Expert 2	Answer Expert 3
hepatitis - keratitis	No	No	No
tomato – stomata	No	No	No
ingesting - ingestion	Yes	Yes	Yes
bronchoscope - bronchoscopy	No	Yes	No
allergic- allergen	No	Yes	No

Table 4.7 Answers from **experts** (medical stuff) about mapping correctness (Yes answer meaning correct mapping).

A surprising thing that can be observed in the results is that there can be disagreement between experts too. Similar to answers from non-experts, the questions asking about incorrect mappings between a non-medical word and a medical term look to be the easier to answer, having high answers agreement. However, questions on mappings between a medical word and a medical term, special when they are related seem to have answers with more disagreement.

The disagreement between experts, in the vision of the author, indicates several things like the fact that some mappings that contain related words but don't necessary indicate the same thing, are debatable and both answers, correct match or incorrect match, can be seen as right; this can indicate that there is a need for better questions design, or this can also raise the question of "who is the expert?"

and if medical stuff (all medical school graduates) can be seen as experts, or people with linguistic studies should be seen as experts.

Comparison between expert and non-expert answers
In order to measure the precision of the answers received from AMT workers, HIT responses were compared to the ones from the experts.
From the 5 mappings:

- 3 were in complete agreement (hepatitis – keratitis: NO, tomato – stomata: NO, ingesting – ingestion: YES)
- one was in partial/major agreement (bronchoscope – bronchoscopy: 2 answers <a Yes and a No> matched the ones from the experts)
- one was in partial/minor agreement (allergic- allergen: 1 answer <Yes> was in agreement with the experts)

By using the same response agreement validation rules as the one used for non-expert answers 3 mappings out of the 5 would be validated and saved in the repository while 2 would be reconsidered for answering again.

Comparing the validated and saved results from the non-experts (4 responses validated) with the one from experts (3 responses validated) one can see that there is a slight difference between the saved data. However, since all 3 expert responses that were validated are included in the ones validated for non-experts, and the other responses had disagreements between experts (sign of a debatable response) we can say that the responses from non-experts are accurate.

Programming the AMT

As stated before, a very big advantage of Amazon Mechanical Turk is the fact that it exposes a programmable interface (API). Using the API, developers can automate HIT creation and can manage responses programmatically.

Some similar research, like the thesis of Greg Little on "Programming with human intelligence" [59] exposes some very interesting use cases of AMT API, ranging from hand writing recognition to image description. The same author presented TurKit [60], a layer on top of the AMT API, which allows developers to easily write algorithms exploiting human computation via HITs. Little also presents modes of programming task on AMT, like parallel (where a task is given multiple assignments and more users can answer it) or iterative (where a task is given multiple serial runs, each run improving the result of the previous one).

The tools resulting from this research are designed by the author to integrate the use of AMT API, in order to automate the process of submitting HITs with word-term mappings (3 assignments per mapping, so this is the parallel mode of programming HITs) and to process the responses and decide whether the mapping is correct or not. If the mapping is considered as incorrect, based on the HITs responses, the mapping will be saved in the *incorrect matching repository*, if it is considered correct it is saved in the *correct fuzzy matching repository* or if a final response for the mapping could not be decided, no further action is taken. However, this integration of the AMT API with the tools developed by the author (presented in the next chapter) is not complete, so this section of improving precision by the use of responses from AMT API is not included in the public release versions of the tools.

Nearly-real time responses

Another benefit of using AMT for validating mappings is the potential of achieving nearly real-time capabilities. Such usage of AMT has been proven by Bigham et. all in VizWiz [61], an application that enabled blind users to take pictures with their smart phone, ask a question about the picture and receive nearly real time answers form AMT workers. VizWiz accesses the AMT trough a project that is a layer on top of AMT API and TurKit, named QuickTurKit, which recruits workers in advance and enables low latency responses.

For further development the author considers the use of QuickTurKit in order to decrease the latency of mapping validation.

4.6 Conclusions

The author presented several methods for improving precision of fuzzy matching for terminology. In the first part of the chapter two automated methods for identifying and filtering false positives were presented, first based on an incorrect matching repository and second by creating a sublanguage specifying matching model, deriving the hashing pattern and using that pattern in the FuzzyHashMap. In the second part the author lists validation modes based on human input, first relying on feedback from application users and second by accessing a crowdsourcing platform. Methods for automatically validating the input from users or from Amazon MTurk workers are explored. Methods relying on feedback from a single user and validated by trap questions seem to be inefficient, while matching responses from multiple users and seeking for agreement proves to be much more efficient.

An important finding is that incorrect mappings between a word not related to the domain and a term within the domain (in this case a non-medical word and medical term) are very easy to identify by non-experts, this type of questions had the most precise responses.

Orchestrating the entire process of 1) getting and saving human validation in the corresponding incorrect or correct matching database, 2) using the new entries for updating the sublanguage specific matching model and hashing pattern and 3) setting the new hashing pattern in the *FuzzyHashMap* is a solution for increasing matching precision in an unsupervised mode. This process, similar to unsupervised learning (or semi-supervised learning if we consider human input as a supervised process), will improve the system performance in terms of matching precision by time.

5 Specialized language level - Use cases and applications

5.1 Introduction

The previous chapters presented techniques for identifying and explaining terminology in natural language. A lot of attention has been given to reduce the rates of false negative by the use of fuzzy string matching and then to reducing the rates of false positive by using validation techniques like incorrect matching repositories, creating sublanguage specific matching model and deriving the hashing pattern and by also exploiting user feedback and crowdsourcing based validation done by workers. The techniques proved good performance, achieving to high precision and working with low latencies.

In this chapter the author presents several tools that resulted after implementing the techniques from the previous chapter in several use cases.

The first use case presented is the one of adapting medical language with the purpose of achieving patient empowerment. The ***text4all terminology interpreter*** tool has been created in this scope, and some user studies have been performed in order to analyze and validate the efficiency of these techniques and tools. The integration of this tool into a tele-assistance service (named Teleasis) designed for elderly persons are also presented. Another use case is when terminology meets foreign languages and needs translation. Although this thesis does not focuses on developing tools for language translation, in this use case the author developed a tool that would help translation by annotating terminology in a specially designed tag set. The resulting tool called ***text4all ITS Term Tagger*** annotates the encountered terms in the Internationalization Tag Set (ITS) version 2. The last use case where the presented techniques were applied is **language analysis.** The developed tool called ***text4all term analysis*** analyzes the language from a given web page and returns statistics of terminology usage in the target web page.
All the tools presented here are designed as web mediators and work directly into the browser, needing no install or special rights.

A detailed presentation of the inner design of the *text4all* service (the project that contains all instruments described in this chapter) is presented in Appendix C. The steps performed in the interaction between the user and the system, the interaction between inner modules and the project structure and design is presenting using UML diagrams (Use Case diagrams, Sequence diagrams and Class diagrams).

5.2 Patient empowerment use case – *terminology interpreter tool*

The case of medical specialized text is one of the most important where enhancing message understanding can have a great impact on the user. Offering a better understanding of medical messages to users is directly related to patient empowerment. Bellow the concept of patient empowerment is defined and discussed.

Patient empowerment is defined as helping people to discover and use their own innate ability to gain mastery over their disease or status [62] - by providing education for informed decision-making, assisting patients to weigh costs and benefits of various treatment options, setting self-selected behavioral goals, and providing information about the importance of their role in self-management.

The assessment of the improvement resulted by applying the principles of patient empowerment seems to be difficult, due to the lack of standards. The models identified are not universally accepted [63]. However, the obvious way to assess the impact of the patient empowerment is to use questionnaires [64] for a statistic analysis of the level of satisfaction of the patients themselves, and/or of the medical personnel involved. The results of this research also used questionnaires to evaluate the impact of the tool we developed.

The research and related work presented in this chapter considers the difficulty of understanding medical language and information a key limitation of patient empowerment. It is commonly known that medical language is very often hard to understand for lay people. Given this the communication between doctors and patients can suffer especially when dealing with remote communication that can appear in systems like tele-care systems or web page based communication. A research project, using a specialized classifier, tried to evaluate how easy it is for regular people to access data expressed in medical language reached the following conclusion "The classifier was then applied to existing consumer health Web pages. We found that only 4% of pages were classified at a layperson level, regardless of the Flesch reading ease scores, while the remaining pages were at the level of medical professionals. This indicates that consumer health Web pages are not using appropriate language for their target audience" [65]. This can affect in a grate manner the accessibility of the patients to their health information. Having a bad understanding of their health status may have a bad influence on their heath evolution. Empowering the patients with more understanding of the medical information related to them will strongly reduce this risk.

The classic solution in this area is language interpretation done by human interpreters. The presence of the interpreter makes it possible for the patient and provider to achieve the goals of their encounter as if they were communicating directly with each other. There are several international institutions like IMIA (International Medical Interpreters Association) [66] that are providing standards and frameworks for medical interpreters.

Another extensive study done by Alla Keselman and other [67] emphasized the importance of increasing *health literacy* for consumers and for developing tools and strategies for consumer-centered health communication. The study discusses the

potential of informatics to "support consumers by bridging two gaps: (1) between user needs and the content of information resources, and (2) between user competencies and resource complexity". The study also acknowledges the limitations of health resources, saying that although "the number of consumer-oriented resources keeps growing, their effective use requires significant lay knowledge and skills in areas ranging from health terminology knowledge to effective use of electronic media. Competency deficiencies among those who most require such capabilities result in a digital divide, or a growing gap between persons who can and cannot benefit from the proliferation of online health information."
We propose a Natural Language Processing (NLP) based tool to annotate medical terminology identified in raw texts or web sites.

5.2.1 Existing work

Text mining is a technique known to be used on medical content for purposes like automatic classification or information retrieval [68].

Classic machine translation tools can be used for translating medical content too, but these tools are very dependent on the training data. The text resulting from this process was evaluated in a research to be mostly incomprehensible [17]. A step forward solving this kind of accessibility issue is given by research and tools analyzing the level of accessibility of specialized language. One research [18] proposed a framework to inform the design of an "interpretive layer" to "mediate" between lay (illness model) and professional (disease model) perspectives.

Probably the most closely related research projects are:

a) A NLP solution, simplifying medical text by replacing difficult terms with synonyms and/or reducing sentence size, has been recently developed, having god results in terms of readability increase [19].

b) A tool identifying and explaining terminology from reports of electronic health records [20].

c) Consumer Health Vocabularies (CHVs) are a popular solution for increasing understanding to medical information for lay users by providing consumer/lay-friendly alternatives to the advanced medical terminology. Such approaches are presented by Zeng in [21] and such a vocabulary is available at [22].

The tool developed has some major differences compared to the existing work. Compared to [19] and [20], our tool does not replace the terminology in the original text, but it annotates the text with terminology explanations. Also, the use of fuzzy matching techniques was chosen because the tool is designed to work on natural language, and because we wanted to keep this project as much as possible language independent (the only dependencies are the dictionary for a specific language and optionally the incorrect matching repository, that can be used with this tool). The drawbacks of using a strictly controlled terminology system on natural language was also acknowledged in some other research [69], stating: *„Any controlled terminology will necessarily lack the richness of detail available from the vocabulary of a natural language; this loss of this detail is one of the trade-offs for having data in a computable form*." This terminology tools mainly falls into the vocabulary or glossary type, as categorized in [70].

5.2.2 *text4all terminology interpreter* tool

This tool is designed to adapt specialized texts by explaining terminology from either raw text or existing web pages. In order to make this terminology tool highly available, a web site and service has been developed, everything working. The tool can be accessed at the address specified at [71]. An overview of the implementation of the tool is presented in Figure 5.1. One can see the flow of the text from user input to fuzzy term matching, incorrect matching filtering, presenting the labeled text to the user, getting feedback for matching precision and validating and saving the feedback from the user. The matching validation done by using the Amazon MTurk crowdsourcing platform is not included, this work being still under development.

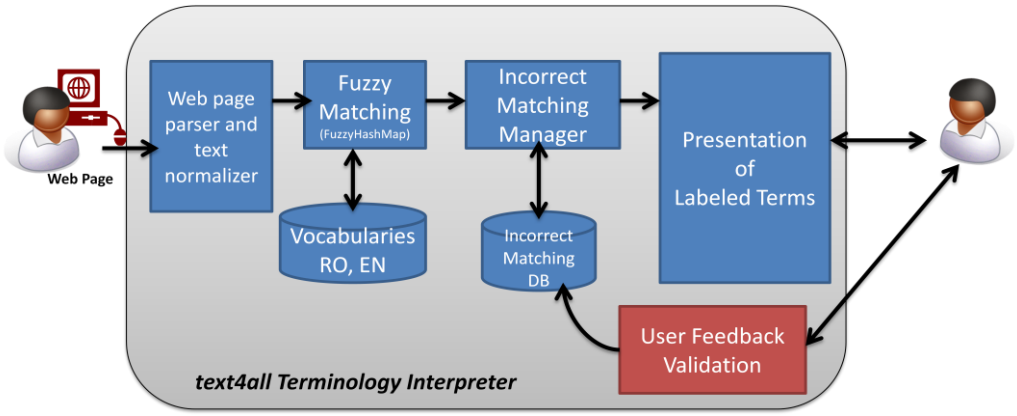

Figure 5.1 Overview of text4all terminology interpreter tool

Usability and interaction aspects of the tools were given much consideration, offering multiple modes for interacting with the application like:

- From web site user interface; this mode is presented bellow in this section
- Directly from URL; the URL was designed so that users can easily adapt the URL of the target web site to route to the text4all terminology service

Next the two use cases of the tool are presented, when adapting raw text and when adapting existing web pages.

Using the service for a custom text: Figure 5.2 presents the case when the user introduces a custom text, and obtains as output the text having terminology explained.

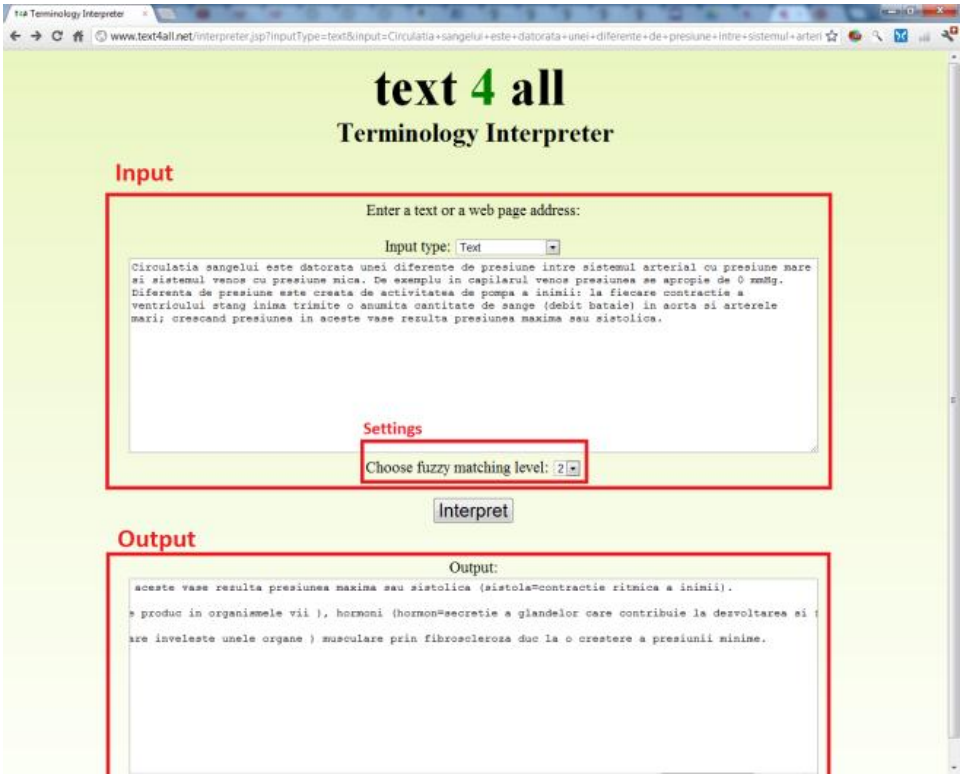

Figure 5.1 Medical terminology interpreter web service used for raw text

In the settings area, for advanced usage, the term recognition approximation level (fuzzy matching level) can be set by the user.

Using the service to adapt an existing web page: If the user wants to see terminology explained on an existing web site, he can enter the URL of the website in the input area. In this case the terminology service is acting as a mediator between the original web site, and the page displayed in the browser. As a result, the web service adds the explanation of recognized terminology as tooltip over the term.

The case when the end user is browsing a web page via the text4all Terminology Interpreter is presented next. In this case the original medical web page browsing is mediated by terminology service as illustrated in figure 5.2.

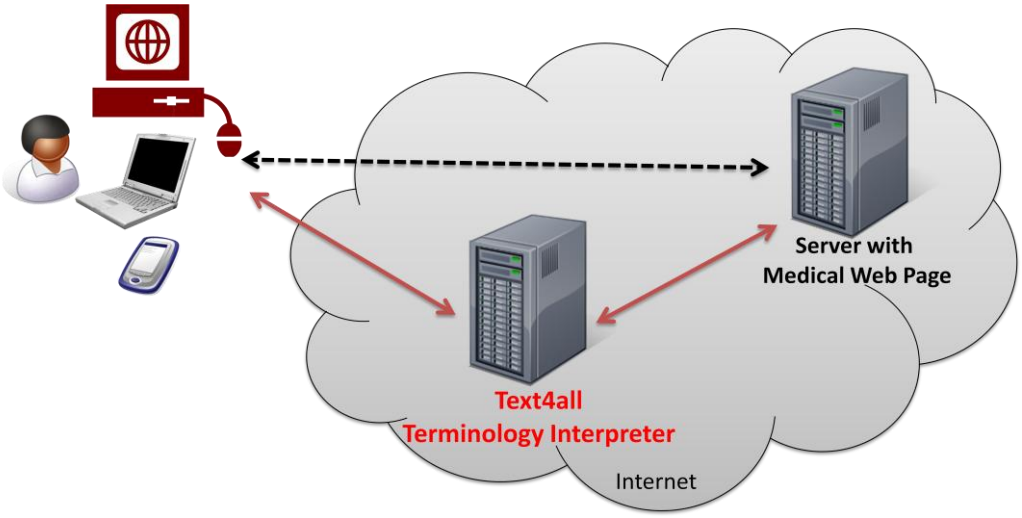

Figure 5.2 Medical terminology mediated browsing

The adapted web page is displayed in the browser having terminology explained. A portion of the adapted web page, having medical terminology identified and explained as tooltip is illustrated in figure 5.3.

