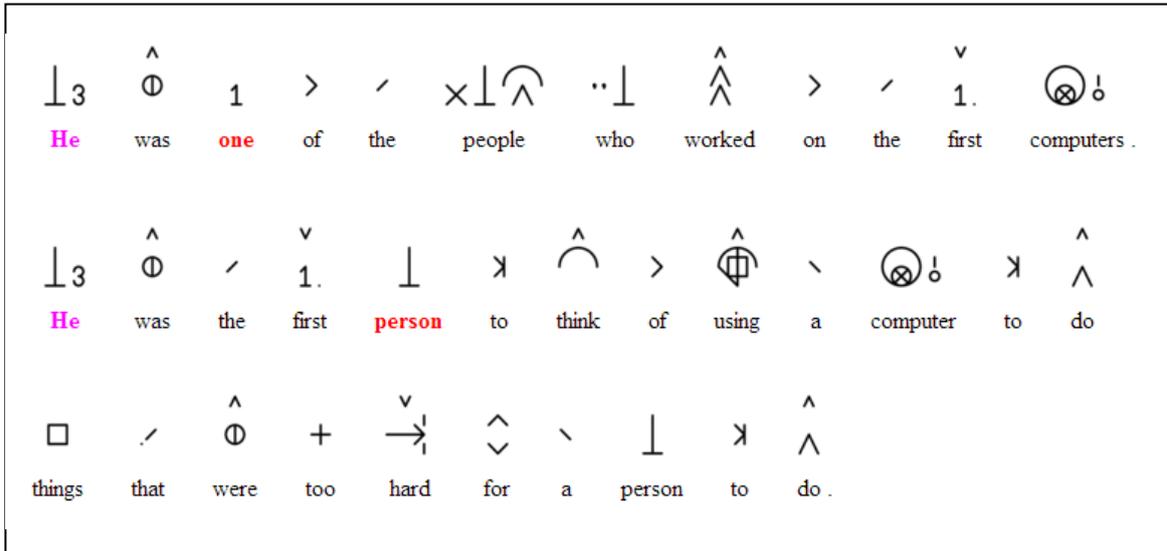




UNIVERSITY OF GOTHENBURG



Simplipedia

Making Wikipedia accessible

Master of Science Thesis in the Program Computer Science

MARCUS TILLANDER

University of Gothenburg
 Chalmers University of Technology
 Department of Computer Science and Engineering
 Göteborg, Sweden, February 2014

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Simplipedia

Making Wikipedia accessible

Marcus Tillander

© Marcus Tillander, February 2014.

Examiner: Peter Ljunglöf

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden February 2014

Abstract

Wikipedia is a massive encyclopedia of information available on the internet, but everyone may not be able to take part of this information. Due to cognitive impairments and other disabilities people may find it hard to read and understand normal text. A possible solution to making the text more accessible for disabled reader is to augment the text with Blissymbolics, a graphical language made of pictograms.

The subject of this thesis is to implement a tool for automatically connect Blissymbolics to the data from Wikipedia articles. The software generates an XML output from which it is possible to create different visualisations which can target different groups.

The software uses *CCF Symbol Server* a service which looks up Blissymbolics for words, these symbols is stored in the XML together with the word which was used to look it up. The software also uses a NLP tool from Stanford, from which certain information is extracted and stored in the XML.

An evaluation of how well the symbols matches the word was also made, the result shows that it has potential to be very effective at this, but a symbol selector needs to be implemented. Without a symbol selector it is not as effective and up to one third of the words may get a bad matching symbol.

This report is finally concluded with a discussion and a few ideas of how the improve the result further.

Acknowledgement

I would like to thank my thesis supervisor Peter Ljunglöf for his support and advices during the thesis. Furthermore i would like to give credit to Blissymbolics Communication International for the images they have created and shared under Creative commons licence. As well as Stanford NLP Group for the software they have developed that have been used in this thesis. Finally i would like to thank the creators of the CCF Symbol Server.

Marcus Tillander. Göteborg, Sweden 17 February 2014

Contents

1	Introduction	1
2	Background	2
2.1	Accessibility	2
2.2	Blissymbolics	2
2.2.1	CCF-Symbol Server	3
2.3	Wikipedia	4
2.4	Natural Language Processing	5
2.4.1	Part-of-speech Tagging	6
2.4.2	Named Entity Recognition	7
2.4.3	Lemmatization	7
2.4.4	Parsing	8
2.4.5	Coreference Resolution	8
3	Method and materials	10
3.1	Reading Wikipedia	11
3.2	Stanford NLP pipeline	13
3.2.1	Part-of-speech	14
3.2.2	Named Entity Recognition	14
3.2.3	Parsing	14
3.3	CCF Symbol Server	15
3.4	Intermediate format	16
3.5	Visualization	18
4	Evaluation	21
4.1	Symbol Server	21
4.2	Grammatical relations	23
5	Discussion	25
5.1	Symbol coverage	25
5.2	Multi-word units	25
5.3	Grammatical relations	26
5.4	Further work	26
	Bibliography	28
A	Appendix	30
A.1	Alan Turing: Full XML example	30
A.2	Alan Turing: Full Visualization example	45

1 Introduction

Today's society put great demands on citizens when it comes to staying updated in different areas. A considerable large part of the population has problems taking part of regular text. According to *Dyslexiföreningen*¹ 5–8% of the Swedish population have dyslexia. This is only one of the possible problems people may have concerning reading. It can also be due to cognitive impairment or other disabilities.

In 1996 a survey made by OCED (Organisation for Economic Co-operation and Development) showed that “8 per cent of the adult population encounters a severe literacy deficit in everyday life and at work” in Sweden, which was the highest scorer. In other countries the figure was even worse, in 14 out of 20 countries, at least 15% had low literacy skills [1].

This group of people may benefit greatly by easy-to-read text, people with severer disabilities may benefit even further if the text is accompanied with Augmentative and Alternative Communication (AAC) symbols[2]. Dyslexic people may benefit from other changes to the text, like layout and font.

In this thesis a program has been developed, its goal is to simplify and make the articles on Wikipedia more accessible. This was done by implementing support for an AAC system called Blissymbolics[3]. It is a set of symbols that is used to convey concepts. This thesis strive to create a modular result which makes it possible to configure it, to target the right audience.

In chapter 2 necessary background to the development of this software is presented. Chapter 3 describes how the software is implemented, as well as a description of the intermediate XML format. In chapter 4, the evaluation and its result is presented. In chapter 5, the result of the evaluation and of the software is discussed, it also discusses what can be improved further.

¹<http://www.dyslexiforeningen.se/>

2 Background

2.1 Accessibility

There exists many guidelines of how to make material accessible, many of them are concerning technical aspects. Aspects like choosing the right font, not using too many colors, having alternative text to images, tables etc. These guidelines are good and should be followed, they aim to make material accessible for all, regardless of equipment, language or mental ability [4]. But in some cases they are not enough, it is also important to make text easy-to-read. This can be done by using shorter sentences, easier words and more[5]. People with different reading disabilities may need even further help, but there are not many guidelines for how to make text accessible for this audience.

However, there exists other methods to make content available for people with disabilities, they go under the umbrella term Augmentative and alternative communication (AAC) [2]. It is a wide set of methods and tools for helping people communicating, both in writing and speech. A common technique is speech synthesis, but there are many more.

2.2 Blissymbolics

Blissymbolics is an AAC system, which was originally developed as a tool for international communication, but has since been used for AAC. Blissymbolics is a purely visual language and it can not be spoken. It is used by people with communication disabilities as a communication aid. But it is not intended for all kinds of disabilities, e.g. blind and deaf have no use of it. It is people with “severe speech and physical impairment” [3] that mainly use it. In some cases, when the users can not speak or write, they have charts over the symbols and point on the ones representing what they want to say.

The language consist of over 4000 symbols that represents words and concepts, each blissymbol can in turn consist of other symbols [6]. In figure 2.1 is an example of how symbols can be put together to represent an other word. In this case we have the word “friend”, it is constructed by combining the words “person”, “feeling” and “positive”. Positive in turn is constructed by combining “plus” and “intensity”. So a friend is a person which you have positive feelings for.

There exist indicators that together with a symbol can represent different word classes. In figure 2.2 there is an example of how the word measurement can be turned into the verb “measure” (as in “to measure”), which is done by adding the action indicator.

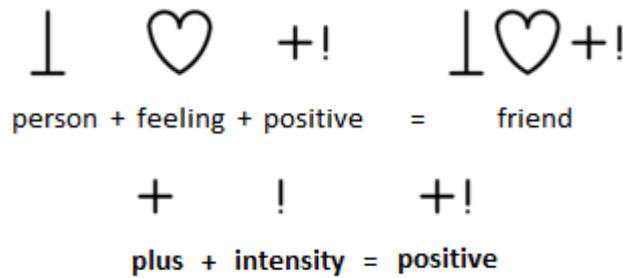


Figure 2.1: A Blissymbolics example (borrowed from [3])

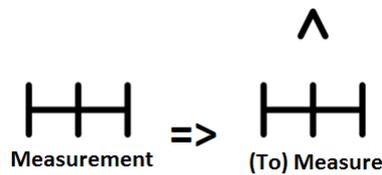


Figure 2.2: A Blissymbolics example with an action indicator

2.2.1 CCF-Symbol Server

The CCF-Symbol Server is a server tool that resulted from the EU project *AEGIS*,² one of its objectives was to develop free software for graphical symbol support. The symbol server is based on the Concept Coding Framework (CCF),³ a framework for communicating with symbols and pictures between programs and over internet.

The CCF symbol server has previously been used in a plug-in for OpenOffice. The user could write text in a document and the symbol server then gave suggestions of possible symbols that matched the word. The symbol server has also been used in a few other projects [7].

The symbol server contains a database of roughly 5000 concepts, when it revives a word it will look this word up in the database. It then returns the Concept ID, any possible symbols for this concept and a description of the concept. The description is retrieved from Wordnet[8], a large English lexical database of concepts. The Symbol server has support for two different AAC systems, Blissymbolics and ARASAAC.⁴

²<http://www.aegis-project.eu/>

³<http://www.conceptcoding.org>

⁴<http://www.catedu.es/arasaac>

2.3 Wikipedia

Wikipedia is a well known massive encyclopedia available for free on the internet. One of its less known language sites is the Simple English Wikipedia, it's written in Basic English which is a simplified version of English [9]. It uses fewer words and has an easier grammar than normal English. Simple English Wikipedia are intended to be used by people with special needs, like children, students who are trying to learn English and people that have problems with learning and reading. All these groups might benefit from using Simple English Wikipedia instead of English Wikipedia.

However, the group of people with reading and learning difficulties might benefit even more if the text is augmented with additional information. An example would be persons with cognitive impairments that might not understand certain words or be able to read at all. By augmenting the text with symbols that the person are familiar with, s/he will understand the text much better.

Wikipedia is built on a software called MediaWiki,⁵ which is a server-based software that is used to host large wikis. MediaWiki uses a markup language, called MediaWiki Markup to format its pages.⁶ It is used to apply font styles as **bold** and *italic* on text. It can also be used to create headings, links, list and other changes to the structure of the text. In table 2.1 some of the markups can be seen. MediaWiki also has support for a number of HTML tags, some of them can be achieved with the wiki markup, some cannot. This means there are two ways of creating the same effect. Which introduces a source of ambiguity which generally makes is difficult to interpret.

Wikipedia also make use of templates and extensions. A template is a page which is used to embed into other pages. Templates are problematic since the code for the templates is not stored in the article page. The software developed in this thesis takes a page as input and returns the result in an other format. That format do not have support for MediaWiki templates and the software can not handle them, so it will result in some content loss, compared to the original page. Extensions has the same problem, they implement some other functionality to the MediaWiki it is installed on. This functionality is not supported by the software developed in this thesis, therefore, extensions may result in either content loss or strange text in the page.

MediaWiki is open source and hence free to use to set up your own wiki, there are many other wikis running on MediaWiki but there exists other alternatives as well. The software developed in this thesis will work on any wiki based on MediaWiki, since they run on the same platform and use the same markup language. There is one restrictions however: the language has to be English since other parts of the program assume it.