Figure 5.3 Part of the adapted web page

In order to add more flexibility and to enhance usability, the user can choose how the definition of the recognized term will be displayed; for this there are two options: a) to mark the term as underlined, and add the definition of the term as

tooltip (this is the default display option); b) to append the definition in parentheses in the original text.

5.2.3 Evaluation methods

The main aspect considered for evaluation was *message understanding increase*:

For this the author decided to perform tests on content from a web site offering medical information for end users and interviews with doctors [72]. The text chosen was about the symptoms and possible treatment of thyroid cancer [73]. It has been selected because the author believed that this kind of web sites (educating end users about medical problems), and this kind of content, explaining symptoms and treatment can benefit more (in terms of patient empowerment) when the terminology is labeled. Also this content was rich in medical terms not so popular and known by lay persons.

The author used a questionnaire (formally called Q1) to evaluate the message understanding increase. The first items from the questionnaire were asking few details about the participant: age, education level and knowledge in medical field. The last item was used for avoiding biased results (filtering out participants that had strong medical knowledge, since they are not in the target audience). Age and education level were asked in order to analyze the results classified on groups of age and education.

The questionnaire was distributed online using mainly a popular social network (Facebook) and some other means (direct emails to some participants). On *Facebook* social network, friends of the author have shared the questionnaire, so that among participants there are mostly persons unrelated to the authors, but there could also be friends of the authors (this was considered less relevant for the test, and used other means to avoid biased results).

The questioner contained two texts (generically called text A and B) about thyroid cancer symptoms and treatment. One of the texts was in the original form, and the other had terminology annotated with *text4all terminology interpreter*. The questioner was designed so that it will randomly display text A annotated and text B as original, or text B annotated and text A as original. This was done in order to avoid influences over the results due to an eventual difference of difficulty between text A and B. The participants were asked to rephrase both texts, using their own words. Then an expert in the area evaluated the level of understanding in both rephrased texts, and decided whether or not the annotated text lead to a better understanding.

In the last question, participants were asked about the impact of terminology annotation.

Another questioner (formally called Q2) was designed to investigate user preferences over explanation presentation type, and the impact over reading performance.

5.2.4 Tests and Results

Several tests were performed, focusing on *message understanding increase, terminology recognition accuracy*, *latency* and *usability*.

5.2.4.1 Message Understanding Increase

Questioner Q1 described above was done in Romanian language, and had **41 participants**. From this, only answers from 37 participants were taken into consideration, since the other 4 responded affirmative when asked if they had strong medical knowledge.

Most of the participants were having higher education: 30 with higher education and 7 with secondary school or high school. One could argue that participants with higher education can lead to biased results, but the results of the test proved that this category can benefit more from terminology adaptation. Also, a big part of the consumers of online health information is composed of persons with higher education. Age distribution was: between 20 and 25 year: 14 participants, between 25 and 50 years 16 participants and over 50 years 7 participants.

On the answer to the question about the impact of the explained terminology over the message understanding the participants responded as indicated in Table 5.1.

Response	I understood the message better	It made no difference	It confused me
Number of respondents	29	7	1

Table 5.1 Answers on the impact of explained terminology over message understanding.

Rephrasing analysis: after analyzing the rephrased texts, the author found that **in 11 cases (29%) the annotated text was better understood** than the text without annotations.

Next, few other trends that were observed from the results, related to age and education are listed:

- The terminology annotation had a smaller impact over understanding increase for participants with lower education compared to those with higher education. This result was unexpected, but since there were only 7 participants with lower education, this finding should be validated with more participants.

- Participants over 50 years tend to reuse medical terminology in the rephrased text, although they were asked to use their own words.

Figure 5.4 and 5.5 illustrate a visual representation of results, both from users' answers and from the test asking to rephrase the message. One can see there is a discrepancy between how many users say this process helps them and how many actually seem to understand the message better. In any case, from both charts it results that this process can be helpful for users.

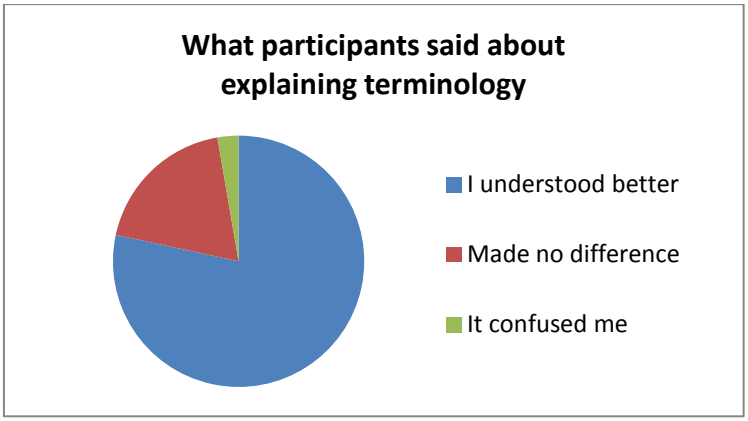

Figure 5.4. The distribution of answers from participants

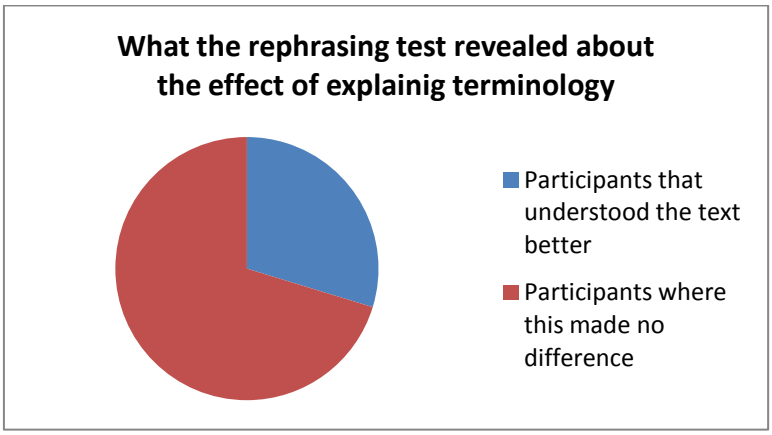

Figure 5.5 The results from text rephrasing test

Reading performance

In the other questioner (Q2), when asked about the impact of the explained terminology over the *reading ease,* the participants responded as indicated in Table 5.2.

Response	It helped me	It made no difference	It disturbed me
Number of respondents	15	3	5

Table 5.2 Answers on the impact of explained terminology over reading ease.

Latency: since this service can be used to mediate web pages browsing, speed is important, in order to maintain a decent browsing experience. After testing several web pages, an average of 3.5 seconds were needed (additionally to original load time of the web page) in order to load the adapted web page (the pages tested had approximately 2000 words per page). The author considers that this is a reasonable time, preserving a decent browsing experience.

5.2.4.2 Terminology labeling presentation & experience

The main focus was on the web page adapter, seeking the best way to present the explanation of the terms and other feedback. Currently the system can present the explained terminology inline (inserted into the text) or as tooltip over the term.

This has been evaluated in questioner Q2, where participants were asked what would be their preferred method to present the explained terms. The results are listed in Table 5.3.

Method	Inline (explanation inserted into the text)	Tooltip over the term	(right) click on the term	I don't know.
Respondents	3	7	12	1

Table 5.3 User preferences for explanation presentation mode

5.2.5 Discussion

The evaluation outcome shown that by recognizing the medical terms and adding the definition of the terms the understanding of the message can increase. However, the evaluation was done on a small medical text. It would be interesting to see if/how the results can change when performing the test on a long text. The author expects the outcome would be even better, since it is hard to make a resume of a short text (we could see a big difference in the number of people who answered that the tool helped them understand the message, and the number of understanding increase reflected in the summaries of the two versions).

It was surprising to identify in the results a trend indicating that participants with lower education were less helped by terminology annotation. Maybe for this category it is better to replace the terminology with synonyms. However this hypothesis needs to be further tested, since in this test there were only 7 participants from this category.

Related to usability, from the evaluation done with Q2 the author deduced that they should enable the web page adapter to offer the terms explanation by request (click or right click on term).

As a general rule, the author decided that in all cases the term in the canonical form will be shown before the definition. This is done in order to reduce the risks of misleading the reader in case of false positive terms recognition (due to fuzzy matching). This way the reader can easily identify and ignore errors.

5.2.6 Conclusions

Patient empowerment is important in order to increase the quality of life of the patients. This research and the suggested solutions contribute to this by supplying to the people in need, but not only, a more understandable and accessible information, even if that information contain terms difficult to understand, as medical ones.

The tool developed in this research is available for everyone at [71] and can be used in several ways, being customizable. Also, this use case of adapting medical terminology on existing web pages was published by the author in a journal article available at [110]. Aspects related to presenting annotated terminology on the web are included in paper [112].

The current results are encouraging, showing that this tool can increase the understanding of medical language. This also shows that fuzzy matching can be used for terminology recognition, in order to increase the recognition rate.

This work will continue with the focus on improving the medical text understanding, the accuracy of terminology recognition improving the existing (English and Romanian) medical vocabularies and adding vocabularies for more languages.

5.3 Integration into TELEASIS *tele-assistance* service

Another use case of the developed terminology annotation service is the integration into tele-assistance projects. Such systems have the role to monitor and remotely assist patients and also to offer access to information that will empower them into self care. One key element of patient empowerment is enhancing accessibility to information of interest. The TELEASIS system [74] suggests several ways to ensure that, like access to a central medical information database, access to additional communication channels.
The TELEASIS project has developed a pilot tele-assistance network with homecare electronic integrated services, allowing tele-assistance of the elderly, at their residence, based on the most recent IT&C technologies, with a medical and as well, a social target. The service-integrating tele-assistance system grants elders the opportunity to benefit from healthcare at home, to enjoy an improved personal lifestyle.

Medical personnel can a) monitor and b) communicate with the patient. Monitorization can be done customized by patient, and custom alarms can be defined, as described in [107] and [108].

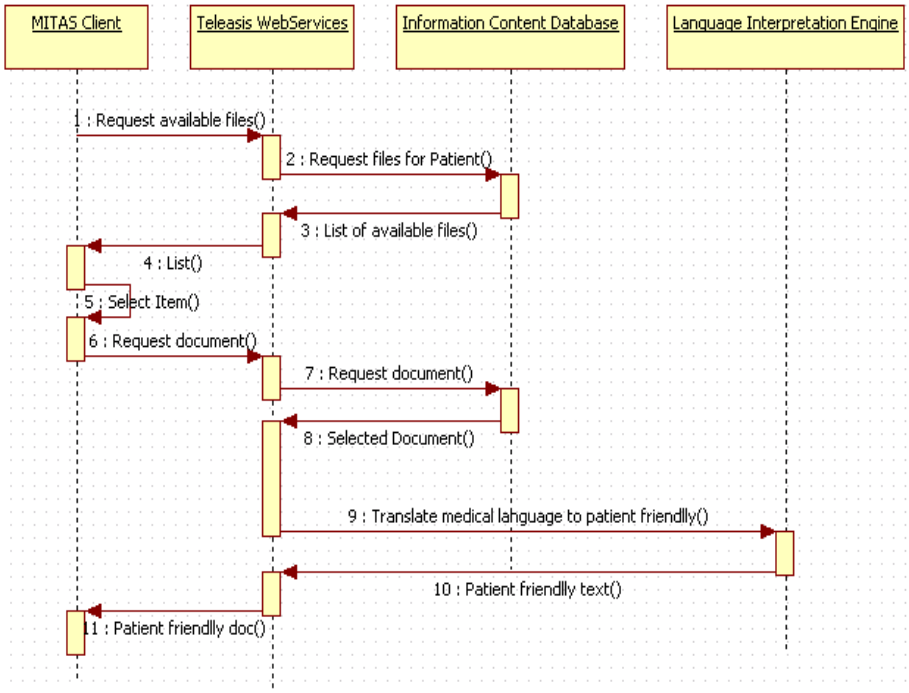

Figure 5.6 Getting patient friendly information

TELEASIS system is also offering patients access to their health data, reports and additional medical information. All this data is stored in an information and content database. Enrolled medical stuff or other power user can add documents to this database, and can set the access rights for patients or groups of patients. In this way each patient can access different documents. While allowing the patients to access medical information proves to be useful, as reminded in the introduction, the patients can encounter big difficulties in understanding that information. For this, TELEASIS is integrating the terminology annotator service that allows the patient to get the medical information adapted. The process of user getting a document containing medical information adapted to lay language from TELEASIS database is shown the sequence diagram illustrated in figure 5.6. Step 9 represents the process of terminology labeling.

Since this service has been proven efficient in other use cases, we considered that TELEASIS users will also benefit from it. However the author has not evaluated the efficiency of the service with real subjects (tele-assisted elderly persons) in this particular use case. More details about the integration of terminology labeling service into this telecare system can be found at [109]

5.4 text4all ITS Term Tagger tool

As mentioned before, on language level there are two main types of accessibility issues, one is foreign languages and the other one is specialized languages and advanced terminology. The solution to the first issue is translation, whether we speak about human or machine translation. In the case of machine translation, when this interacts with terminology things may get more complicated, because sometimes the translation doesn't use adequate terminology translation. For such scenario, a dedicated markup set named Internationalization Tag Set (ITS acronym) was defined and standardized [75].

5.4.1 ITS (Internationalization Tag Set)

W3C consortium mentions that "from the viewpoints of feasibility, cost, and efficiency, it is important that the original material is suitable for downstream phases such as translation. This is achieved by appropriate design and development. The corresponding phase is referred to as internationalization. A proprietary XML vocabulary may be internationalized by defining special markup to specify directionality in mixed direction text." There are other reasons too for having content ready for localization and translation. One of interest for this research is the correct translation of terminology, which is some areas like medical, laws and others may be very important. For this, a group within W3C and external experts worked on defining the ITS standard. Here is a short description of the second version (2.0 – current version) of ITS:

"*Internationalization Tag Set* is a *(ITS) 2.0* addresses some of the challenges and opportunities related to internationalization, translation, and localization. ITS 2.0 in particular contributes to concepts in the realm of metadata for internationalization, translation, and localization related to core Web technologies such as XML. ITS does for example assist in production scenarios, in which parts of an XML-based document are to be excluded from translation. ITS 2.0 bears many commonalities with its predecessor, ITS 1.0 but provides additional concepts that are designed to foster enhanced automated processing – e.g. based on language technology such as entity recognition – related to multilingual Web content." [75]

5.4.2 Terminology annotation

Given the existing platform developed for terminology annotation for end users, the author built a dedicated service and web page for annotating terminology based on ITS 2.0, section 8.4 <Terminology> specifications. Additional to this, all annotations done for end users in the other transcoding services also apply ITS 2.0 terminology markup in the original code. This was done because the resulted adapted pages can be used for further translation in MT systems, which can benefit from terminology markup. In this direction, Text4all already has embedded support for MT (currently Google Translate).

5.4.3 Existing work

There are already a set of tools and services designed for various ITS 2.0 annotations. Some are performing named entities annotation [76] in order to help machine translation tools easily identify and handle names. The most related project is the *Terminology Annotation Web Service (TAWS)* service done by Tilde, presented in **[11] .** This is a web service that takes as input a text (free text, HTML or other specific extensions) and some indicators of the language and domain of the text,

identifies terminology within that domain and returns an ITS 2.0 compliant text having terminology annotated. Behind the scenes this services uses a TaaS (Terminology as a Service) cloud service built on top of EuroTermBank terminology repository available at [77]. EuroTermBank is one of the largest high class European terminology repository, freely available online that was built based on best practices, as described in [78].

5.4.4 The ITS tagger tool

The ITS 2.0 terminology tagger that was built by the author is similar to the TAWS service, taking as input free text or HTML, identifying terminology and returning the text with terminology annotated based on ITS 2.0 specification. Additionally to TAWS, text4all ITS term tagger can identify and annotate fuzzy matching terminology (terminology in inflected form). It also automatically calculates and marks the "term confidence", based on the edit distance between the identified term and the term in the canonical form.

To illustrate how text4all ITS term tagger works let's consider the following text as input into the system:

"…is particularly helpful if pericardial effusion…"

The result of the annotation service can look like:

1. Simple Term annotation:

```
  … is particularly helpful if <quote its:term="yes"> pericardial
</quote> effusion …
```

2. Term with confidence annotation:

```
  … is particularly helpful if <quote its:term="yes"
its:termConfidence="0.6"> pericardial </quote> effusion …
```

One can see that along with the terminology specific annotation (its:term="yes") the annotation also supports marking of the confidence on the term annotation. Text4all uses this attribute (its:termConfidence) in order to mark the confidence of the identified text based on the edit distance from the canonical form. Given the high fuzzy matching precision, this approach is considered relevant by the author.

A screen capture illustrating the interface and usage of ITS term tagger is presented in figure 5.7.

The service is designed so that it can be integrated by third party applications, and the exposure of a production ready API is in plan.