⁵ <http://www.mediawiki.org/wiki/MediaWiki>

⁶ At http://en.wikipedia.org/wiki/Help:Wiki_markup all the syntax is presented

Description	Syntax
Italic or bold	<code>"<i>italic</i>"</code> or <code>"bold"</code>
Wikipedia links	<code>[[Link]]</code> or <code>[[Link]]ed</code>
Redirected links	<code>[[Target Text]]</code>
References	<code><ref>Text and/or link</ref></code>
Templates	<code>{{Template name}}</code>
Math formulas (using Latex markup)	<code><math> 2x \times 4y </math></code>
Markup that must start on a newline	
Headings	<code>= Section =</code> <code>== Subsection ==</code> etc.
Bullet/Number list	<code>*/# List item</code> <code>**/## Sublist item.</code>
Indenting text	<code>: one step indented</code> <code>:: two steps, etc.</code>

Table 2.1: Some examples of MediaWiki Markup

2.4 Natural Language Processing

Natural Language Processing (NLP) is the process of analysing human (natural) languages to derive meaning from them, NLP includes many different tasks. In this thesis a NLP-tool developed by Stanford has been used ⁷ it is a State-of-the-art NLP tool, with an integrated framework which is capable of doing many of the tasks in the field. The core NLP suit, which is used in this thesis is capable of tokenization, part-of-speech tagging, named entity recognition, parsing, and co-referencing. Of these all except co-referencing is used in the developed software. Many of these tasks uses machine learning algorithms, hence they need models to train on, these models are also included in the suite. But it is also possible to create your own models which could change how the tool annotates the input. This may be useful when a certain type of text is targeted.

The tool works as a pipeline, doing one task before it starts on the next. Each step analyses the text and gives more information about it, some of the tasks annotate words, while other works on the structure of the text. Some tasks are depending on others to work, in figure 2.3 these dependencies is showed.

The first task for any NLP tool is text segmentation, which includes tokenization and sentence splitting. Tokenization is the process of breaking the text up into words, symbols etc. and sentence splitting breaks these words into sentences. None of these tasks is as trivial as one would think. E.g. usually a full stop (.) ends a sentence, but there are cases where a full

⁷<http://nlp.stanford.edu/>

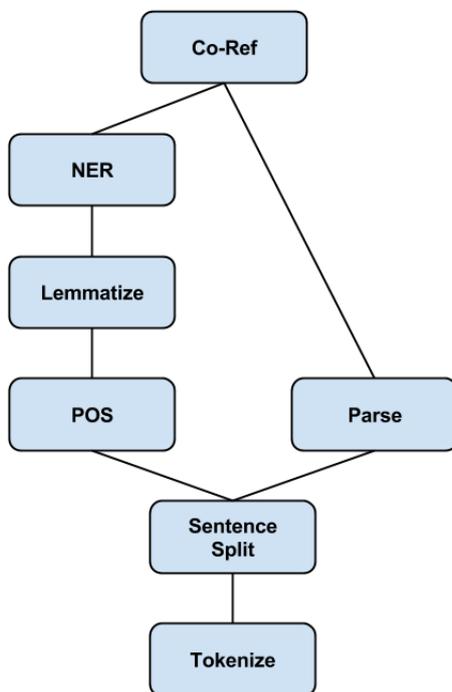


Figure 2.3: Dependencies of different NLP tasks

stop appear in the middle, e.g. when abbreviations are used.

Segmentation is needed for all other tasks to work, other tasks include Part-of-speech tagging (POS), Named Entity Recognition (NER), Parsing and more.

2.4.1 Part-of-speech Tagging

Part-of-speech tagging is the task of annotating each word with its part-of-speech i.e. word class. The tagger tries to find verbs, nouns, adjectives etc. but it can be more precise than that. There are more classes and many sub-categories, some POS-taggers have almost 200 different tags in their system, the number varies depending on the language it tagges. There also exists a Universal POS tag set [10], which consists of 12 tags, and works for 22 different languages. It is created to work for many languages, therefore the name Universal.

The Stanford tagger⁸ uses the Penn Treebank tag set, it consists of a total of 48 different tags for words and punctuations [11]. The part-of speech tagger from Stanford uses machine learning algorithms in order to find the right tag for a word. These algorithms use a large corpus to train on, i.e.

⁸<http://www-nlp.stanford.edu/software/tagger.shtml>

they use large samples of text, which has been manually annotated, to predict what POS a word should have [12, 13].

In figure 2.4 an example of the sentence:

“Carl likes eating German sausage, he also likes milk.”

can be seen. The same sentence will be used in each of the parts below to show how the different tasks works. As can be seen each word gets a POS, even punctuations get one.

Carl _{NNP} likes _{VBZ} eating _{JJ} German _{JJ} sausage _{NN} , he _{PRP} also _{RB} likes _{VBZ} milk _{NN} .

Figure 2.4: POS tagging example

2.4.2 Named Entity Recognition

The Named Entity Recogniser tries to find entities (words) in the text that are names. It can be names of people, places and organisations, it can also recognise dates, time and money. This tool uses a Conditional Random Field (CRF) classifier that can be trained on different models to be able to work for any classification task. In Stanford⁹ it comes with three different models that it has trained on, giving the tool the ability to classify words as Location, Person, Organization, Money, Percent, Date, Time and Miscellaneous [14].

Figure 2.5 shows two named entities, the name of a person, “Carl” and the name of a country, which goes under the category Misc (Miscellaneous). The other words do actually get annotated in a category called “O”, but it is not shown in this example.

Carl _{Person} likes eating German _{Misc} sausage, he also likes milk.

Figure 2.5: NER tagging example

2.4.3 Lemmatization

Lemmatization is the process of annotating each word with it’s lemma i.e. base form. A word may have many different inflections but still have the same lemma, an example would be “jump”, “jumped”, “jumping”, they all share the same lemma, “jump”. There are other words that are inflected in an other way than by changing the suffix, “better” is one example, its

⁹<http://www-nlp.stanford.edu/software/CRF-NER.shtml>

lemma is “good”. The lemma can be used together with POS to determine the meaning of a word in its context [15].

In figure 2.6 we see that each word get annotated with its lemma.

Carl _{Carl} likes _{like} eating _{eating} German _{german} sausage _{sausage} ,, he _{he} also _{also} likes _{like} milk _{milk} .
--

Figure 2.6: Lemmatization example

2.4.4 Parsing

The parser¹⁰ works on a sentence level, finding the grammatical structure of each sentence. It uses probabilities derived from annotated corpora to find the most likely way of structuring the sentence. Stanford uses a dependency structure, but there exists other structures too. The words in a sentence are connected through grammatical relations, there are approximately 50 relations [16, 17]. These relations are binary and gives information as what the subject or object of a word is [18].

The example sentence from above, “Carl likes eating German sausage, he also likes milk.”, gives eight relations. Most of them are fairly easy to understand, all the relations the parser yields can be seen in figure 2.7. The relations can be read like the following sentence: “The nominal subject of ‘likes’ is ‘Carl’” or “The direct object of ‘likes’ is ‘sausage’”. A word can only have one subject/object relation, if there are more, they will be connected to the subject/object through a conjunct relation.

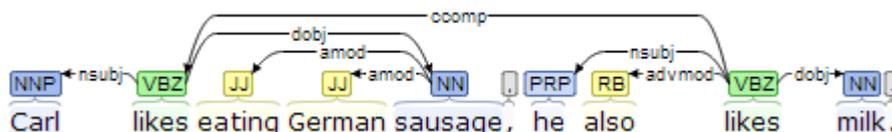


Figure 2.7: The dependency graph Stanford parser generates for the example.

2.4.5 Coreference Resolution

Coreference resolution is a task in NLP which seeks to find words in a text which refer to the same thing. It creates a link between these words, typically it is pronouns, i.e. words like “it”, “the” or “him” etc. which refers to a name or thing. But a link is also created if the same name appears again, these

¹⁰<http://www-nlp.stanford.edu/software/lex-parser.shtml>

references are stored in a graph and a representative word is selected as the entity the others refers to [19, 20, 21, 22, 23].

In figure 2.8 we can see that “he” is a reference to “Carl”, no other references were found in the example.

Carl_{Ment} likes eating German sausage, he_{Mention} also likes milk.

Figure 2.8: Coreference example

3 Method and materials

The program takes text from Wikipedia (with MediaWiki markup in it) as input. It then works on this text through a few steps, a flow over the steps can be seen in figure 3.9. It starts out with the text and ends up in an XML format, the XML can then be used to generate different visualisations. The first step is a simple reader made with regular expressions. These expressions are designed to match on the markup language as well as normal text. After this the text is passed on to Stanford NLP tool for analysis, and then the CCF symbol server looks up symbols for the words. Finally the text is glued together with all previous parts (markup, NLP and symbol data) into the XML format. By creating different visualisations it is possible to match the output to target the audiences needs or to create output for different devices.

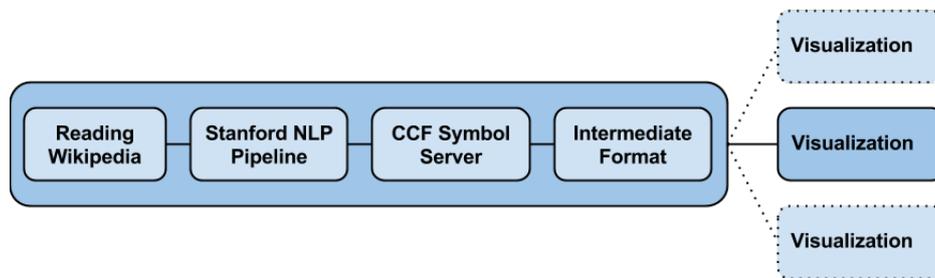


Figure 3.9: Flow over the program

To illustrate how the program works and what each step does, an example article has been chosen. The article is about Alan Turing and only the first few lines will be in the example. In figure 3.10 parts of the article can be seen as it looks on simple Wikipedia.¹¹

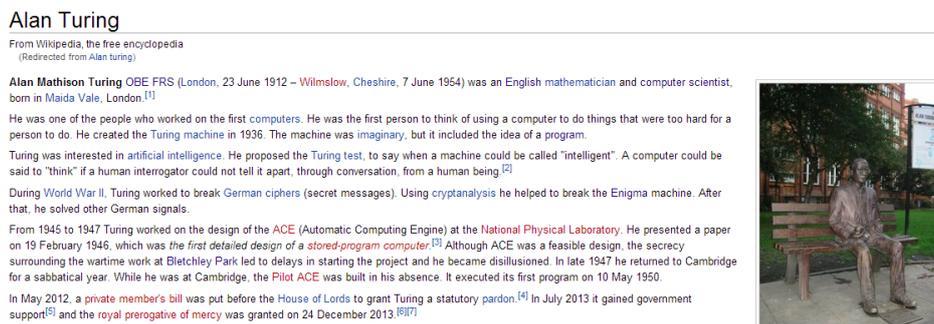


Figure 3.10: What the article Alan Turing looks like on simple Wikipedia

¹¹URL to the article used : simple.wikipedia.org/wiki/Alan_turing

3.1 Reading Wikipedia

There exists a number of parsers for Wikipedia, but none of the analysed parsers returned the articles in a format which could be continued from. Either the available parsers returned the Wikipedia page in a HTML format, which would mean the HTML had to be parsed instead. Other available parsers returned pure text, without the markup, which would mean links, lists, etc. would be hard to distinguish. This output was not adequate for the purposes of this software, therefore a reader was implemented, constructed of several regular expressions. In figure 3.11 the article, with the markup, can be seen.

```
[[File:Alan Turing Memorial Closer.jpg|right|thumb|A
[[statue]] of Alan Turing]]
[[File:Bombe-rebuild.jpg|thumb|right|A rebuild of a machine
made by Alan Turing]]

'''Alan Mathison Turing''' [[Order of the British Empire|OBE]]
[[FRS]] ([[London]], 23 June 1912 - [[Wilmslow]],
[[Cheshire]], 7 June 1954) was an [[English people|English]]
[[mathematician]] and [[computer scientist]], born in
[[Maida Vale]], London.<ref>Newman M.H.A. 1955. Alan
Mathison Turing. 1912-1954. ''Biographical Memoirs of
Fellows of the Royal Society'' '''1''': 253.
[http://rsbm.royalsocietypublishing.org/content/1/253]</ref>

He was one of the people who worked on the first
[[computers]]. He was the first person to think of using a
computer to do things that were too hard for a person to do.
He created the [[Turing machine]] in 1936. ...
```

Figure 3.11: A few lines of the article, with markup.

The reader uses a list to set priority for each regular expression, the earlier in the list the higher priority it has. Each expression is accompanied by a number, which is used to identify what was read. The reader has about 20 different regular expressions, which it tries to match against the text. Some of them are targeting MediaWiki Markup, while some are targeting text, punctuations and new lines. The reader looks at the beginning of the text string and tries to read it, going through the list until it finds a matching regular expression. When a match is found the matching text is subtracted from the original string. The matched text is then put into a list, together with the expressions ID. This is then looped until all of the text has been

read.

In table 3.2 some of the regular expressions is presented, they are written with Java's regular expression library. Some of the expressions look more complex than they need to, but this is needed due to the fact that content on Wikipedia are user generated. The expressions have been modified to handle as many oddities as possible.