While developing this tool, the author proposed the addition of a new attribute that would be used for terminology occurring in derivate form. The additional attribute would contain the canonical form (lemma) of the term, and would be used to help translation tools better identify the term, eliminating the need for lemmatization, steaming or fuzzy matching processes and improving precision. By the time of writing this thesis, this new attribute is considered a proposal and is still discussed in the W3C working group responsible for ITS 2.0

Figure 5.7 Screen capture of text4all ITS term tagger

5.5 *text4all Term Analysis* tool

While most use cases for terminology recognition are for adapting or annotating text within existing web pages, there is also a need for language analysis use case. Language analysis is an important process that can help better understand certain aspects of a language. In chapter 3 the author presented an analysis done on Romanian an English medical sublanguage, looking over the distribution of terms in canonical form compared to the one in derivate/inflected form. This analysis was done with *text4all term analysis* tool.
This is a tool that takes the address of a web page as input and returns statistics of terminology usage in the target web page. While this tool has been created to be

used in this research in order to better understand how terminology occurs in natural language in specific domains, it can be useful outside of this work, for linguistic analysis. Currently it only does the analysis dependent on the existing terminology dictionaries. As future work the author wants to enable the addition of external dictionaries (import dictionary function) in order to make if usable for any language/domain. Figure 3.1 presented an overview of the architecture of the tool. Figure 5.8 presents a screen capture of the tool, after performing an analysis. One can see the input area on top followed by the results area, which contains number about total words and terminology distribution followed by the list of terms together with their frequency of appearance.

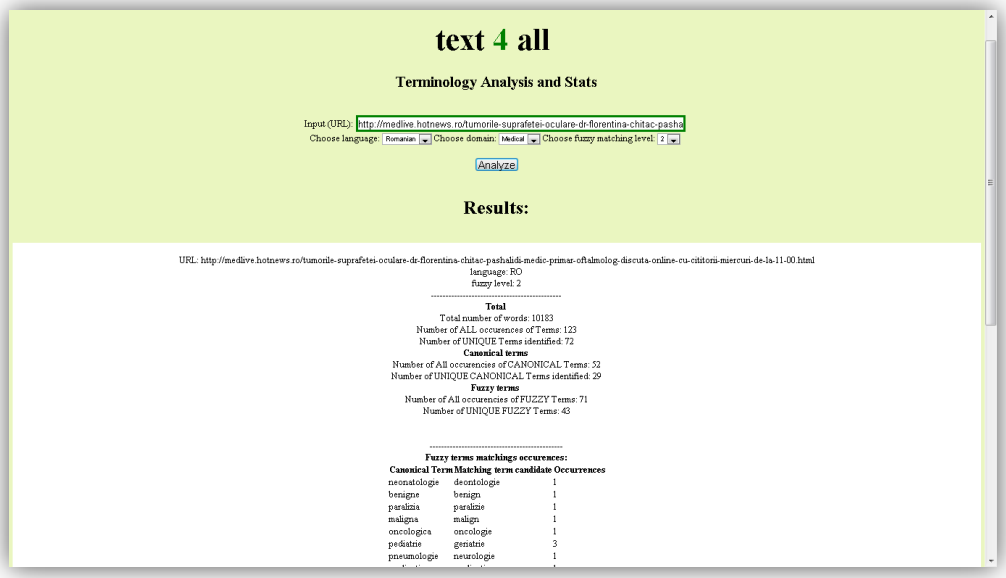

Figure 5.8 Screen capture of text4all term analysis tool

Tools design and technologies used

All the tools presented in this chapter (*text4all terminology interpreter*, *text4all ITS tagger* and *text4all term analysis*) are web based services that run with no browser or local dependencies.

Behind the scenes, the system is powered by a Java Servlet on the server side that is *transcoding* the original web page, adapting links (so that they redirect to an adapted version when the user selects them), applying special styles, processing and adapting language/terminology. All the terminology recognition and annotation is done by a dedicated module developed in Java. The module uses instances of *FuzzyHashMaps* for performing in-memory search for dictionary terms in a fuzzy manner. All the data (dictionaries, incorrect matching repositories) is saved and loaded from XML files.

On the client side HTML, JSP (JavaServerPages), JavaScript, JQuery plugin and Google Chart API are used in order to create and manage the user interface and certain UI elements.

The project is deployed and running on *AppEngine*, the cloud infrastructure from Google. The usage of cloud technologies enabled rapid deployment from the programming environment, and most important offers a reliable and scalable running environment. Scalability is particularly important for natural language processing tools, which can have high usage spikes due to the input of large texts, especially when used in clients-server architecture (when multiple clients use the application simultaneously). A load test was performed by the author by using *LoadImpact* web service [114], which simulates the simultaneous use of a web service/web page by multiple clients. Up to 50 virtual clients, stressing *text4all terminology interpreter* service (from various locations), were used, and the results shown that the service worked ok and the response delay was almost equal to the case when only one client accesses the service.

5.6 Conclusions

This chapter presented several use cases where identifying and explaining specialized terminology can be useful. All the examples presented are using considering medical terminology. The tools designed and developed by the author to cover the listed use cases are also presented. The most important tool, *text4all terminology interpreter*, is presented and evaluated with user studies. The studies show that the process of explaining terminology is useful and can help readers understand the message better.

Discussion: Why not replace terminology? An important question while designing this was whether we should only identify and explain medical terms (labeling) or we should make a complete translation (rephrasing), removing medical terminology and adding lay alternatives. Some medical texts (especially text contacting diagnostics) are considered, looking at the effect of replacing medical terminology over the authority of the message. An interesting study by Ogden et al. [104] has shown that by only using lay vocabulary instead of medical terms in diagnostics, the patients can under evaluate their health status and language may decrease the confidence level of the message for patients. The study concludes: "*Although much current prescriptive literature in general practice advocates the use of lay language in the consultation as a means to promote better doctor-patient partnerships, the issue of diagnosis is more complex than this. Patients attribute greater benefits to the use of medical labels for themselves and state that such medical labels are of greater benefit to the doctor*". So there has to be equilibrium among readability increase and side effect user risks that can occur in language adaptation. The decision to label terminology and present the explanation as tooltip or other means was preferred instead of completely replacing the terminology and / or rephrasing the text.

The other tools used for terminology analysis or annotation for improved translation using ITS2.0 standard are presented and discussed too.

All the tools presented here are functional and public available online at the addresses mentioned for each tool in this chapter.

6 Text adaptation at presentation level

6.1 Introduction

As mentioned in the universal view over text accessibility there are multiple layers (levels) of text accessibility. An important layer is presentation, or interaction. This layer is specific to text presented in digital of printed form. This research mostly focuses on digital text on the web, thus HTML annotated text or raw text on the web. Text presentation has a big influence over text accessibility and general reading experience, being important for both users with or without reading related disabilities.

Users with reading related disabilities are also being called people with *print disabilities*. According to Reading Rights Coalition [79] a print-disabled person is: *"A person who cannot effectively read print because of a visual, physical, perceptual, developmental, cognitive, or learning disability".* So print-disabled person are all those who have difficulty reading text in common designs and thus need to specify different text characteristics (World Health Organization 2011); including:

- people with low vision,
- people with declining eyesight due to ageing, and
- people with dyslexia and other reading-related impairments.
- people with color blindness
- others…

For all users with and without print disabilities, allowing customization of web documents proves to be useful. For internet users without such disabilities, this can be useful for temporal or contextual access limitations like:

- Environmental factors: (browsing in special light conditions)
- Mobility use (reading text on smart phone while walking or in the car)
- Time context (reading text in the evening, having tired eyes)

This part that takes into consideration the presentation layer of text on the web is identified in the figure 6.1 in the big picture of universal text access.

There is a lot of work done in the area of text accessibility on the web. Most of them relate to several accessibility standards specific to the web.

The major standard in the area of web accessibility on the web is Web Content Accessibility Guidelines [3] elaborated by the World Wide Web Consortium (also called W3C) [1]. There is a tremendous amount of research related to WCAG standard, ranging from web sites studies compared to the standard, automatic WCAG evaluation, developing websites that are compliant to WCAG, adapting web pages to make them compliant to the standard and so on.

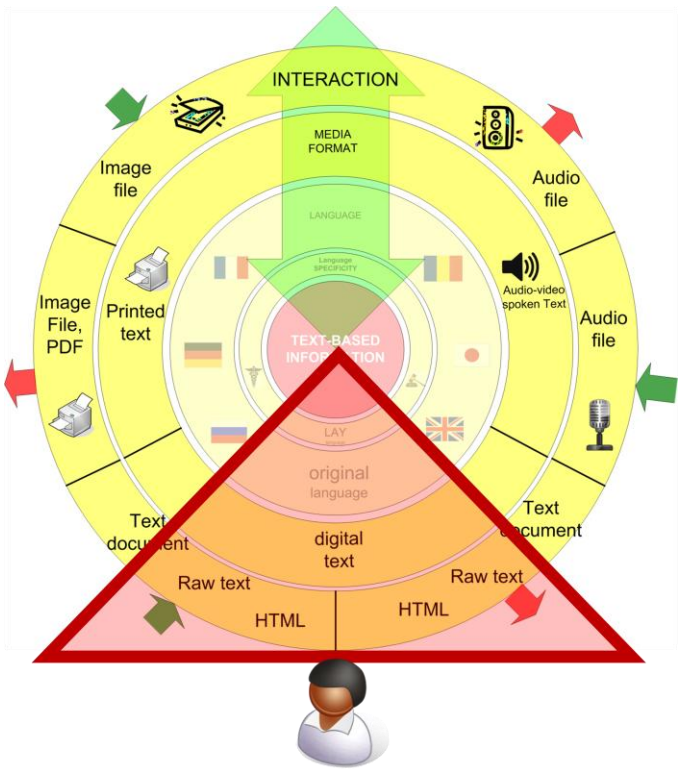

Figure 6.1 Web specific interactions on text presentation layer, in the universal text accessibility model.

6.2 Customization

However, in many cases, making web sites compliant to standards at a certain level might not be enough. There is a increasing need to make things customizable by end users in order to meet everybody's needs. It has been repeatedly mentioned that it is hard to achieve to designs where "***one fits all***", and the more realistic syntagm is "***one fits one***".
In the consortium held by W3C on text customization [80] multiple studies and examples indicated the need of allowing users to customize the web pages.

Henry describes the need for text customization in a research study [81] and drives several important conclusions:

- "Research results and published guidelines have different recommendations for many aspects of text display."
- "Without customization, a users' needs can conflict with general best practice."

- "Without customization, one user's needs can conflict with another user's needs."

Sanata et all also mentions in a research paper [82] that "end user customization plays a central role in accessibility considering dyslexia" and emphasizes the need of more study and standardization on such issues.

Motivating example
Some users complain about encountering reading issues due to the fonts used in the web page. For example Lee describes his experience with some fonts in a paper [83] stating "Unfortunately, I am unable to focus on text which is displayed on a flat screen when sub-pixel rendering software (such as Microsoft's ClearType) has been used to "smooth" the fonts. The text looks blurred to me (irrespective of adjustments). This causes visual disturbance and, typically, a bad headache as well."
All these studies and user experiences reveal there is a big need for customization.

Related work
Customization is mostly needed on text presentation aspects like: text color, background color, font size, font style and family and spacing (character, line).
In order to achieve text customization, users have a wide range of technical choices. Considering the location where the adaptation is done, it can be classified on three categories:

- **Client side**: This can be done by using custom browser options and settings, custom style sheets (formally called User Styled Sheets or USS) set in the browser, or dedicated browser extensions; Most major browsers have support for adding USS.
- **Server side**: web content authors can add customization capabilities directly into the web document (most common is text size adjusting or color schemes)
- **Mediators**: There are several mediator projects (also called *transcoding* projects) that can adapt the look of existing web documents. Their main advantage I that they usually need no installation, and are platform independent.

On the **client side**, a very popular solution used for text customization on the web is User Style Sheets (USS). Henry describes in a study [84] performed with multiple participants the experience users have while using USS, and investigates aspects of USS usage. She mentions that "users' approaches to USS varied:

- Global USS – Most users had one global user style sheet. One user had 2 style sheets for different situations: one for "reading" and another for "composing" (using a WYSIWYG editor). Some USS reset nearly all aspects of text display, and some set only a few aspects; for example, one USS changes only the line-height.
- Site-specific USS – Some users created style sheets for specific websites. One user said if he uses a website frequently, he copies the website's style sheet and makes revisions to it. He had over 150 website-specific user style sheets. Another user had four site-specific USS." And the study goes on emphasizing the diversity in use of USS.

Other client side solutions are browser extensions. A very popular extension is Readability [85] that allows users to see a clean version of a web page and to adapt

the look of the text. There are others dedicated software or browser extensions that are dealing with customization, however, this work does not focus on client side, system dependent tools, rather than on mediators.
Since this research focuses on tools implemented as web **mediators**, the category of mediators for text customization is more important here representing related work. There is some research done with text mediators, most related one being the Accessibility Gateway or Web Adjuster, a mediator done by Silas S. Brown [86] Web adjuster is mainly designed for low vision users, having several adaptable low vision styles but it also allows deeper customization of web sites.
Another tool that is designed as a web mediator is WebAnywhere [26] which is a screen reader on the go, and in it's latest version it also allows text magnification and high contrast schemes for the text being read. However the main functionality of this project is that of a screen reader that works directly into the browser.
Compared to Web adjuster, the tool created and presented by the author in this thesis has several differences, including the options to change the layout of the page in several modes, the option to integrate language adaptation and translation in the text customization process and a dedicated section for people with dyslexia.

Proposed tools
The author proposes some tools for fighting accessibility issues on text presentation layer by enabling adaptation and text customization of existing web pages. All the presented tools are part of the *text4all* project. For presentation layer text4all has the following services:

- **text4all Web Page Customizer** which enables text customization of existing web sites
- **text4all DysWebxia** which enables web page customization specific to persons with dyslexia

All tools within text4all, when performing adaptation on a web page not only that they adapt the look of the web page but also adapt the links, redirecting them to an adapted version using the same setting as the current page.

6.3 *text4all* Web Page Customizer

This is tool that enables users with or without print disabilities to adapt aspects of the original web page. It can be very useful for users with low vision or other print disabilities but it can also be useful for users without disabilities that encounter access issues dues to contextual limitations. Such a case is reading text from a web page with bright background while situated in a dark place. Also the same applies during the evening when eyes are tired.
Design considerations
The tool allows the user to enter a specific URL and to select the adaptations to be applied on the target web page. It behaves like a browser in a browser.
It has two frames, first is for URL input and adaptation settings while the second is for displaying the adapted web page. The settings area allows the user to choose form predefined adaptation templates (like template for low vision).
Adaptation settings include templates for users with low vision, users with dyslexia or custom. When using custom settings the following aspects of text can be

changed: text color, background color, font size, font family, letter spacing, line spacing.
Another adaptation that users can do is the layout of the web page. Three options are available:

- **Original layout**; in this case the original layout and structure of the web page is preserved
- **Serialized layout**: in this case the original layout and styling is removed, and the information is presented serially, as it appears in the HTML document
- **Clean Layout**: in this case the main content of the web page is selected, other sections like navigation menus, comments area and others being filtered out. This is similar to the way Readability project operates.

Technical details
Since this project mostly considers text accessibility on the web, the presentation of HTML data was taken into consideration. Presentation of HTML is controlled by using Cascade Style Sheets (CSS). CSS [87] is a mechanism for adding style (e.g., fonts, colors, spacing) to Web documents. Being standardized by World Wide Web Consortium (W3C) it is recognized by most existing web browsers. In order to adapt the look of the text on web the author studied way of adapting the CSS of existing web sites. The simplest way is to override the default CSS by using a custom one, with "*!important*" attribute.

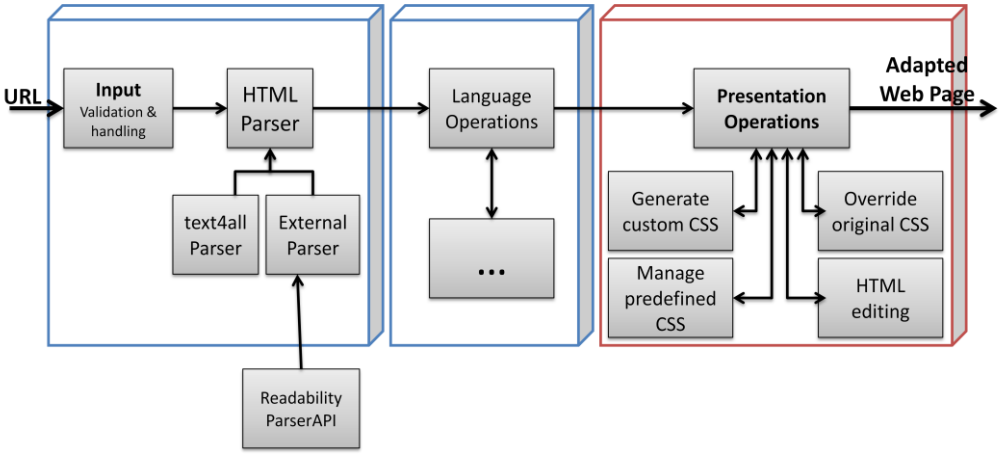

Figure 6.2 *text4all* architecture overview. The last module (from the right) handles presentation level adaptation

An overview of the text4all architecture, with focus on the module responsible for text adaptation at presentation level is presented in figure 6.2. In this module the original CSS is overridden with another one, which can be predefined or generated according to the customization settings entered by the user.

Examples Next, several adaptation examples are presented. The target web page is considered a Wikipedia article in English language about Diabetes [88]. The original web page is presented in figure 6.3.