The regular expression quantifier operators can be either greedy or reluctant, the difference is whether the quantifier should capture as much as possible or as little as possible. For the purpose in this software they should capture as little as possible, that's why most quantifiers have the question mark (?) accompanying it. The expressions can also capture groups and refer to them later, capturing is done with parenthesis and the referencing is done with backslash and a number (\1 usually). The number identifies what parenthesis it refers to, one (1) being the first.

Regular Expression	Reads	ID
<code>\{\{[^{}]*?\}\}</code>	Templates	1
<code><([a-zA-Z]\w*)>(\s .)*?</\1></code>	All tags(ref,math etc.)	3
<code>\[\[File:.*\]\]</code>	Files	4
<code>(*+ #+).*</code>	Bullet & Number lists	11
<code>(''?) .*?\1</code>	Bold & italic	13
<code>(=+). .*?\1</code>	Headings	14
<code>\w(?:-\w-?)*"</code>	Text	17
<code>(([,?!]+) \special)</code>	Punctuations	18
<code>\n\n</code>	Double newlines(Paragraphs)	21
<code>\n</code>	Newline	22

Table 3.2: A table of regular expressions and what they parse

The regular expression reader has some drawbacks; regular expressions is not powerful enough to handle nested tags, and in a few cases, templates are nested. When such a case appears, all expressions will fail the match and a special case is reached. This case keeps track of the opening and closing tags (for templates) until all opened are closed. Regular expressions can also be very confusing and bugs can be hard to notice and cause unexpected behaviour.

When the text has run through the reader, the result will be in a large list, with a number identifying what it was read as and the text it read. From this list, the text of the page can be extracted without the markup. This cleaned text is need for the next step in the pipeline, the Natural Language Processing. In figure 3.12 the first few lines of the markup can be seen, together with the ID from the expression which read it.

The reader has some restrictions, it has no support for templates and hence no support for tables, info boxes and more. It would be possible to

```

“[[File:Alan Turing Memorial Closer.jpg|right|thumb|A
[[statue]] of Alan Turing]]”4
“[[File:Bombe-rebuild.jpg|thumb|right|A rebuild of a machine
made by Alan Turing]]”4,
22
“””Alan Mathison Turing”””13, “[[Order of the British
Empire|OBE]]”7, “[[FRS]]”7, “(”18, “[[London]]”7, “,”18,
“23”17, “June”17, “1912”17, “-”18, “[[Wilmslow]]”7,
“,”18, “[[Cheshire]]”7, “,”18, “7”17, “June”17, “1954”17,
“)”18, “was”17, “an”17, “[[English people|English]]”7,
“[[mathematician]]”7, “and”17, “[[computer scientist]]”7,
“,”18, “born”17, “in”17, “[[Maida Vale]]”7, “,”18, “London”17

```

Figure 3.12: What the reader read the text as.

implement templates, but they would have to be implemented one by one, which is very time consuming. Images are not supported either, but it would be fairly easy to implement.

3.2 Stanford NLP pipeline

In the natural language processing the text is passed to the Stanford NLP API. It’s configured to run through tokenization, sentence splitting, POS-tagging, lemmatization, named entity recognition (NER) and finally parsing, as can be seen in figure 3.13

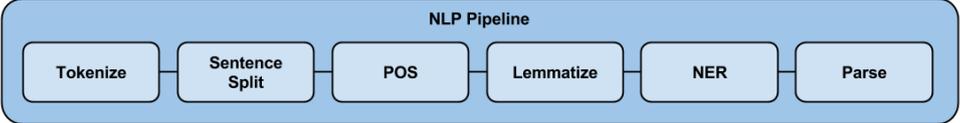


Figure 3.13: The NLP pipeline

Tokenization and sentence splitting is needed for rest of the analysing tools to work. It also gives an clear indication of where sentences ends, which may be useful when formatting the text.

It was decided to not use co-reference resolution (CoRef) as it worked poorly on the articles it was tested on. An example is an Wikipedia article about April, the CoRef concluded that occurrences of “April” referred to “late April, early May”. This makes it sound like April refers to the end of the month instead of the whole month, which it should.

The result the NLP-tool generates is a list of sentences. From each sentence a dependence graph with all grammatical relations for that sentence

can be extracted, as well as all words in that sentence. For each word it is possible to get information about that word, e.g. lemma, POS or NER.

3.2.1 Part-of-speech

The Part-of-speech tagging is used to find what words are of a certain word class. Each word get annotated with the POS which can later be extracted. The example articles can be seen in figure 3.14, the first few sentences of the article is shown. A description of what the different tags means can be found in [11].

Alan_{NNP} Mathison_{NNP} Turing_{NNP} OBE_{NNP} FRS_{NN} (London_{NNP},
23_{CD} June_{NNP} 1912_{CD} -: Wilmslow_{NNP}, Cheshire_{NNP}, 7_{CD} June_{NNP}
1954_{CD}) was_{VBD} an_{DT} English_{JJ} mathematician_{NN} and_{CC} computer_{NN}
scientist_{NN}, born_{VBN} in_{IN} Maida_{NNP} Vale_{NNP}, London_{NNP}.

He_{PRP} was_{VBD} one_{CD} of_{IN} the_{DT} people_{NNS} who_{WP} worked_{VBD}
on_{IN} the_{DT} first_{JJ} computers_{NNS}. He_{PRP} was_{VBD} the_{DT} first_{JJ}
person_{NN} to_{TO} think_{VB} of_{IN} using_{VBG} a_{DT} computer_{NN} to_{TO} do_{VB}
things_{NNS} that_{WDT} were_{VBD} too_{RB} hard_{JJ} for_{IN} a_{DT} person_{NN} to_{TO}
do_{VB}. He_{PRP} created_{VBD} the_{DT} Turing_{JJ} machine_{NN} in_{IN} 1936_{CD}.

Figure 3.14: The example lines annotated with POS.

3.2.2 Named Entity Recognition

For NER to work lemmatization is needed, which is why it is in the pipeline. NER is used to find named entities in the text, the entities it looks for are Location, Person, Organization, Money, Percent, Date, Time and Misc. All words which are not in any of these categories get annotated with an “O”, but this category is left out of the example in figure 3.15. As can be seen the NER is not perfect and sometimes miss some entities, e.g. the surname Turing.

3.2.3 Parsing

From the NLP parser some of the grammatical relations are extracted. It is the nominal subject, the direct object of the root and the root itself. Only these three relations will be marked in the example in figure 3.16. As can be seen it does not always work very well. It works best when the root is a predicate, as it is in the last sentence.

Alan_{Person} Mathison_{Person} Turing OBE FRS (London_{Location}, 23_{Date} June_{Date} 1912_{date} – Wilmslow_{Location}, Cheshire_{Location}, 7_{Date} June_{Date} 1954_{Date}) was an English_{Misc} mathematician_{Misc} and computer scientist, born in Maida_{Location} Vale_{Location}, London_{Location}. He was one_{Number} of the people who worked on the first_{Ordinal} computers. He was the first_{Ordinal} person to think of using a computer to do things that were too hard for a person to do. He created the Turing_{Misc} machine in 1936_{Date}.

Figure 3.15: Example with named entities annotated.

Alan Mathison Turing OBE FRS_{subj} (London, 23 June 1912 – Wilmslow, Cheshire, 7 June 1954) was an English mathematician_{root} and computer scientist, born in Maida Vale, London. He_{subj} was one_{root} of the people who worked on the first computers. He_{subj} was the first person_{root} to think of using a computer to do things that were too hard for a person to do. He_{subj} created_{root} the Turing machine_{obj} in 1936.

Figure 3.16: Subject, object and root annotated for each sentence in the example.

3.3 CCF Symbol Server

When the natural language processing is done, the next step is to find a matching Blissymbolic to every word. This is done through a web-API, the API may return several suggestions that matches the given word or none at all. The API does not return the image itself, instead it returns the name of the image file, this means the files need to be from Blissymbolics Communication International (BCI)¹² otherwise the names might not match.

If only one match is returned it is used to represent that word, this might not always be the correct symbol, but it works rather well. In the case when multiple alternatives are returned the first one is chosen, this is not the optimal solution and is something that can be improved in the future. To improve the time it takes to run the program a table with words and corresponding symbol is saved locally on the computer. This improves speed and as the table grows, the API will be called less and less, this also offers an way to manually change the table to change minor mismatches. It is however not an way to differentiate multiple alternatives.

In figure 3.17 the word is annotated with the selected symbol, not all

¹²www.blissymbolics.org

words get a symbol and all symbols may not have the right symbol. In the symbol table however, all alternatives are stored.

Alan Mathison Turing OBE FRS (London, 23 June_{June} 1912 – Wilmslow, Cheshire, 7 June_{June} 1954) was_{be,am,are,is,exist-(to)} an_{a,an,any} English_{English_(language)} mathematician_{mathematician_(academic)} and_{and,also,plus,too} computer_{computer} scientist_{scientist,researcher}, born in_{in,inside,interior,internal} Maida Vale, London. He_{he,she,him,her,one} was_{be,am,are,is,exist-(to)} one_{one_(digit)} of_{about,concerning,in_relation_to,of,on} the_{the} people_{people,tribe,clan,folk} who_{who,whom,that-(relative)} worked_{work-(to)} on_{about,concerning,in_relation_to,of,on} the_{the} first_{first,primary} computers_{computer}. He_{he,she,him,her,one} was_{be,am,are,is,exist-(to)} the_{the} first_{first,primary} person_{person,human_being,individual,human} to_{until,till,to} think_{think,reason-(to)} of_{about,concerning,in_relation_to,of,on} using_{use-(to)} a_{a,an,any} computer_{computer} to_{until,till,to} do_{do,act-(to)} things_{thing,object} that_{that_(there)} were_{be,am,are,is,exist-(to)} too_{and,also,plus,too} hard_{difficult,hard} for_{for_(in_exchange_for)} a_{a,an,any} person_{person,human_being,individual,human} to_{until,till,to} do_{do,act-(to)}. He_{he,she,him,her,one} created_{create-(to)} the_{the} Turing machine_{machine,appliance,engine,motor} in_{in,inside,interior,internal} 1936.

Figure 3.17: Words annotated with symbol file name.

3.4 Intermediate format

The next step is to transform all the data into a intermediate format. Information from the markup (format, links etc.), from the NLP (POS, NER and grammatical relations) and from the symbol server is needed. The intermediate format is in XML and constructed to be simple and use as few tags as possible. The basic structure of the XML format is made up of five tags, these tags contains the basic structure of the page, and the text in it. There are a few other tags available, these tags are needed to makes the output richer, in table 3.3 all tags are displayed.

The *page* tag contains the title of the page as an attribute, it also is the root tag. Inside this tag there can be the paragraph tag (named *p*), this tag has no attributes and is only used for structuring purposes. In turn the paragraph tag contain sentence tags (named *s*) and just as the paragraph it doesn't have any attributes and is used to encode the structure of the text. The sentence tag contains word tags (named *w*) and punctuation tags (named *pun*). The word tag has a few attributes, as can be seen in table 3.3 above. Data from the NLP and CCF Symbol server are stored in these attributes. The punctuation tag is used to store punctuation characters.

Tag	Attributes	Explanation
page	title	Root element, the attributes contains the name of the article.
<p>		Paragraph.
<s>		Sentence.
<w>	dep ne pos sym	Word, the attribute contains the dependency label. Contains the value from the Named Entity recogniser. Contains the value from the Part-of-speech. Contains the name of the image suggested by the CCF Symbol server.
<pun>		Contains a punctuation character.
<link>	target	A link with the target URL as attribute.
<html>	tag	A tag used to encode html tags via the attribute.
	type	A list item, the attributes specifies type (number or bullet)
<dt>		Definition list
<dd>		Definition item, also used to indent text

Table 3.3: All XML tags and their attributes

The other tags available in the XML format are described in the list below.

<**head**> The heading tag, it has an attribute of what size the heading is, 1 is largest, 6 is the smallest. These are the numbers wikipedia uses, they correspond to the different heading tags (<h1> to <h6>) in HTML [24]. This tag is on the same level as sentences, it contains the heading in pure text, i.e. no word tags is used. Making the heading a part of the paragraph tag, as attributes, was considered but since all paragraphs don't have headings it was decided to not construct it that way.

<**link**> The link tag is used to create links, both links to Wikipedia articles and links to other sites. The tag has an attribute of what site the link leads to. The tag is placed inside the sentence tag and contains word tags. An initial idea was to have an attribute in the word tag for each word that link to other articles. But this would be problematic because there are many links that span over multiple words. Which would mean these words would have to be interpreted either as one word or two links to the same targets, instead this tag was created.

<**html**> The html tag is a bit special, it is used to encode simple HTML tags as an attribute. Mostly it is used to create bold and italic text, since these occur frequently in Wikipedia articles. The tag is also placed in the

sentence tag and contains word tags for which the effect should appear. Besides bold and italic Wikipedia also has support for HTML tags which can't be achieved with the MediaWiki Markup language. These can also be encoded in the `html` tag, this tag is created for this purpose and helps keeping the XML format clean. However it has some restrictions, it only works for HTML tags without attributes.