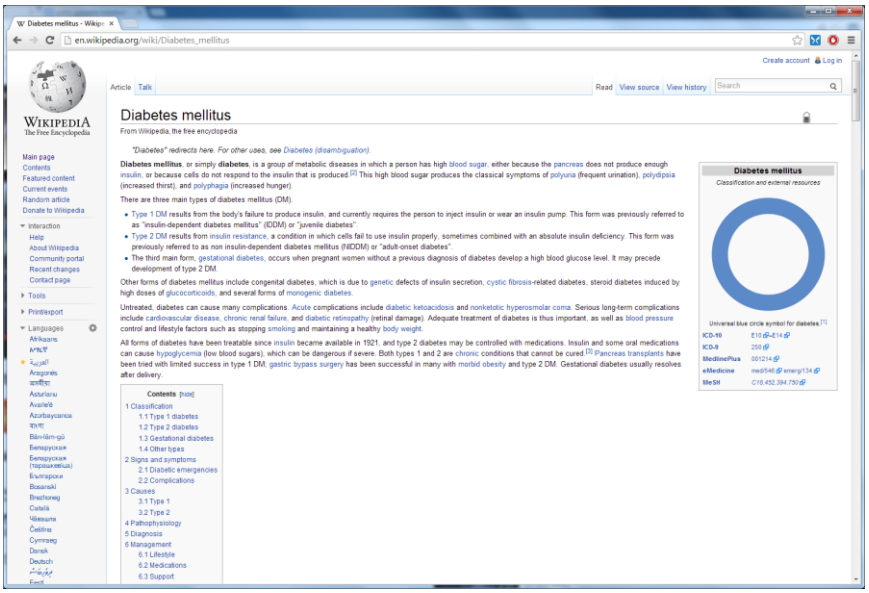

Figure 6.3 Original version of a Wikipedia article about diabetes

In the next example the adaptation will change the color of the text and of the background and the font of the text but will preserve the original layout. Figure 6.4. illustrates this case. Following that, in another example the same page is adapted using the Low Vision template. This adaptation changes the layout of the web page, serializing content, changes colors, font sizes and assigns different colors for some HTML elements, like links. The template is based on a style developed by Silas S. Brown [89]. It is illustrated in Figure 6.5.

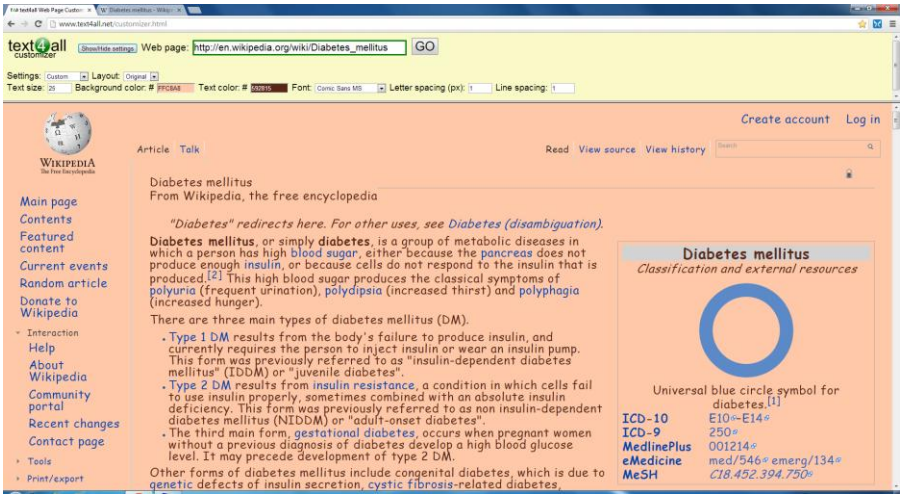

Figure 6.4 text4all Web Page Customize showing an adapted Wikipedia article having font style, size, text color and background color changed. The original layout is preserved.

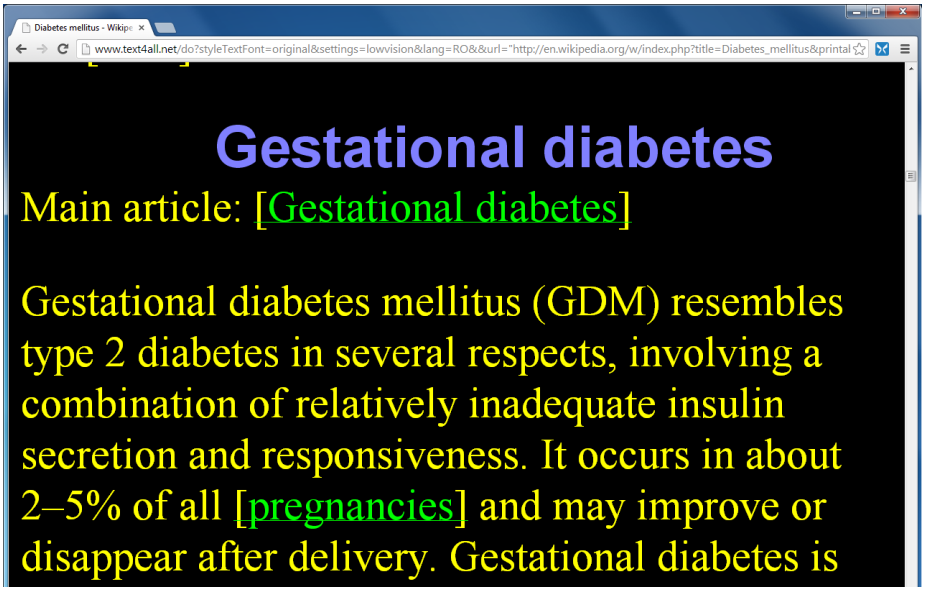

Figure 6.5 text4all Web Page Customize using the Low Vision settings

In conclusion, by offering multiple text customization options, text4all Web Page Customizer can be useful for a wide range of users with print disabilities, including low vision users or elderly persons.

Being designed as a mediator the service has the advantage of working anywhere, directly into the browser. However, the downside of this approach is that it has limitations on the web sites it can adapt, being unusable on complex websites like Facebook, Twitter or online email.

This specific tool (Web Page Customizer) has been tested by two persons with low vision, their option being useful in the development phase. However some more extended user studied is planned as future work in order to validate the usefulness of the service. Other technical details and challenges were included by the author in a dedicated paper available at [111].

6.4 text4all DysWebxia

This section is dedicated to users with dyslexia. Similar to Web Page Customized, the DysWebxia section offers users with dyslexia the options to adapt the web page to better fit their needs. This section was designed in collaboration with Luz Rello, and most of the adaptations done are based on the research done by Luz with users with dyslexia. Most of the research done by Luz consists of eye tracking studies with lots of persons with dyslexia, followed by detailed analysis. Several user studies [90] [91] [92] [93] [94] presented by Luz specify some guidelines for the look of the text so that text becomes easier to read for persons with dyslexia. These guidelines, that constitute the DysWebxia 1 model, have been taken into

consideration in order to design and develop the adaptation options of text4all. Several aspects of the adaptation include:

- adapt colors scheme (text and background colors)
- adapt text size
- adapt fonts (fonts like Arial and Helvetica would be recommended)
- spacing seems to be less important, according to results from [94]

DysWebxia service can do more than adaptations on text presentation level. It also performs several language level adaptations, also suggested by research study including eye tracking study with persons with dyslexia. Adaptations at presentation and language level are part of the DysWebxia 2.0 model presented in paper [95], which includes text4all. The language level adaptations are:

- Showing synonyms for difficult words, as suggested in research [96]
- Replacing numerical expressions with numbers
- Replacing fractions with percentages, as suggested in research [97]

For the adaptations done at language level three languages are currently supported: English, Romanian and Spanish. text4all DysWebxia also supports the adaptation of page layout.

In figure 6.6 the home page of text4all dysWebxia project is illustrated. One can see that the page is composed of an input area for the target web page address, a settings area for customizing the adaptation and a live preview area.

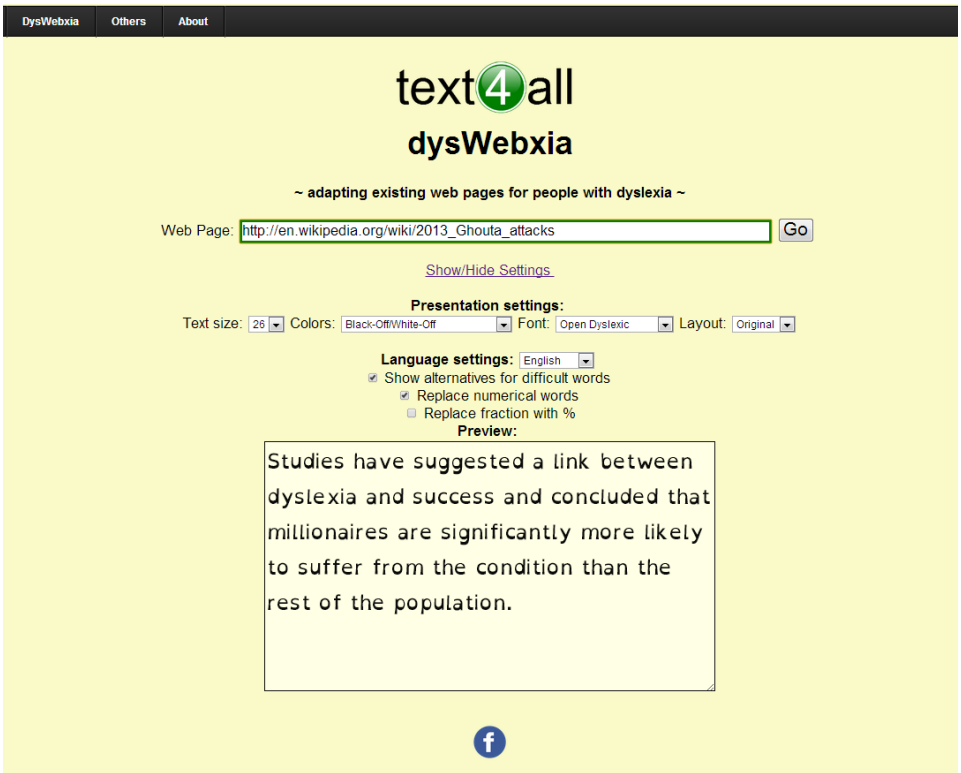

Figure 6.6 *text4all dysWebxia* web page

Similar to the Web Page Customizer, the DysWebxia service adapts the look of the text by overriding the cascade style sheets of the original website. It proposes a default adaptation based on the text settings with the best reading results in the studies performed by Luz and also enables the customization of adaptation.

At language level adaptation, for synonyms some dictionaries are being used, while for replacing numerical expressions with numbers and fractions with percentages basic string parsing and replacing is performed.

Figure 6.7 presents a comparison between the original web site and the web site adapted using text4all DysWebxia. One can notice the additions of tooltips with synonyms for complex words. Also the original layout of the web page is preserved, also preserving other functionalities like video streaming.

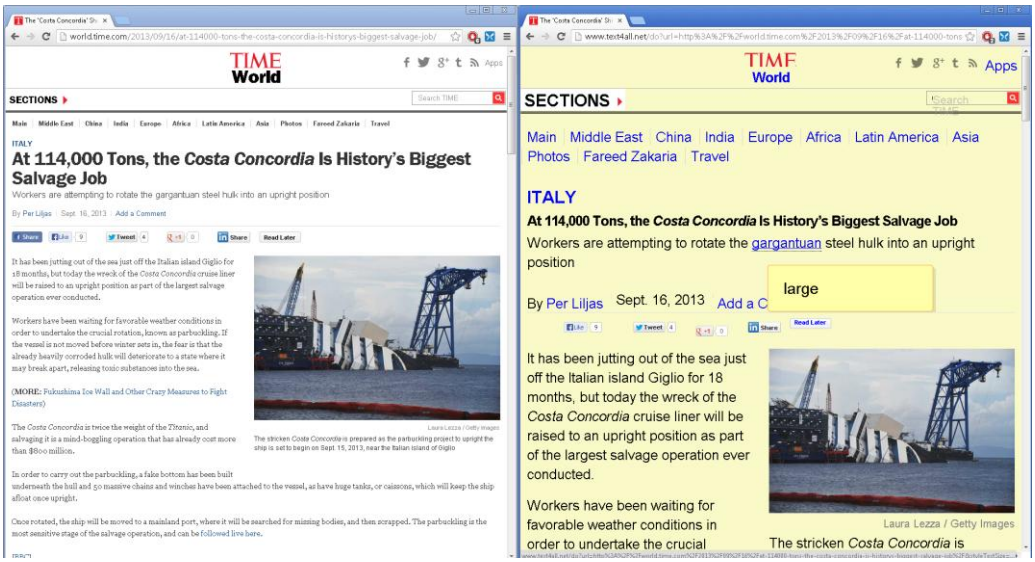

Figure 6.7 Original web page vs. DysWebxia adapted web page

text4all dysWebxia was included in a recent study [103] that was comparing reading tools for persons with dyslexia, and it was among the few that completed all required facilitates for improving the reading experience for readers with dyslexia.

More details about the functionality such as HTML parsing, and others can be found in the UML diagrams and the associated explanations from Appendix C.

6.5 Conclusions

This chapter presents techniques and tools used to adapt the look of the text from existing web site. The resulting tools are mostly useful for users with dyslexia or low vision, but can also help users with or without print disabilities. Based on the user studies done by the collaborators on the techniques implemented here, these tools prove to increase text accessibility and readability. The service designed for

customization (*text4all customizer*) has not yet been evaluated, while the service designed for users with dyslexia (*text4all dyswebxia*) was part of a study evaluating reading tools for users with dyslexia, and was among the few to fulfill all required capabilities. [103] Also *text4all dyswebxia* exemplifies the impact and usefulness of combining adaptations at both presentation level and language level. At the time of writing this thesis both services for adapting text at presentation level (*text4all Customizer* and *text4all DysWebxia*) are actively being used in real world.

7 Tools design considerations

The tools designed in this research take into consideration several important and novel aspects. Most important aspect is that the tools were design having in mind the big picture of text accessibility, and they fit into model proposed by the author for universal text access. This model and the exact positioning of the developed tools is presented in this chapter together with some general aspects of the model. Other design consideration like interaction with the tools via URLs are presented and exemplified.

7.1 Towards a Universal Accessibility Model for Text

The author explored techniques and implemented tools for adapting text at language level, focusing on specialized language level (*text4ll terminology interpreter)* and at presentation level (*text4all customizer*). Tools like *text4all dyswebxia* show that combining adaptation of text at multiple levels (language and presentation) can be useful for users with dyslexia. In this section the author explores how text limitations and adaptations can be classified on levels (also called *layers* in this work) in a universal accessibility model for textual information.
In the related work chapter the author already positioned existing techniques and technologies in this universal accessibility model. The parts of the model that have contributions from the author, and have been implemented in this research are highlighted in the universal view.

Text limitation layers
This research proposes and uses a layered accessibility model. The following main limitations there were identified and are represented in the model:

- Media format (printed text, digital text , audio text)
- Language (foreign languages)
- Specialized language (containing advanced terminology)

The author didn't focus on the media format and language level, since there is already substantial research and high quality tools. However, the author integrated some existing tools from these levels (like Google Translate for language level) in order to integrate translation facilities into the tools. Figure 7.1 illustrates the main layers with the main accessibility limitations.

Another important layer: *interaction layer* comes into play when implementing this model in tools that allow interaction with users. This layer takes into consideration modalities of interacting with the textual content in any format (digital, printed or audio). Since this research focuses on text from the web, only interaction with raw text or HTML text has been studied here. The model containing the interaction level is presented in figure 7.2.

Figure 7.1 Textual information accessibility limitation layers

One can see that the interaction layer is strongly coupled with the media format layer. Before focusing on the interaction of text from the web (HTML), that has been the focus of this study, the general distribution of interaction input/output, for each text media type is presented:

- Digital Text media format:
  - Input / Output:
    - **HTML via browser**
    - Other structured or linked data
    - **Raw text**
    - Text documents

- Printed text:
  - Input
    - Digital Image file, PDF
    - Printed text via scanner
  - Output
    - Digital Image file, PDF
    - Printed text via printer

- Audio test:
  - Input
    - Audio file
    - Microphone / Telephone
  - Output
    - Audio file
    - Speaker
    - Telephone

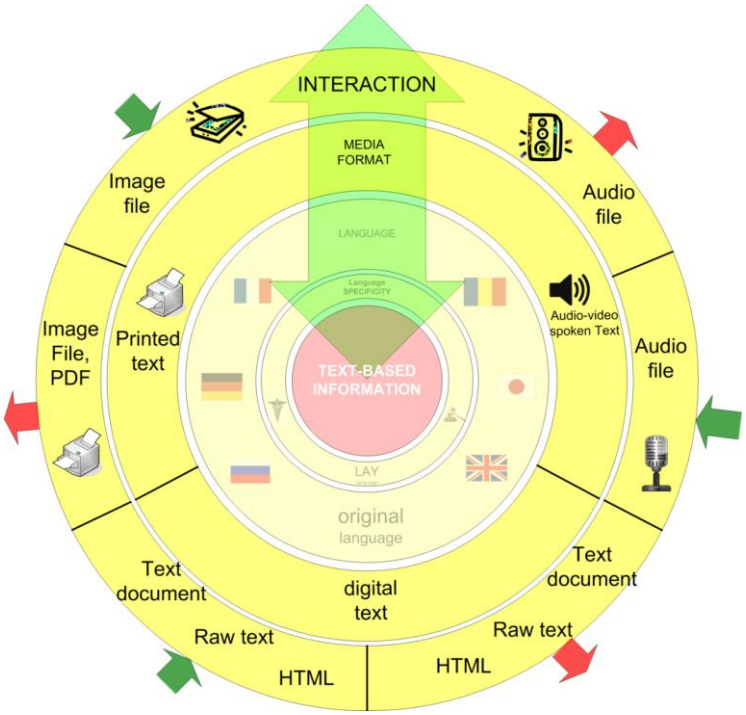

Figure 7.2 Universal Accessibility Model with Interaction Layer

The combined adaptation of text, from printed text to digital, from original language to another one and from digital to audio text has been explored by the author in a project where a multilingual printed text reader tool was developed [98]. This tool was reading printed text by using a scanner or a web cam, convert it digital text using optical character recognition techniques, enabled translation by the use of machine translation tool (Google Translate API) and synthesized spoken text using text to speech techniques. The tool was mainly useful for persons with visual disabilities, but could also accommodate other use cases. One of the limitations of the tool is that it was developed as a desktop application, making it work only on some specific operating systems (Windows, since it was developed in .NET), thus reducing its availability.