`` This tag is used to encode lists, it has an attribute *type* describing if it should be a bullet list or a numbered list. It goes in the paragraph tag and contains sentences. List tags can also be nested. In that case the nested tag goes in the already opened list tag.

`<dt>` & `<dd>` Is used to create definition lists, the *dt* tag defines the terms while the *dd* tag is used to describe each term. Neither of these tags has any attributes, the *dt* tag is put on the same level as the list tag (inside the paragraph tag). The *dd* tag needs to be put on the same level and it has to be directly after the *dt* tag, otherwise it will not be a description of the term. The *dd* tag can also be used to create indentations, this is what happens if it doesn't appear directly after a *dt* tag. It can also be nested to create a larger indentation.

Some of these tags has similarities with HTML tags, this is by design. Partly because that is a well known language and the intention is that it should be easy to understand how the XML works. But also because HTML is a desirable end-format and since it's an easy way to make it available for the public. In figure 3.18 an example of the XML can be seen, it's not a very large example, but it gives an idea of how it works. In the appendix the example in it's whole can be seen.

To create this intermediate format the output from the earlier stages is used, the list from the reader, the sentence list from the NLP tool and the symbol file names from the symbol server. Depending on what was read, it's handled accordingly. Things as templates, comments and other can be sorted out, while normal text or links can be handled properly. Lists, headings, links, etc. is mapped to a corresponding tag in the XML format. Words also get mapped to a tag, with attributes for every relevant annotation from the NLP-tool and the symbol server. When all lists has been handled the XML output is written to a file.

3.5 Visualization

From the XML file it is possible to generate output for different kind of formats and devices. This can be done by stepping through the XML and handle each tag in a proper way.

```

▼<page title="Alan Turing">
  ▼<p>
    ▼<s>
      ▼<html tag="b">
        <w dep="" ne="PERSON" pos="NNP" sym="">Alan</w>
        <w dep="" ne="PERSON" pos="NNP" sym="">Mathison</w>
        <w dep="" ne="O" pos="NNP" sym="">Turing</w>
      </html>
      ▼<link target="OBE">
        <w dep="" ne="O" pos="NNP" sym="">OBE</w>
      </link>
      ▼<link target="FRS">
        <w dep="nsubj" ne="O" pos="NN" sym="">FRS</w>
      </link>
      <pun></pun>
      ▼<link target="London">
        <w dep="" ne="LOCATION" pos="NNP" sym="">London</w>
      </link>
      <pun>,</pun>
      <w dep="" ne="DATE" pos="CD" sym="">23</w>
      <w dep="" ne="DATE" pos="NNP" sym="bliss/June.png">June</w>
      <w dep="" ne="DATE" pos="CD" sym="">1912</w>
      <pun>-</pun>
      ▼<link target="Wilmslow">
        <w dep="" ne="LOCATION" pos="NNP" sym="">Wilmslow</w>
      </link>
      <pun>,</pun>
      ▼<link target="Cheshire">
        <w dep="" ne="LOCATION" pos="NNP" sym="">Cheshire</w>
      </link>
      <pun>,</pun>
      <w dep="" ne="DATE" pos="CD" sym="">7</w>
      <w dep="" ne="DATE" pos="NNP" sym="bliss/June.png">June</w>
      <w dep="" ne="DATE" pos="CD" sym="">1954</w>
      <pun></pun>
    </s>
  </p>
</page>

```

Figure 3.18: Example XML

In this thesis an example of how it might look when generating an HTML output is included. This was done by putting each word in a *div* tag and assign a class for its POS, NER and grammatical relation. The symbol goes in another *div* tag and can then be placed above the word. Since all words has a few classes it is possible to mark words of a certain word class, words that are names or words with a grammatical relation. This can be used to mark words that are more important and help the reader to understand the text better. This makes the HTML output very dynamic, making it possible to tailor the result to the target audience. To show an example of this an HTML output was generated, the result can be seen in figure 3.19, the whole example can be found in the appendix. The colors some of the words have are used to encode the root and its subject and object. Red is used as the

root, green for objects and magenta for subjects.

Alan Turing

Alan Mathison Turing OBE FRS (London, 23 June 1912 – Wimslow, Cheshire, 7

mathematician and computer scientist, born in Maida Vale, London.

He was one of the people who worked on the first computers.

He was the first person to think of using a computer to do things that we

He created the Turing machine in 1936.

Figure 3.19: An example of a HTML output

4 Evaluation

The evaluation of CCF-Symbol Server and how well the grammatical relations describes the sentences has been evaluated by me, I'm not an expert on either natural languages or Blissymbolics. I used my common sense to evaluate if these are correct or not, a lot can be figured out by looking at the names of the Blissymbolic files. The evaluation will show how good the symbol server are. The evaluation was done in a couple of steps, first I checked how many of the words had a symbol, I also evaluated if that symbol matched the word or if there was a better alternative suggested by the service. This resulted in two accuracy scores, one for the worst case in which all words with multiple suggestions got an incorrect symbol. The other score is the best case when all choices are correct.

The articles that were chosen for the evaluation are presented in figure 4.20. These articles are chosen rather arbitrarily, but some restrictions were thought of when choosing them. They should not be too short and not to large, together they should cover as much of the MediaWiki markup as possible.

- Alan Turing
- November 30
- Allsvenskan
- Pizza
- Farming
- Programming language
- International System of Units
- Star Wars
- Logic
- Super glue

Figure 4.20: The articles used in the evaluation.

4.1 Symbol Server

The following result was achieved from the evaluation, it shows that on average 75% of the word tokens have a symbol, of these symbols less than 2% are bad and has no better alternative. Since the symbol is arbitrary chosen in the case of multiple suggestions, there will be symbols that are not correctly matched, these represent up to 31% of all the symbols. Conversely, 67% of all symbols only have one alternative and it is matching the word.

In total 50% of all word tokens will have a correctly matching symbol, this is the worst case, meaning that all words with multiple symbols have been faulty chosen. In the best case, in which all choices have been correct, 73% of the text will have a correctly matching symbol.

The figures behind these numbers are presented in tables 4.4 and 4.5. Table 4.4 shows the number of good respectively bad matches when there

only is one suggestion and when there are multiple ones. In the case of multiple suggestions, it is considered bad if none of them is correct. It also shows that when there only is one alternative, it is less than 1% that it is the wrong symbol. While it is 4% chance that no alternative is correct in the case of multiple alternatives.