Since the author is dedicated to making tools that are highly available and work directly on the web, efforts were made to migrate all this process on the cloud. The effort was challenged at the interaction level, by the limited interaction capabilities of the web browsers with local devices. The main reason behind these limitations is the lack of standardization for accessing local resources from web pages. However, these issues are now analyzed by comities responsible for web standards (part of W3C) and several dedicated standards are under work, like the Media Capture and Streams standard, a draft version being available at [99], and Devices APIs requirements accessible at [100], the last one being at a mature version.

When the major web browsers will be compliant to these standards the task of implementing such a cloud based service and consume it online, directly from the browser, without the need of installing dedicated interaction software, will be much easier to do.

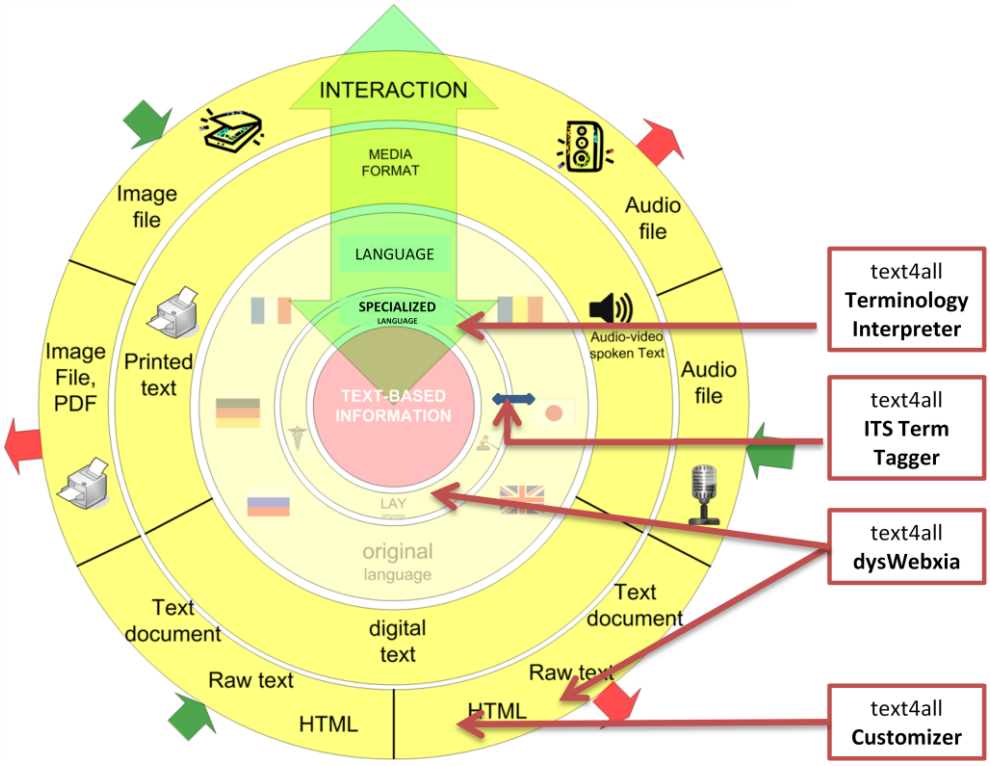

Figure 7.3 Tools (server side) implemented in this research, and their location in the universal text accessibility model

The only parts of the model that were implemented as cloud based services in this research are the adaptations of text at specialized language layer, and the adaptation of text. The tools designed and developed in this research are positioned into the model of universal text accessibility as illustrated in Figure 7.3.

Although not implemented and not validated, it is the belief of the author that a service that would implement all or most of the parts of this model (on the server side), and will allow accessible and usable interaction directly from the browser, would accommodate the need of most users with disabilities related to text accessibility. Furthermore, such a service will also benefit other users, by allowing processes like:

- Multilingual printed text reader
- Speech to speech translation
- Speech to speech specialized text adaptation
- Speech to speech text simplification
- Print to print translation
- Multilingual audio/video captioning
- …

7.2 Tools - user interaction

Another aspect that was carefully analyzed while designing and developing the tools is the user interaction. One way, let's say the classic one, for interacting with the web service is by using the Graphical User Interface (GUI). Another interaction option considered by the author is the use of URLs or URIs [101] that the user enters in the web address field of web browsers.
Since the part of interacting with the tool by the use of the GUI has been presented already in all the sections presenting the tools, the author will focus here on presenting the other interaction option, via URLs.

7.2.1 Interaction based on URLs

Address bar in web browsers is the place where the user enters the domain of the web page (the URL) in order to access that specific web page. The author exploits this place for doing more than just entering the target URL, but also for a) easily adapting an existing web site and b) for changing adaptation options.
Using the URL for more than just entering the target web page, but also for enabling and enhancing web page interaction is not a novel idea. The aspects of having URLs as UI, and also elements of good URL design, have been discussed by Nielsen in [102]. A similar accessibility related project, Web Adjuster [86] done by Silas Brown, uses domain rewriting in order to facilitate an easier web page transcoding, operable directly from the URL and with benefits for the transcoding process (it is easier to adapt links within the site, some working without being changed).

7.2.1.1 text4all URL design

The tools done in this research (published on the text4all.net domain) give special attention to URL design, making sure the domain names, query parameters and values are "hackable", being easy to remember and to guess. This allows users to change the customization settings, language settings or other settings directly from the URL.

For example, the ***terminology interpreter*** tool uses the following domain and query parameters:

http://www.text4all.net/do?langOp=medToLay&lang=RO&fuzzyLevel=2&url=http://medlive.hotnews.ro/in-ce-consta-terapia-cu-unde-de-soc-si-ce-beneficii-are-dr-alin-popescu-medic-primar-medicina-sportiva-discuta-online-cu-cititorii-vineri-de-la-ora-13-00.html

The parameters with their values are:

langOp=medToLay: indicating that the language operation to be applied is adaptation from medical to lay language.

lang=RO: indicating that the input language is Romanian

fuzzyLevel=2: indicating the maximum edit distance for terminology matching

url=http:// … : indicating the web page that has to be adapted

One can see that the query parameters together with values were designed to be intuitive so users can understand and easily change them. Also the web service is error tolerant with user input, treating query parameters and values case insensitive, and allowing variants for the names (for example *langOp* is also recognized in forms like: *langop*, *languageOperation*).

Another example of query parameters and values design, used for text4all DysWebxia is presented bellow.

http://www.text4all.net/do?op=dyswebxia2&styleTextSize=22&styleTextColor=000000&styleBackground=FAFAC8&styleTextFont=Helvetica&layout=original&lang=EN&langOp1=DifficultWords&langOp2=replaceNumbers&url=http%3A%2F%2Fen.wikipedia.org%2Fwiki%2FHistory_of_romania

The query parameters and values are generated automatically when using the user interface, so users have some friendly interfaces as alternatives.

7.2.1.2 URL Shortcuts

In order to improve the interaction experience of the users with the service, URL shortcuts for adapting web pages have been designed. Using these shortcuts make the adaptation process more transparent and the mediator tool less visible, helping the user to focus on the target web page rather than the mediator tool. When using URL shortcuts the process of adapting a web page is different from the other cases because the user starts the adaptation process directly from the target web page.

Let's consider an example where a user with dyslexia reading a Wikipedia article whishes to adapt the page in order to make it more readable. For this he has to go to the web address of the web page, and type or paste at the beginning of the web address the URL shortcut: "***text4all.net/d/***" where "d" comes from dyslexia. When

he loads this new address the user gets the web page adapted to the default DysWebxia settings. Figure 7.4 shows the URL of the original web page while figure 7.5 presents the URL of the adapted web page.

Figure 7.4 Original web page (with original URL)

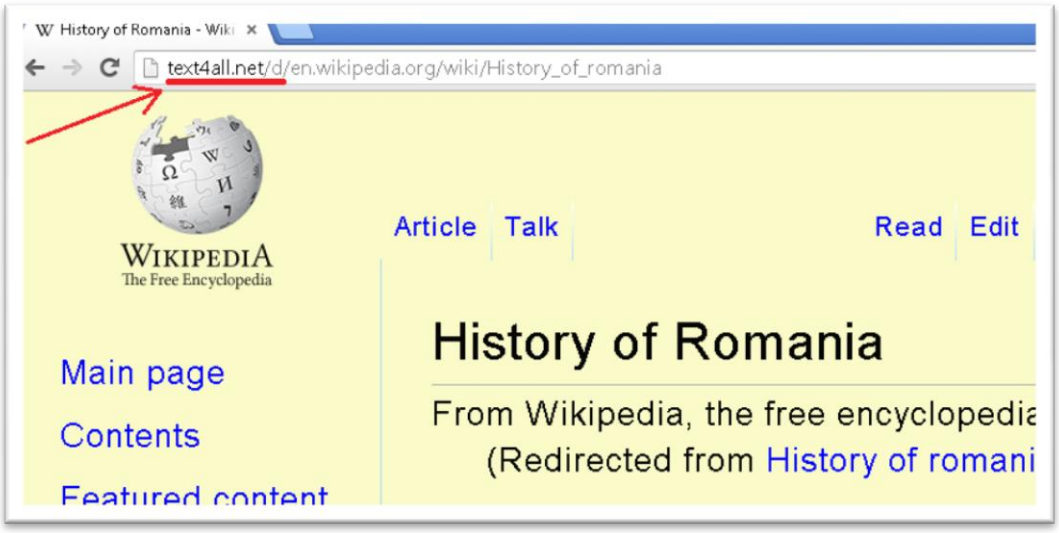

Figure 7.5 Adapted web page (having the URL prefixed with "text4all.net/d/")

Other URL shortcut is designed for *Terminology Interpreter* service.

This type of interaction has not yet been tested with user studies, but the author believes that this alternative interaction type can be benefic in some use cases for users. Although this interaction method implies a certain memory load, because the user has to memorize the URL shortcuts, once this is done, adapting a web page becomes just a simple URL adjustment (or command, or copy/paste operation).

8 Conclusion

This thesis presented several techniques and tools designed and confirmed to improve accessibility and understanding of text for lay users. Furthermore the levels of text limitations are presented together in a universal view, exploring the big image of text accessibility, and interaction between access layers.

This research gives much attention to language level access limitations caused by the presence of specialized language. Here the author goes from language study and theory to applications, and from applications to implications. The author presents a possible solution applied to medical language, going through all stages, from language analysis, techniques proposal and evaluation to actual application development, optimization and user evaluation. The performance of the resulting system can be improved along with its use by iteratively updating the matching model and increasing the incorrect matching repository. Fuzzy matching techniques are preferred for dealing with derivate terminology encountered in natural language. Automated methods combined with supervised processes are used to optimize the performance of the tools. Fuzzy matching validation done by humans is also included, and ways to automatically filter and manage such input are presented.

Also this thesis explores way of making text more accessible by adapting the look of the text, end by enabling customization of existing web pages in order to fit the needs of more user types.

All the tools presented in this thesis were designed with much consideration to usability and availability of the tools. The tools were implemented as web mediators running on the cloud, making them run anywhere, directly into the browser.

To increase the chances of making an impact in the real world, most of the tools resulted from this research were published online by the author under the name *text4all*, at the domain with the same name, available at [106], and some of the services are being actively used by readers on the web.

In the opinion of the author, the objectives set for this thesis were fulfilled. Methods and tools for improving text understanding for domain-specific languages were validated by user tests, the *FuzzyHashMap* data structure designed by the author proved to work efficiently in this research project and also in other projects done by other researchers; most of the resulting tools were published online and are used in real world.

8.1 Contributions

The **major contributions** of this thesis are:

- Elaboration of techniques (precise fuzzy matching by the use of matching model, custom hashing pattern, user feedback validation and crowd sourced based matching validation) and their implementation, designed to improve message understanding by explaining terminology in specialized languages.
- Created the *FuzzyHashMap* data structure that enables fast and accurate fuzzy string matching. This data structure has already been used by other researches in projects like genome anchoring, search engines or question generators.

Other contributions, including the resulted tools and studies performed through this work are presented in the following:

- The design of a model for universal text accessibility (all the developed tools were localized within this model)

Studies:

- Study and statistics of terminology occurring in canonical form compared to derived form for Romanian and English medical language
- Study of fuzzy string matching validation done by humans. Studies were done by questioners distributed through a social network and by Human Intelligence Tasks performed on Amazon Mechanical Turk crowdsourcing platform. Studies revealed two important aspects on human based validation for terminology matching: (i) using trap questions for validating matches responded by a single user proved to be inefficient and (ii) comparing the response of multiple users and looking for response agreement is efficient (even without the use of trap questions)

- Study on the efficiency of medical terminology annotation (explanation) over message understanding and eventually patient empowerment

Language level specific methods and associated tools:

- ***text4all terminology interpreter***: This tool, working completely online, can take the address of a web site as input and identify and explain the terminology. The terminology explanation can be presented in multiple ways, the most appreciated modes by users being on click over the term or tooltip on hover. User studies done by the author on the impact of showing terminology explanation have validated the efficiency of such methods and tools.
- ***text4all term analysis***: takes the address of a web page as input and returns statistics of terminology usage in the target web page. While this tool has been created to be used in this research in order to better understand how terminology occurs in natural language in specific domains, it can be useful outside of this work, for linguistic analysis. Currently it only does the analysis dependent on the existing terminology dictionaries. As

future work the author wants to enable the addition of external dictionaries (import dictionary function) in order to make if usable for any language/domain.

- ***text4all ITStagger***: takes as input raw text or HTML and annotates the terminology using Internationalization Tag Set (ITS 2.0) in order to improve translation of specialized language. While developing this tool, the author proposed and discussed with the W3C working group responsible for ITS 2.0 the necessity of adding an additional attribute for terminology occurring in derivate form that will mention the term in the canonical form (lemma). By the time of writing this thesis, this new attribute is considered as a proposal and is still discussed in the working group.

Combined levels or presentation level tools:

- ***text4all dysWebxia***: a service designed for users with dyslexia, that takes a web page address as input and adapts the look and aspects of the language in order to make the web page more dyslexia friendly. This project is based on the DysWebxia 2.0 model (done by the collaborator Luz Rello), which was created after extensive user studies, mostly using eye tracking techniques. The tool and the model also fit into the universal text accessibility model presented in this research. The tool has been classified as fulfilling most of the required adaptation and customization needed for users with dyslexia, making it a good alternative for such readers.
- ***text4all Customizer***: a service taking as input the address of a web page and enables the user to adjust the look of the web page; mostly useful for users with low vision or other print disabilities. Although no user evaluation of the tool has been done, the service shows high potential for readers on the web by exploiting the adaptation needs described by users in other studies.

8.2 Final considerations, aims and goals

Better access to information, empowering users (with or without disabilities) to access textual information easier and empowering patients to better understand medical information are the fundamental goals of this work. A series of methods and tools were implemented to achieve, at least partially, these aims. The service for terminology annotation has already a satisfactory precision. More can be done for this, and since crowdsourcing seems to be a promising solution, by providing the integration of human intelligence in programmable algorithms, as future work, the authors intends to complete the integration of AMT API with the terminology annotating services. Further goals on this are to achieve nearly real time validation in order to have immediate or on-demand matching precision improvement, and to explore the recently appeared programming languages dedicated to *programming the crowd*.

One of the main issues when it comes to terminology recognition is the difficulty of finding and integrating dictionaries into the proposed system. As a solution to this, the author plans to integrate the *EuroTermBank API* into the system, opening the door for importing and updating thousands of dictionaries currently existing in *EuroTermBank* [77]. This will have a very big impact, due to the wide range of

domains and languages that will be available for the suite of services resulted from this work, including *terminology interpreter*, *ITS tagger*, *term analyzer*.

This thesis provides a new view over text accessibility by proposing a model for universal accessibility. The problem with fully implementing the entire model is the current limitations that appear when it comes to interaction between the web and the local devices. As web technologies, browsers and especially standards are advancing and more power is given to web applications, including access to local devices, online instruments for text transformation will be able to combine the use of local text input devices (such as scanners and microphones) with the availability of the cloud, and provide highly interactive online text adaptation systems.

A future goal is to have the entire model implemented, so that text can be transformed at all needed levels (specialized language, language, media format and presentation language). It is the belief of the author that such instruments would revolutionize text manipulation, enabling and enhancing the experience of consuming information regardless of users' abilities or disabilities, independent of text representation and independent on the platform where the service is being used. That is what the author considers the ultimate user empowerment and independence in accessing textual information.

References

[1] W3C – World Wide Web Consortium; Web page: http://www.w3.org/ Accessed on 20 Sep 2013

[2] Executive Agency for Health and Consumers, *The European Health Literacy Survey* 2009 – 2012 (COMPARATIVE REPORT ON HEALTH LITERACY IN EIGHT EU MEMBER STATES) accessible at: http://ec.europa.eu/eahc/documents/news/Comparative_report_on_health_literacy_in_eight_EU_member_states.pdf Accessed on 25 Nov 2013

[3] W3C Web Accessibility Initiative: *Web Content Accessibility Guidelines*, available at: http://www.w3.org/WAI/intro/wcag Accessed on Sept 2013

[4] Web Accessibility National Transition Strategy; The Australian Government's adoption and implementation of Web Content Accessibility Guidelines version 2.0 (WCAG 2.0); http://www.finance.gov.au/publications/wcag-2-implementation/index.html

[5] Conway V. L., Website accessibility in Australia and the Australian Government's National Transition Strategy; W4A '11**:** Proceedings of the International Cross-Disciplinary Conference on Web Accessibility; March 2011

[6] W3C Consortium, Internationalization Tag Set (ITS); Available at: http://www.w3.org/TR/its/ Accessed on 5 June 2013

[7] *Google Translate* – Statistical Machine Translation tool; available at http://translate.google.com/ Accessed on 20 August 2013

[8] *Bing Translator* - Statistical Machine Translation tool; available at http://www.bing.com/translator Accessed on 20 August 2013

[9] *SDL Automated Translation* – machine translation tool available at: http://www.sdl.com/products/automated-translation/ Accessed on 20 August 2013

[10] Von Ahn, Luis. "*Duolingo–free language education fort he world*." URL: http://www.duolingo.com/ Accessed on 17 March 2013

[11] Mārcis Pinnis, Tatiana Gornostay, Pēteris Ņikiforovs; *ITS 2.0 Enriched Terminology Annotation Use Case*, http://taws.tilde.com/ W3C workshop: Making the Multilingual Web Work; Rome, Italy; March 2013; Poster available at: http://www.w3.org/International/multilingualweb/rome/posters/mlw-lt_rome2013poster-11.pdf

[12] M. Starlander, P. Bouillon, N. Chatzichrisafis, M. Santaholma, M. Rayner, B.A. Hockey, H. Isahara, K. Kanzaki, Y. Nakao (2005). Practising Controlled Language through a Help System integrated into the Medical Speech Translation System (MedSLT). In Proceedings of the MT Summit X, 12-16 September, 2005, Phuket, Thailand.

[13] M. Rayner, P. Bouillon, N. Chatzichrisafis, M. Santaholma, M. Starlander, B. A. Hockey, Y. Nakao, H. Isahara, K. Kanzaki (2006). MedSLT: A Limited-Domain Unidirectional Grammar-Based Medical Speech Translator. In Proceedings of First International Workshop on Medical Speech Translation, HLT-NAACL, June 9, 2006, New York.

[14] P. Bouillon, N. Chatzichrisafis, S. Halimi, B A Hockey, H. Isahara, K. Kanzaki, Y. Nakao, B. Novellas Vall, M. Rayner, M. Santaholma, and M. Starlander (2007). MedSLT: A Multi-Lingual Grammar-Based Medical Speech Translator. In Proceedings of First International Workshop on Intercultural Collaboration, IWIC2007, January 25-26, Kyoto, Japan

[15] M. Rayner, P. Bouillon, J. Brotanek, G. Flores, S. Halimi, B. A. Hockey, H. Isahara, K. Kanzaki, E. Kron, Y. Nakao, M. Santaholma, M. Starlander, N. Tsourakis (2008). The 2008 MedSLT System. In proceedings of Coling 2008 Workshop on Speech Processing for Safety Critical Translation and Pervasive Applications, Manchester, UK

[16] P. Bouillon, G. Flores, M. Georgescul, S. Halimi, B. A. Hockey, H. Isahara, K. Kanzaki, Y. Nakao, M. Rayner, M. Santaholma, M. Starlander, N. Tsourakis (2008). Many-to-Many Multilingual Medical Speech Translation on a PDA. In proceedings of The Eighth Conference of the Association for Machine Translation in the Americas. Waikiki, Hawaii.

[17] Zeng-Treitler Q, Kim H, Rosemblat G, Keselman A., Can multilingual machine translation help make medical record content more comprehensible to patients?, Stud Health Technol Inform. 2010;160(Pt 1):73-7.

[18] Soergel D, Tse T, Slaughter L., Helping healthcare consumers understand: an "interpretive layer" for finding and making sense of medical information. *Stud Health Technol Inform.* 107(Pt 2):931-5. PubMed; 2004;

[19] Kandula, S., D. Curtis, and Q. Zeng-Treitler. A Semantic and Syntactic Text Simplification Tool for Health Content. in AMIA Annu Symp Proc. 2010.

[20] Zeng-Treitler Q, Goryachev S, Kim H, Keselman A, Rosendale D., Making texts in electronic health records comprehensible to consumers: a prototype translator. AMIA Annu Symp Proc. 2007 Oct 11:846-50.

[21] Zeng, Qing T., and Tony Tse. "Exploring and developing consumer health vocabularies." *Journal of the American Medical Informatics Association* 13.1 (2006): 24-29.

[22] Open Access, Collaborative Consumer Health Vocabulary website: http://www.consumerhealthvocab.org/

[23] Brown, S. and Robinson, P. (2001) 'A World Wide Web Mediator for Users with Low Vision.' CHI 2001 Workshop No. 14.

[24] Myers, W. (1998) BETSIE (BBC Education Text to SpeechInternet Enhancer). available at: http://www.bbc.co.uk/education/betsie/.

[25] Hanson, V. and Richards, J. (2003) 'A web accessibility service: update and findings' SIGACCESS Access. Comput. 77-78 (September 2003), 169-176.

[26] Bigham, J., Prince C. and Ladner, R. (2008) 'WebAnywhere: A Screen Reader On-the-Go', In Proceedings of the 2nd Cross-Disciplinary Conference on Web Accessibility. Beijing, China, 2008

[27] Hironobu Takagi and Chieko Asakawa, "Transcoding Proxy for Nonvisual Web Access," Proceedings of ACM ASSETS 2000, pp. 164-171, Nov. 2000.

[28] Oracle corp., Java HashMap datastructure: http://docs.oracle.com/javase/6/docs/api/java/util/HashMap.html

[29] Topac, V., "Efficient fuzzy search enabled hash map," Soft Computing Applications (SOFA), 2010 4th International Workshop on , vol., no., pp.39,44, 15-17 July 2010

[30] Healy, John, and Desmond Chambers. "Fast and Accurate Genome Anchoring Using Fuzzy Hash Maps." In *5th International Conference on Practical Applications of Computational Biology & Bioinformatics* (PACBB 2011), pp. 149-156. Springer Berlin Heidelberg, 2011.

[31] Healy, John, and Desmond Chambers. "De Novo Draft Genome Assembly Using Fuzzy K-mers." In BIOTECHNO 2011, *The Third International Conference on Bioinformatics, Biocomputational Systems and Biotechnologies*, pp. 104-109. 2011.

[32] G Navarro, RA Baeza-Yates, E Sutinen, J Tarhio, Indexing methods for approximate string matching, IEEE Data Eng. Bull. 24 (4), 19-27; 2011

[33] C. Galvez, F. Moya-Anegón; Approximate Personal Name-Matching Through Finite-State Graphs; Journal of the American Society for Information, 2007

[34] S Ji, G Li, C Li, J Feng; Efficient type-ahead search on relational data: a TASTIER approach; Proceedings of the 35th SIGMOD international conference on Management of data, 2009

[35] M Hadjieleftheriou, C Li; Efficient approximate search on string collections; Proceedings of the VLDB Endowment, 2009

[36] A. Urrutia; L. Tineo; C. Gonzalez; FSQL and SQLf: Towards a Standard in Fuzzy Databases; Handbook of Research on Fuzzy Information Processing in Databases; Pages: 270-298; 2008

[37] Giegerich, R.; Kurtz, S. (1997), "From Ukkonen to McCreight and Weiner: A Unifying View of Linear-Time Suffix Tree Construction", Algorithmica 19 (3): 331–353, doi:10.1007/PL00009177.

[38] Dyslexia International, About Dyslexia – Incidence, Available at: http://www.dyslexia-international.org/Educational%20Authorities/About%20dyslexia%20ea Accessed on 10 Dec 2013

[39] X. Yang, B. Wang, and C. Li. Cost-based variable-length-gram selection for string collections to support approximate queries efficiently. In SIGMOD, 2008

[40] FuzzyHashMap open source project web address: http://fuzzyhashmap.sourceforge.net/

[41] Russell, R., and M. Odell. "Soundex." *US Patent* 1, 1918.

[42] V. Levenshtein. Binary codes capable of correcting spurious insertions and deletions of ones. Probl. Inf. Transmission 1, 8-17. 1965

[43] R. W. Hamming; Error Detectiong and Error Correcting Codes; The Bell System Technical Journal, vol xxIx, No. 2; 1950

[44] Guoliang Li, Shengyue Ji, Chen Li, Jianhua Feng; Efficient type-ahead search on relational data: a TASTIER approach; International Conference on Management of Data, Rhode Island, USA, 2009

[45] Weaver, Warren. "*Translation*". In Locke, W.N.; Booth, A.D. Machine Translation of Languages: Fourteen Essays. Cambridge, MA: MIT Press. 1949

[46] Google N-gram service; Available at: https://books.google.com/ngrams; Accessed on 25 Sep 2013

[47] World Health Organization, Visual Impairment and Blindness; updated on October 2013; Available at: http://www.who.int/mediacentre/factsheets/fs282/en/ Accessed on 03 Dec 2013

[48] Gale, W., K. Church, and D. Yarowsky, "A Method for Disambiguating Word Senses in a Large Corpus," Computers and the Humanities, 26, pp 415-439, 1992.

[49] Medical advices website, 2006, Article in Romanian about thyroid cancer; accessed on Jan 10 2013; http://www.sfatulmedicului.ro/Afectiunile-tiroidei/cancerul-tiroidian_774

[50] *text4all Terminology Analysis and Stats* tool, available at http://www.text4all.net/termanalysis.html; Accessed on 11 Sep 2013.

[51] Precision and recall article and figure, Available at: http://en.wikipedia.org/wiki/Precision_and_recall; Accessed on 7 Oct 2013;

[52] Van Rijsbergen, C. J. (1979). *Information Retrieval* (2nd ed.). Butterworth.

[53] Amazon Mechanical Turk crowdsourcing platform; available at: https://www.mturk.com Accessed on 16 Aug 2013

[54] Google Docs – part of Google Drive; available at: https://drive.google.com/ Accessed on 28 Oct 2013

[55] Von Ahn, Luis, and Laura Dabbish. "Labeling images with a computer game." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 319-326. ACM, 2004.

[56] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast---but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (EMNLP '08). Association for Computational Linguistics, Stroudsburg, PA, USA, 254-263

[57] Anna Rumshisky. 2011. Crowdsourcing word sense definition. In *Proceedings of the 5th Linguistic Annotation Workshop* (LAW V '11). Association for Computational Linguistics, Stroudsburg, PA, USA, 74-81.

[58] Rumshisky, A., N. Botchan, S. Kushkuley, and J. Pustejovsky (2012). Word sense inventories by nonexperts. Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey. European Language Resources Association (ELRA). 2012

[59] Greg Little, Programming with Human Computation, PhD Thesis, MIT 2011; Available at: http://groups.csail.mit.edu/uid/other-pubs/glittle-thesis.pdf Accessed on 8 Aug 2013

[60] Greg Little, Lydia B. Chilton, Max Goldman, and Robert C. Miller. *TurKit: human computation algorithms on mechanical turk.* In Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 57-66. 2010.

[61] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom Yeh. *VizWiz: nearly real-time answers to visual questions.* In Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 333-342. 2010

[62] Funnell, M, Patient Empowerment, in *Critical Care Nursing Quarterl*y: April/May/June 2004 - Volume 27 - Issue 2, p. 201-204

[63] Aujoulat, I., Marcolongo,R., Bonadiman L., Deccache, A., Reconsidering patient empowerment in chronic illness: A critique of models of self-efficacy and bodily control, *Social Science & Medicine*, Volume 66, Issue 5, March 2008, pages 1228-1239.

[64] Wildner, M., Development of a questionnaire for quantitative assessment in the field of health and human rights, *Social Science & Medicine*, Volume 55, Issue 10, November 2002, pages 1725-1744.

[65] Trudi Miller; Gondy Leroy; Samir Chatterjee; Jie Fan; Brian Thoms , A Classifier to Evaluate Language Specificity of Medical Documents, in HICSS 2007. *40th Annual Hawaii International Conference on System Sciences*, 2007

[66] International Medical Interpreters Association; accessed on 10 Jan 2013: http://www.imiaweb.org/default.asp

[67] Keselman, Alla, et al. "Developing informatics tools and strategies for consumer-centered health communication." *Journal of the American Medical Informatics Association* 15.4 (2008): 473-483.

[68] Raja U, Mitchell T, Day T, Hardin JM., Text mining in healthcare. Applications and opportunities, J Healthc Inf Manag. 2008 Summer;22(3):52-6.

[69] Cimino JJ.; High-quality, standard, controlled healthcare terminologies come of age; Methods Inf Med. 2011;50(2):101-4. Epub 2011 Mar 17.

[70] Keizer N. F., Abu-Hanna A., Zwetsloot-Schonk J. H. M. ; Understanding Terminological Systems I: Terminology and Typology. Methods of Information in Medicine , Issue 1 2000 (Vol 39): 16-21

[71] Text4all - Terminology Interpreter project, Accessed on 10 Jan 2013: http://www.text4all.net/interpreter.jsp

[72] Romanian web site with medical information, interviews and discussions; accessed on 10 Jan 2013 http://medlive.hotnews.ro/

[73] Medlive website – "get in touch with doctors" section, 2011; An interview in Romanian language with dr. Daniel Ciurdariu about thyroid cancer surgery; accessed on 10 Jan 2013; http://medlive.hotnews.ro/video-dr-daniel-ciurdariu-medic-chirurg-in-cazul-cancerului-de-tiroida-operatia-este-mai-dificila-decat-cea-pentru-un-nodul-simplu-sau-pentru-gusa-interventia-este-supusa-accidentelor-si-complica.html

[74] Puscoci S., Stoicu-Tivadar L, Integrated tele-assistance platform – TELEASIS, in The Second IFAC Symposium on Telematics Applications TA 2010, Timişoara, Romania, 5-8 October, 2010: 97-102.

[75] W3C Consortium, Internationalization Tag Set (ITS) Version 2.0; Available at: http://www.w3.org/TR/its20/ Accessed on 10 November 2013

[76] The Jožef Stefan Institute, *Text Analytics in ITS 2.0: Annotation of Named Entities*; W3C workshop: Making the Multilingual Web Work; Rome, Italy; March 2013; Poster available at: http://www.w3.org/International/multilingualweb/rome/posters/mlw-lt_rome2013poster-09.pdf

[77] EUROTERMBANK CONSORTIUM, *EuroTermBank* project; available at: http://www.eurotermbank.com/ Accessed on 14 Oct 2013

[78] Henriksen, Lina, Claus Povlsen, and Andrejs Vasiljevs. "EuroTermBank–a Terminology Resource based on Best Practice." Proceedings of LREC 2006, the 5th International Conference on Language Resources and Evaluation. 2006.

[79] *The definition of "print disabled"?*. Reading Rights Coalition. Retrieved 25 October 2013. http://www.readingrights.org/definition-print-disabled

[80] Text Customization for Readability Online Symposium 19 November 2012; Available at: http://www.w3.org/WAI/RD/2012/text-customization/; Accessed on 03 Feb 2013

[81] Henry, S.L. Developing Text customization Functionality Requirements of PDF Reader and Other User Agents. In: eds. *Proceedings of Computers Helping People with Special Needs, 13th International Conference, ICCHP 2012, Linz, Austria, July 11-13,*

[82] Santana, V.F., Oliveira, R., Almeida, L.D.A.; Baranauskas, M.C.C. *Web Accessibility and People with Dyslexia: A Survey on Techniques and Guidelines*, Proceedings of the Inter-national Cross-Disciplinary Conference on Web Accessibility (W4A '12). New York, NY, USA, 2012

[83] Anthony Lee, *Accessibility issues due to sub-pixel rendering* ," in Text Customization for Readability, W3C Online Symposium, Nov. 2012. Available from: http://www.w3.org/WAI/RD/2012/text-customization/p7 [retrieved: August, 2013].

[84] S. L. Henry, "CSS for Readability: Analysis of user style sheets to inform understanding users' text customization needs," in Text Customization for Readability, W3C Online Symposium, Nov. 2012. Available from: http://www.w3.org/WAI/RD/2012/text-customization/r14 [retrieved: August, 2013].

[85] Readability reading service and browser extensions. Available at: https://www.readability.com/ Accessed on Oct 20 2013

[86] Brown, S. and Robinson, P. (2001) 'A World Wide Web Mediator for Users with Low Vision.' CHI 2001 Workshop No. 14.; 2001

[87] Cascade Styles Sheets (CSS) overview: http://www.w3.org/Style/CSS/Overview.en.html Accessed on 21 June 2013;

[88] Web page in English language about Diabetes http://en.wikipedia.org/wiki/Diabetes_mellitus Accessed on Oct 20 2013

[89] Silas S. Brown; Stylesheets for low vision; available at: http://people.ds.cam.ac.uk/ssb22/css/ Accessed on 11 Oct 2012

[90] Rello, Luz; Kanvinde, Gaurang & Baeza-Yates, Ricardo., Layout Guidelines for Web Text and a Web Service to Improve Accessibility for Dyslexics. W4A 2012: The

9th International Cross Disciplinary Conference on Web Accessibility. Lyon, France, 16-17 April. 2012

[91] Rello, Luz. *DysWebxia: A Model to Improve Accessibility of the Textual Web for Dyslexic Users*. Newsletter ACM SIGACCESS Accessibility and Computing. January 2012.

[92] Rello, Luz & Marcos, Mari-Carmen. *An Eye Tracking Study on Text Customization for User Performance and Preference*. LA-WEB 2012: The 8th edition of the Latin American Web Congress. Cartagena, Colombia, 25-27 October. 2012

[93] Rello, Luz & Baeza-Yates, Ricardo. *Optimal Colors to Improve Readability for People with Dyslexia.* Text Customization for Readability, W3C WAI RDWG Online Symposium, 19 November. 2012.

[94] Rello, Luz; Martin Pielot; Mari-Carmen Marcos & Roberto Carlini. *Size Matters (Spacing not): 18 Points for a Dyslexic-friendly Wikipedia,* W4A 2013: The 10th International Cross Disciplinary Conference on Web Accessibility. May 13-15, 2013, Rio de Janeiro, Brazil.

[95] Rello, Luz; Ricardo Baeza-Yates; Horacio Saggion; Stefan Bott; Roberto Carlini; Clara Bayarri; Azuki Gorriz; Saurabh Gupta; Gaurang Kanvinde & Vasile Topac. *DysWebxia 2.0! More Accessible Text for People with Dyslexia,* W4A 2013 - The Paciello Accessibility Challenge: The 10th International Cross Disciplinary Conference on Web Accessibility. May 13-15, 2013, Rio de Janeiro, Brazil.

[96] Rello, Luz; Ricardo Baeza-Yates; Laura Dempere & Horacio Saggion. *Frequent words improve readability and short words improve understandability for people with dyslexia.* INTERACT 2013: 14th IFIP TC13 Conference on Human-Computer Interaction. Cape Town, South Africa, 2013.

[97] Rello, Luz; Susana Bautista; Ricardo Baeza-Yates; Pablo Gervás, Raquel Hervás & Horacio Saggion. *One half or 50%? An eye-tracking study of number representation readability* INTERACT 2013: 14th IFIP TC13 Conference on Human-Computer Interaction. Cape Town, South Africa, 2013.

[98] Topac, V. Stoicu-Tivadar, V, *Software Architecture for Better Text-Based Infromation Accessibility,* AICT '09. Fifth Advanced International Conference on Telecommunications, Veneţia, Mai 2009.

[99] W3C Media Capture and Streams, W3C Working Draft 03 September 2013, http://www.w3.org/TR/mediacapture-streams/ Accessed on Nov 25 2013

[100] W3C Device APIs Requirements, W3C Working Group Note 15 October 2009 http://www.w3.org/TR/dap-api-reqs/ Accessed on Nov 25 2013

[101] T. Berners-Lee, R. Fielding, L. Masinter; *Uniform Resource Identifiers (URI): Generic Syntax;* (RFC 2396) http://www.ietf.org/rfc/rfc1738.txt August 1998;

[102] Jakob Nielsen, *URL as UI*, http://www.nngroup.com/articles/url-as-ui/ Accessed on Nov 30 2013

[103] Rello, Luz & Simone D. J. Barbosa. Do People with Dyslexia Need Special Reading Software?. INTERACT 2013 Workshop on Rethinking Universal Accessibility: A broader approach considering the digital gap. Cape Town, South Africa, 2013.

[104] Ogden, J., Branson, R., Bryett, A., Campbell, A., Febles, A., Ferguson, I., Lavender, H., Mizan, J., Simpson, R., Tayler, M.; What's in a name? An experimental study of patients' views of the impact and function of a diagnosis; Fam Pract. 2003 Jun; 2003; p:248-53.

[105] Jean-Baptiste Michel*, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, William Brockman, The Google Books Team, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden*. *Quantitative Analysis of Culture Using Millions of Digitized Books*. Science (Published online ahead of print: 12/16/2010)

[106] V.Topac, *text4all* set of tools for adapting text on the web. Available at www.text4all.net Accessed on 20 Nov 2013

[107] Stoicu-Tivadar, V., Stoicu-Tivadar, L., Topac, V., & Berian, D.. A WebService-based alarm solution in a telecare system. In *Applied Computational Intelligence and Informatics, 2009. SACI'09. 5th International Symposium on* (pp. 117-122). IEEE., May 2009

[108] Stoicu-Tivadar, V., Stoicu-Tivadar, L., Topac, V., & Berian, D. A WebService-based alarm solution in a telecare system. In *Applied Computational Intelligence and Informatics, 2009. SACI'09. 5th International Symposium on* (pp. 117-122). IEEE., May 2009

[109] Topac, V. and Stoicu-Tivadar, V.. "Patient empowerment by increasing information accessibility in a telecare system." *Studies in Health and Technology Informatics* 169: 681-685., 2011

[110] Topac, V., and Stoicu-Tivadar, V. Patient Empowerment by Increasing the Understanding of Medical Language for Lay Users., *Methods Inf Med*. journal 2013;52(5):454-62. doi: 10.3414/ME12-02-0006. Epub 2013 Sep 27.

[111] Topac, V. The development of a text customization tool for existing web sites. In *Text Customization for Readability Symposium*. W3C, November 2012.

[112] Topac, V., Stoicu-Tivadar, V, Evaluation of Terminology Labeling Impact over Readability, in *Easy-to-Read on the Web Symposium,* W3C, Dec 2012.

[113] MOHAMMED, Athraa Jasim; HUSNI, Husniza. Blogs Search Engine Using RSS Syndication and Fuzzy Parameters, *UACEE International Journal of Computer Science and its Applications* - Volume 2: Issue 3 [ISSN 2250 - 3765], Dec 2012

[114] *LoadImpact* web service for performing load tests. Available at http://loadimpact.com/ Accessed on: 02 Dec 2013

[115] Cleverdon, Cyril W., Jack Mills, and Michael Keen. "Factors determining the performance of indexing systems." (1966).

[116] Yesilada, Yeliz, Robert Stevens, Simon Harper, and Carole Goble. "Evaluating DANTE: Semantic transcoding for visually disabled users." ACM Transactions on Computer-Human Interaction (TOCHI) 14, no. 3 (2007): 14.

[117] Yesilada, Y., Harper, S., Goble, C., and Stevens, R. "Dante annotation and transformation of Web pages for visually impaired users". In Proceedings of the 13th International Conference on World Wide Web, 2004

[118] D. Lunn, S. Bechhofer, and S. Harper, "The SADIe Transcoding Platform," in W4A '08: Proceedings of the 2008 International Cross-Disciplinary Conference on Web Accessibility (W4A), New York, NY, USA, 2008, pp. 128-129.

.

Appendix

Appendix A.

Canonical vs. Derivate form of terminology analysis (revised) results:

This is an example of what *text4all term analysis* tool returns for the given web page. The bolded terms represent incorrect matching terms that were discovered by manual revision. Sample of analysis results for Romanian language:

URL: http://medlive.hotnews.ro/tumorile-suprafetei-oculare-dr-florentina-chitac-pashalidi-medic-primar-oftalmolog-discuta-online-cu-cititorii-miercuri-de-la-11-00.html
language: RO
fuzzy level: 2

Total
Total number of words: 11621
Number of ALL occurences of Terms: 150
Number of UNIQUE Terms identified: 82
Canonical terms
Number of All occurencies of CANONICAL Terms: 69
Number of UNIQUE CANONICAL Terms identified: 31
Fuzzy terms
Number of All occurencies of FUZZY Terms: 81
Number of UNIQUE FUZZY Terms: 51

Fuzzy terms matchings occurences:

Canonical Term	Matching term candidate	Occurrences
benigne	benign	1
coxartrozei	coxartroza	1
paralizia	paralizie	1
tisulare	tisular	1
oncologica	oncologie	1
pediatrie	**geriatrie**	**3**
medicatia	medicatie	1
plastica	**plastida**	**1**
pupilei	pupila	1

medicamente	medicament	1
interni	**internist**	**1**
preoperator	preoperatoriu	1
protejam	**proteza**	**1**
microbul	microb	1
oftalmologic	oftalmologie	6
simptome	simptom	2
corneei	cornee	1
retinei	retina	1
ischemica	ischemie	1
nutritie	**denutritie**	**3**
psihologic	psihologie	1
miocardic	miocard	1
(glanda	glanda	1
traumatologie	**reumatologie**	**1**
oftalmologice	oftalmologie	2
neonatologie	**deontologie**	**1**
maligna	malign	1
pneumologie	**neurologie**	**1**
prenatala	prenatal	2
psihologia	psihologie	1
vitamine	vitamina	1
conjuctivita	conjunctivita	3
neurologice	neurologie	1
antibiotice	antibiotic	1
carotide	**parotida**	**1**
oftalmolog	oftalmologie	11
reumatolog	reumatologie	1
neoplasme	neoplasm	1
urologie	**virologie**	**1**
bacteriologic)	bacteriologie	1
conjunctiva	conjunctivita	1
conjunctivala	conjunctivita	3

hepatic	hepatita	1
benigna	benign	1
faciale	facial	2
tomografii	tomografie	1
medicala	medicina	2
oftalmologica	oftalmologie	1
maligne	malign	1
proteja	**proteza**	**1**
diagnosticul	diagnostic	3

Appendix B

Responses from the questioners about the *word – medical term* matching

The responses from the three questioners about the *word - medical term* mappings in Romanian used for both Trap Questions Study and Multiple Users Agreement

Responses for questioner 1:

oncolog - oncologie; Primul cuvant provine din propozitia: "…Medicul oncolg decide daca este cazul sa…"	dispret - dispnee; Primul cuvant provine din propozitia: "... a fost tratat cu dispret de catre..."	cerebral - cerebel; Primul cuvant provine din propozitia: "…un atac cerebral la fiecare sase secunde…"
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Da - se refera la acelasi termen	Da - se refera la acelasi termen
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite

Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Da - se refera la acelasi termen	Da - se refera la acelasi termen
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite

Responses for questioner 2:

cerute - cerumen; Primul cuvant provine din propozitia: "…sfaturile au fost cerute de catre medicul…"	**radioterapia - radioterapie; Primul cuvant provine din propozitia: " …Radioterapia este rareori utilizata in stadiile initiale;…"**	**treptat - trepan; Primul cuvant provine din propozitia: "…apoi boala trece treptat in faza de…"**
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu stiu.	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre	Da - se refera la acelasi termen	Nu - e vorba despre

lucruri diferite		lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite

Responses from questioner 3:

cerebral - cerebel; Primul cuvant provine din propozitia: "…un atac cerebral la fiecare sase secunde…"	**benigne - benign; Primul cuvant provine din propozitia: "...Majoritatea cancerelor de colon debuteaza ca si proliferari benigne de celule, sub forma polipilor colonici…"**	**program - prognat; Primul cuvant provine din propozitia: "...a fost lansat un program de asistenta a pacientilor…"**
Da - se refera la acelasi termen	Da - se refera la acelasi termen	Nu stiu.
Nu - e vorba despre lucruri diferite	Nu stiu.	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Nu stiu.	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Da - se refera la acelasi termen	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite
Nu - e vorba despre lucruri diferite	Da - se refera la acelasi termen	Nu - e vorba despre lucruri diferite

Appendix C

This section presents a description of *text4all* specifications, functionality and the inner design of several modules by using UML diagrams.

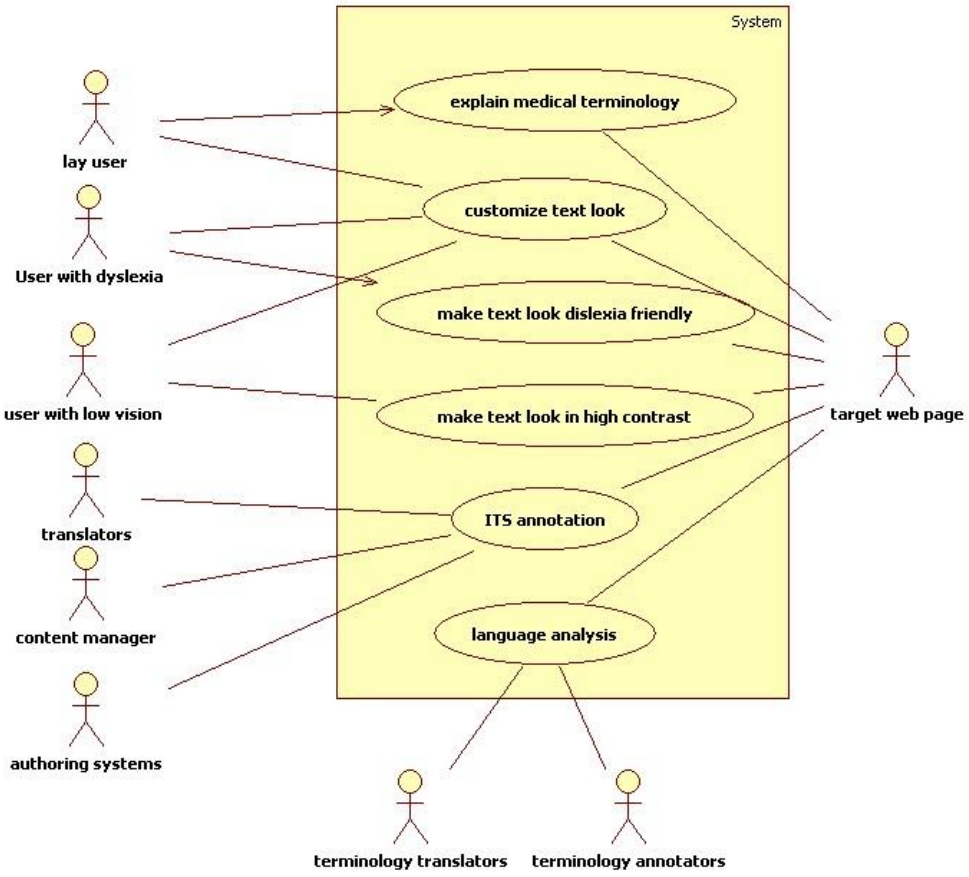

Fig. A.c.1 Use Case diagram illustrating most of the use cases for text4all and the involved actors.

In Figure A.c.1 one can see that all use cases are associated with the target web site actor, since it is always involved in any adaptation process done by the system. The main use cases are presented together with the actor involved:

- The use case of explaining medical terminology intended for lay user
- The use case of customizing the look of the text, intended for users with dyslexia, with low vision or other users.
- The use case of customizing a web page to make it dyslexia friendly intended for users with dyslexia
- The use case of making a web page look in high contrast mode intended for users with low vision

- The use case of annotating terminology using ITS standard, useful for translators, authoring systems that need to annotate the content or content managers that want to semantically enrich their text.
- The use case of language analysis intended for actors interested in performing language and terminology studies, like translators or terminology annotator tools

Next, the steps involved in the process of performing web page adaptation done by *text4all* web mediator are presented.

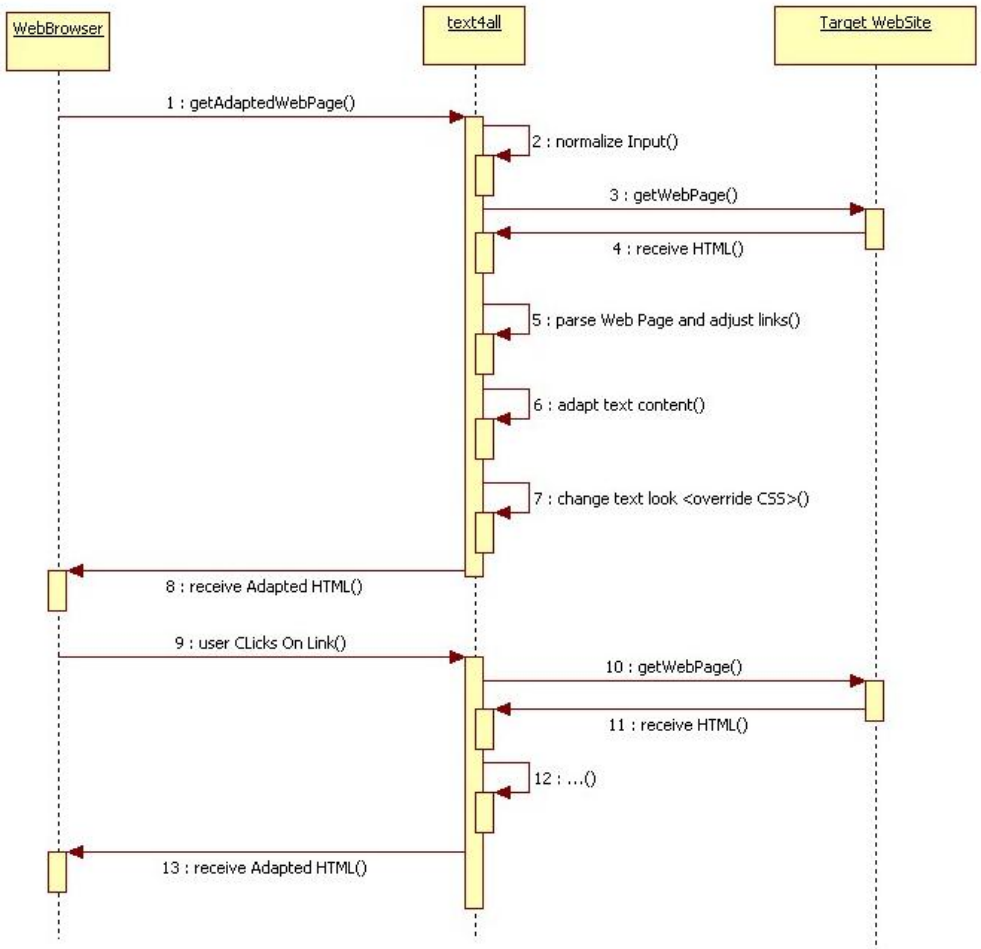

Figure A.c.2 Sequence diagram of adapting a web page at language and/or presentation level by using text4all

Figure A.c.2 presents the sequence of steps performed in order to adapt a web page. After the user enters the target web page and the adaptation setting into the *text4all* web service, the URL is being normalized (step 2) by making sure the proper protocol is entered and the URL is valid. After this, *text4all* web service will

query the HTML data from the target URL and will start the adaptation process. The adaptation is done in several steps:

- First the HTML is parsed (step 5); the links are redirected so that they will lead to an adapted version of their address. This is done selectively, only adapting links that lead to a new web page, while preserving the original link for images, scripts and other elements that need to be requested from the target web page.
- Next the language is being adapted (step 6), if the user requested so. The adaptation involves recognizing terminology or difficult words (the last one is used in *dysWebxia* project). The sequences within this step are presented in figure A.c.3. The techniques used for adapting the language are presented in full details in chapter 3, 4 and 5 in the thesis.
- In step 7 the look of the webpage is changed. Depending on the settings entered by the user, a predefined CSS will be used, or a custom one will be generated, and it will be used to override the default CSS of the webpage.

After these steps the adapted HTML content is ready to be served to user. The next time the user will click on a link from the adapted web page, it will lead him to the adapted version of that page.

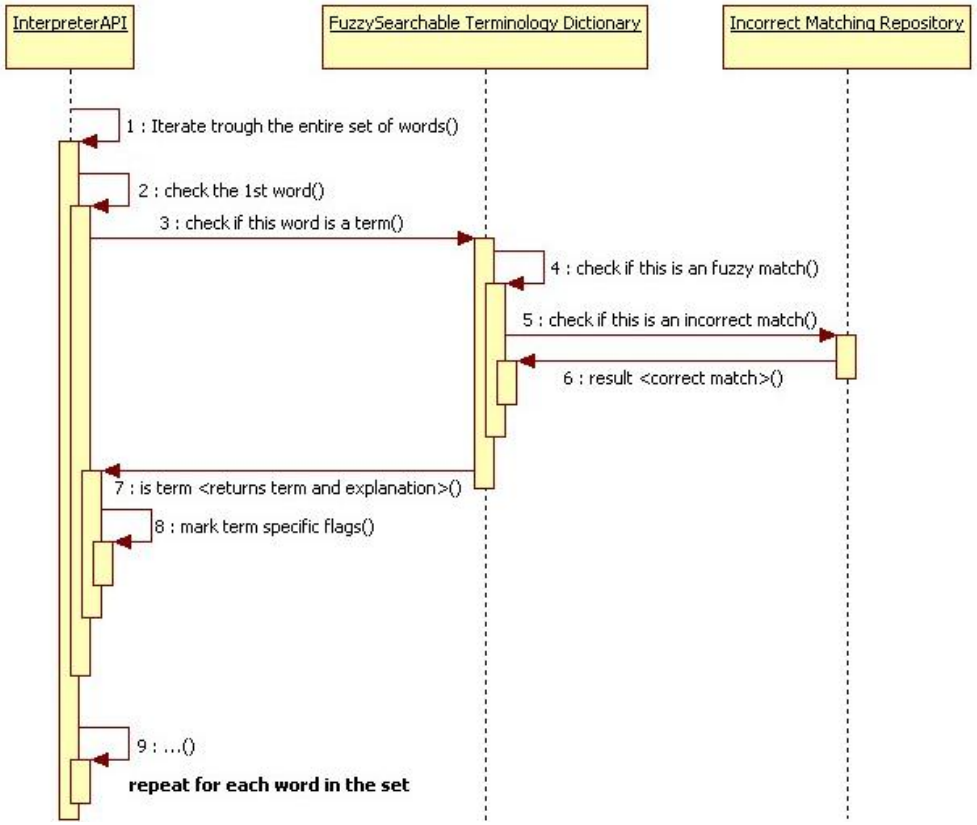

Figure A.c.3 Sequence diagram presenting the steps for recognizing and explaining terminology (the case when the searched word is a term is presented)

Figure A.c.3 presents the steps done in the module responsible for terminology recognition. The *InterpreterAPI* is the part that handles the language processing. The process involves iterating trough the entire set of words (step 1 in the sequence) that were split prior to this phase. While iterating a repeating process is done for each word. The process done for the first word is presented in step 2. The author considered the case when this word in specialized term. In order to determine this, in step 3 the Fuzzy Searchable Terminology Dictionary was interrogated and a fuzzy match was confirmed in step 4. Once a fuzzy match is identified, the match is searched in the Incorrect Matching Repository (step 5) to validate the fact that this is not an incorrect match. When this is confirmed (in step 6), the term is returned together with the associated explanation to the *InterpreterAPI* (in step 7), which is saving the term metadata in a specific structure, so that when the iteration of the words is done, the identify terms will contain their explanation and other flags.
The steps ranging between 2 and 8 (inclusive) are repeated for each word in the list.

Next the architecture of text4all service is presented in Figure A.c.4, illustrating the three main modules: Input management and Parsing, Language Level Adaptation and Presentation Level Adaptation.

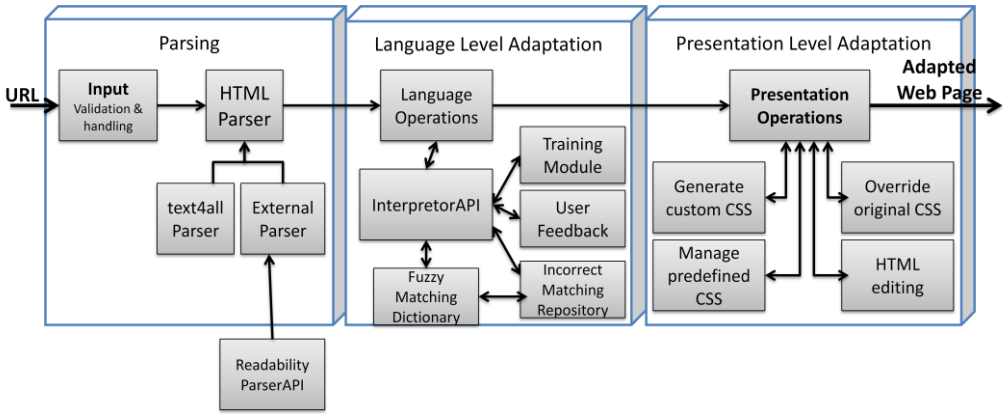

Figure A.c.4 Overview of the architecture of the *text4all* service, presenting the main modules

The Language Level Adaptation module includes the *InterpreterAPI* and the other parts that were presented in the Sequence Diagram from fig. A.c.3. Part of the internal class design of this module, focusing on the *InterpreterAPI* is presented in the class diagram from figure A.c.5. One can see that the *FuzzyHashMap* data structure, used for keeping and fuzzy searching terms, is kept together with its associate Incorrect Matching Repository in the class *DictionariesWrapper*. The part of loading the data in the data structures (that can be done from XML or from a database) is omitted here, to avoid complexity of the scheme. The *InterpreterAPI* is taking as input a *String* text, representing the entire message content, splits the string, checks each word to see if it is a term, and returns a list of *Word* objects, that includes flags indicating if the term is a term or not, and the metadata for terms.

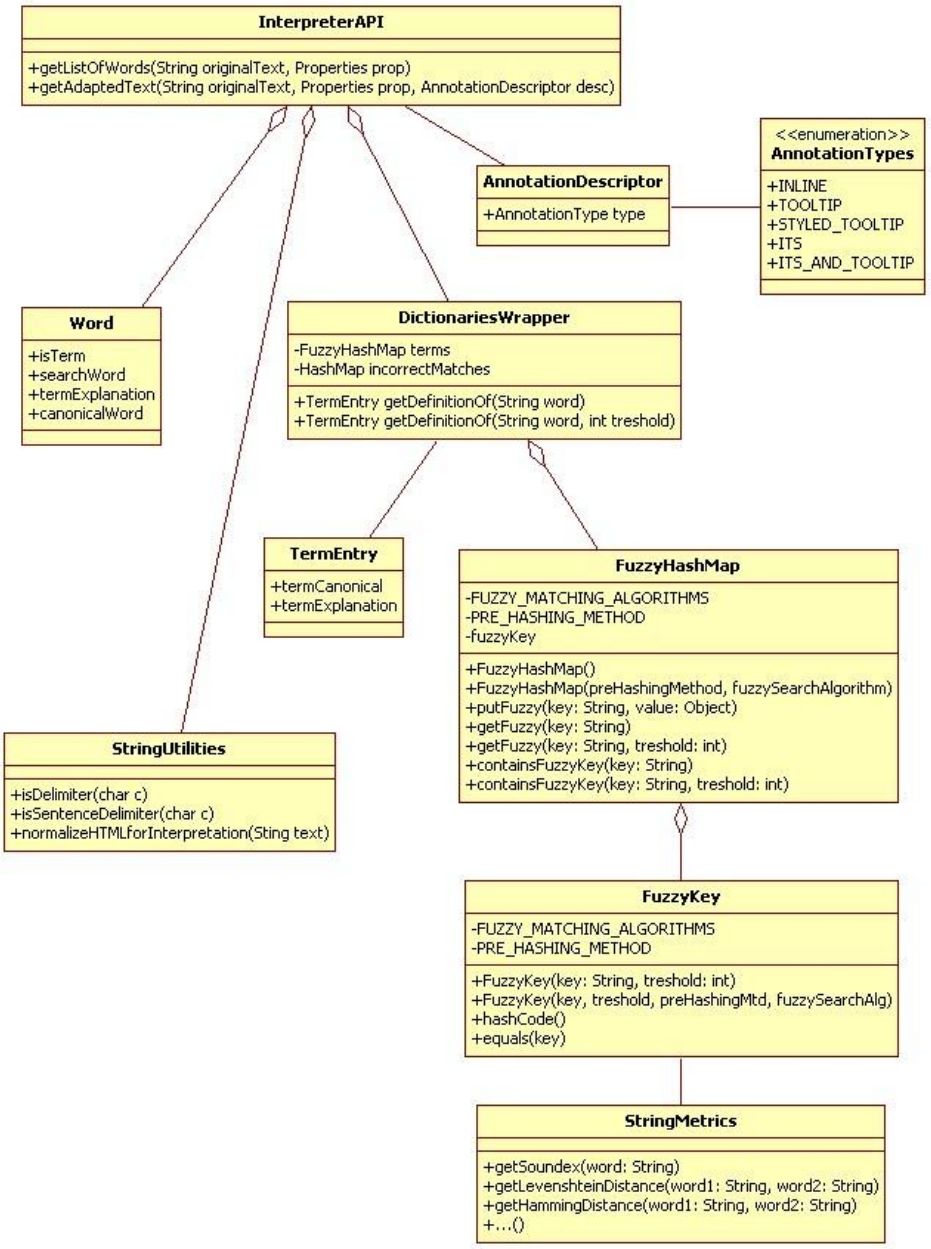

Figure A.c.5 Class Diagram of *InterpreterAPI* module from *text4all*

The Parsing module takes the input from the user, validates and normalizes it and then parses the HTML, paying attention at several details: 1) redirect links so that they will lead to adapted versions of the original web pages; 2) make sure we set the proper layout from this step. The layout can preserve the original one, laniaries

the content or clean the content by eliminating all original navigation elements and structure and preserving only the main content. Cleaning up pages can be useful for reading online articles and it is not intended for pages that need a lot of navigation (like the home page). Cleaning pages is not a trivial task, involving advance HTML parsing, filtering and manipulation. *text4all* uses two implementations for this task: one made by the author, implemented inside the project and an external one, more advanced, by incorporating the *ReadabiltyAPI.* Some class structure details of the Parser are presented in the UML Class Diagram from figure

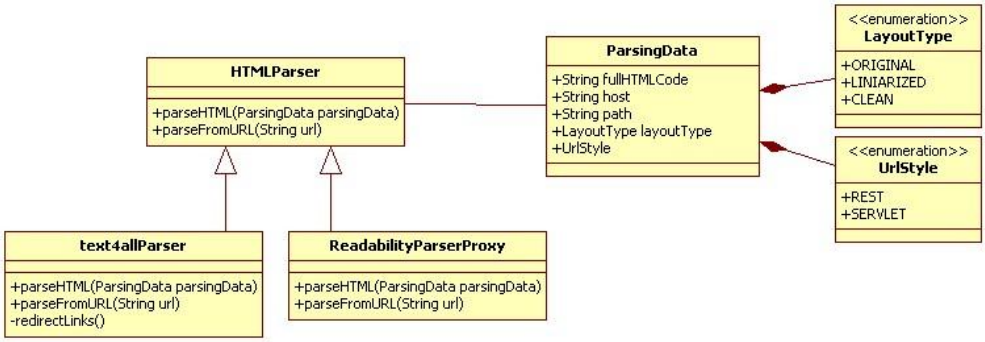

Figure A.c.6 Class Diagram of Parser module from *text4all*

text4all can use two kinds of URLs when adapting a web page: *Servlet* style (with "key=value" parameters) or REST style. The later one is used more when interacting with *text4all* all directly from the address bar by writing URL shortcuts.

This is not an exhaustive description of *text4all* insight, there are other modules that could be detailed in Class Diagrams or functionalities that can be detailed in Sequence Diagrams or other kind of UML diagrams, but the author selected the parts that were considered more important in the system.

(The following sections are in Romanian Language)

Curriculum vitae

Vasile TOPAC este născut în Arad la data de 04 Noiembrie 1984 şi a urmat cursurile Colegiului Economic Arad, specializarea Matematică-Informatică. În anul 2008 finalizează cursurile Facultății de Inginerie din cadrul Universității "*Aurel Vlaicu*" Arad, devenind inginer diplomat în domeniul "*Ingineriei sistemelor și calculatoarelor*", specializarea "*Automatică și informatică industrială*". Pe parcursul acestor studii beneficiază de bursă Erasmus pentru un semestru de studiu la Universitatea Fernando Pessoa din Porto, Portugalia. De asemenea este implicat în concursuri tehnice, obţinând locul II la finala naţională de la Suceava a concursului ImagineCup 2008, secţiunea Software Design. În anul 2008 începe activitatea profesională în industrie ocupând poziţia de inginer programator la o companie locală de dezvoltare software.

Din octombrie 2008 efectuează studii universitare de doctorat în cadrul Universității Politehnica Timișoara, în domeniul Calculatoare și Tehnologia Informației, având ca sferă de interes modalităţi de îmbunătăţire a accessibilităţii informaţiei şi domenii conexe precum interacţiunea om-calculator, procesarea limbajului natural sau structuri de date şi algoritmi. Tema de cercetare urmăreşte îmbunătăţirea înţelegerii textelor conţinând limbaj specializat. De asemena, are în vedere şi îmbunătăţirea accesibilităţii textului pe parte de prezentare (aspect vizual) pentru anumite categorii de utilizatori.

Pe parcursul activităţii de cercetare participă la o serie de conferinţe în ţară şi în străinătate. În anul 2011 primeşte *Google Student Award* pentru a participa la conferinţele *Web4All* şi *WWW*, conferiţe care s-au dovedit a fii un punct de cotitură în activitatea lui de cercetare, marcând începutul unor colaborări fructuoase. Activitatea de cercetare s-a materializat într-un număr de 14 articole publicate în volumele unor manifestări ştiinţifice din ţară şi străinătate din care 1 articol într-o revistă ISI cu factor de impact 1.6, 5 articole indexate ISI Proceedings şi 5 articole indexate in baze de date internaţionale (BDI). De asemenea în urma activităţii de cercetare au rezultat o serie de aplicaţii software, dintre care una open source.

LISTA PUBLICAŢIILOR REZULTATE ÎN URMA TEZEI DE DOCTORAT, PUBLICATE SUB AFILIERE UPT

1. Lucrări ştiinţifice publicate în reviste indexate ISI

[1] **V Topac**, V Stoicu-Tivadar, *Patient Empowerment by Increasing the Understanding of Medical Language for Lay Users,* Methods of Information in Medicine, 52(5):454-62. doi: 10.3414/ME12-02-0006. Sep 2013; (Impact factor 1.600)

2. Lucrări ştiinţifice publicate în volumele unor manifestări ştiinţifice (Proceedings) indexate ISI Proceedings

[1] V Stoicu-Tivadar, L Stoicu-Tivadar, **V Topac**, D Berian, *A WebService-based alarm solution in a telecare system*, Applied Computational Intelligence and Informatics,. SACI'09. 2009

[2] **V Topac**, V Stoicu-Tivadar, Software Architecture for Better Text-Based Infromation Accessibility, Telecommunications, 2009. AICT'09. Fifth Advanced International Conference on, Venice 2009 **[Best Student Paper]**

[3] **V Topac**, V Stoicu-Tivadar, *Software architecture for improving accessibility to medical text-based information.* Studies in health technology and informatics 150, 146, MIE2009, Sarajevo 2009

[4] V Stoicu-Tivadar, L Stoicu-Tivadar, D Berian, **V Topac**, *WebService-based solution for an intelligent telecare system*, Intelligent Engineering Systems (INES), 2010

3. Lucrări ştiinţifice publicate în reviste de specialitate indexate BDI

4. Lucrări ştiinţifice publicate în volumele unor manifestări ştiinţifice (Proceedings) indexate BDI

[1] **V Topac**, *Efficient fuzzy search enabled hash map*, Soft Computing Applications (IEEE-SOFA), 2010 4th International Workshop on, 39-44, Arad 2010 **[Best student paper]**

[2] **V Topac**, V Stoicu-Tivadar, *Patient empowerment by increasing information accessibility in a telecare system.* Studies in health technology and informatics 169, 681, MIE2011, Norway 2011

[3] **V Topac**. 2011. *Towards a universal accessibility for textual information*. In Proceedings of the International Cross-Disciplinary Conference on Web Accessibility (W4A '11). India, 2011 **[Google Student Award]**

[4] D Berian, V Gomoi, **V Topac**, *A hybrid solution for a telecare system server*, Applied Computational Intelligence and Informatics (SACI), 2011 6th

[5] Luz Rello, Clara Bayarri, Azuki Gòrriz, Ricardo Baeza-Yates, Saurabh Gupta, Gaurang Kanvinde, Horacio Saggion, Stefan Bott, Roberto Carlini, **Vasile Topac**, *DysWebxia 2.0!: more accessible text for people with dyslexia*, Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility, W4A2013, Brasil 2013

5. Lucrări ştiinţifice publicate în volumele unor manifestări ştiinţifice internaţionale (Proceedings) din străinătate

[1] V. Topac, A web page adapter for users with visual limitations, YOUNG RESEARCHERS CONSORTIUM, International Conference of Computers Helping People, ICCHP 2012, Linz, Austria, July 2012

[2] **V. Topac**, *The development of a text customization tool for existing web sites*, in Text Customization for Readability, W3C Online Symposium, Nov. 2012. Available from: http://www.w3.org/WAI/RD/2012/text-customization/p7
[3] **V. Topac**, V Stoicu-Tivadar, *Evaluation of Terminology Labeling Impact over Readability.* Easy-to-Read on the Web Symposium, W3C Online Symposium, Dec. 2012, Available at: http://www.w3.org/WAI/RD/2012/easy-to-read/paper1/