Description	Match	Count	Percentage	Total
Single alternative	Good	2978	99%	67.2%
	Bad	18	1%	0.4%
Multiple alternatives	Good	1380	96%	31.1%
	Bad	56	4%	1.3%

Table 4.4: Symbol suggestions

In table 4.5 numbers for each individual article can be seen, in particular the articles *November 30* and *Allsvenskan* are interesting to look at. These two articles have much lower symbol to word ratio. In *November 30*, 42% of the word tokens have a symbol and in *Allsvenskan* it's 40%. When skimming over the November article it is noticeable that many of the words without symbols are in fact numbers or names, which is an indication of why they don't have as many symbols. The same observation could be made on the *Allsvenskan* article, it also was a lot shorter then the other ones, which may be a contributing factor.

Article	Words	Symbols	Sent	Sym/Word	Word/Sent
Alan Turing	435	339	34	0.78	12.8
Allsvenskan	60	24	21	0.40	2.8
Farming	890	712	76	0.80	11.7
International Sys- tem of Units	540	413	34	0.76	15.9
Logic	324	276	32	0.85	10.1
November 30	577	246	58	0.43	9.9
Pizza	897	667	49	0.74	18.3
Programming lan- guages	1124	930	83	0.83	13.5
Star Wars	885	663	76	0.75	11.6
Super glue	192	159	11	0.83	17.5
Total	5924	4432	474	0.75	12.5

Table 4.5: Numbers for each article.

The reason for the *Allsvenskan* article being among the ones evaluated was to see how the software handled non English characters. But it also shows an other problem, sometimes names may be the same word as some-

thing else, an example is the surname *Brown* or *Smith*. They are common surnames but they have an other meaning in English as well. Brown may refer to the color brown, smith can refer to a smith (as in blacksmith). This problem may cause confusion if the symbol for the color brown appears as the surname.

An other interesting aspect of the evaluation was to look at the number of unique words, as can be seen in table 4.6. In total there were 2002 unique words, of these 960 or 48% did not have a symbol. There are 690 (34%) unique words with one alternative for a symbol and 352 (18%) unique words with multiple symbol alternatives. This adds up to 52% of the unique words having a symbol. This compared with 75% from above, is a clear indication that the word with a symbol is used more frequent than words without symbol.

During the evaluation it was noticeable that many of the words without symbols were names, numbers and other entities that should not have a symbol. By comparing the number of words without symbols to the number of unique words without symbols, we see that there are 1492 occurrences of words without symbols, of these 960 are unique. So each of these words are used on average 1.55 times. By doing the same comparison on the 1042 unique words with symbols, we have 4432 occurrences and every word is used on average 4.25 times. This indicates that the words without symbols are not as common as the ones with, this may explain why there are no symbols for them.

Description	Result
Unique words without symbols	960
Unique words with a single symbol	690
Unique words with multiple symbols	352
Total unique words	2002

Table 4.6: Number of unique words and symbol

There were also an other problem that became apparent. In a few cases, when certain words appear together with an other, the meaning of those words might be another than if they were apart. However, the evaluation was conducted on a word by word basis, so they count as matches as long they have a correct symbol for each word. Similarly, words combined with a dash(-) did not have any symbol, but they would have had if they were two separate words.

4.2 Grammatical relations

The evaluation of the grammatical relations is an even more subjective assessment. The first step was to evaluate whether the relation was accurate or not. Each sentence could have at most three relations, the direct object, the

Description		Count	Total	Percentage
Three relations	Representative	39	66	14%
	Not representative	27		
Two relations	Representative	98	177	37%
	Not representative	79		
One relation			135	28%
No relation			96	20%

Table 4.7: The results from the evaluation of the grammatical relations.

nominal subject or the root. The root is always one of them (except when there are no relations), the other relation may be either the object or the subject. It is least common with three relations, only 14% of the sentences have three relations, 37% of the sentences have two relations and 49% has one or zero relations.

The second thing that was looked upon was how well these words described the content of the sentence. Each sentence with three relations was marked as either “representative” or “not representative”. In this case 39 sentences was marked as representative, these sentences were often relatively short. The other 27 sentences with three relations, were marked as not representative. These were in general longer sentences and those three words were not enough to describe the whole sentence. In the case of two relations per sentence, the assessment was similar to the one described above, but it was a bit less strict, since it were one word less. The numbers can be seen in 4.7 and it clearly shows that there is much room for improvement.

5 Discussion

Images, tables and other visual aids are important to have from an accessibility point of view, but these have turned out to be problematic to include from Wikipedia. There are some content loss in the transformation from the article page to the intermediate format. Mainly images, tables and info boxes, which are due to MediaWiki Markup and the fact that they are created through templates. Images could probably be included fairly easy but tables and info boxes are more complicated. An other potential reason for content loss is the fact that everything on Wikipedia is user generated, this leaves much room for ambiguity. The MediaWiki Markup accepts both the markup and normal HTML, which may not be covered in the software. There are “best practices” but there are no guaranties, this is also true regarding the text content. The Simple English Wikipedia uses Basic English, but there are no guarantee all editors follow this. Sometimes Basic English lack the necessary words to describe a topic.

An alternative solution would be to, at an earlier stage in the program flow, convert everything back to MediaWiki Markup and then host it on a separate MediaWiki server. This would result in less content loss, and templates would not be a problem. Since MediaWiki has a feature to export an article and the templates it uses. But it would make it harder to implement Blissymbolics, since MediaWiki has no support for the symbols or to pair them with the words. It also would mean that it only could be used through the web, and no other formats would be possible.

5.1 Symbol coverage

The result from the evaluation are good, the Symbol Server works well with potentially 73% coverage, the worst case is rather good as well. Especially when considering that there are many words that should not have a symbol, e.g. names, numbers, etc. In fact these entities should probably be restricted to not have any symbol at all. It could be done by looking for each words NER, if it has a NER tag it should not have a symbol, and vice versa. This would also solve the problem of names getting a symbol when they double as a word.

5.2 Multi-word units

The phenomenon discovered during the evaluation is called multi-word unit (MWU). The problem when certain words, or combination of words, have one symbol when they are used together, but others when they are apart. An example of this is “living room”, when requesting the words “living” and “room” as two separate words, they get one symbol each, but the combination of the two yields a symbol that is more correct, similarly for the words “prime

minister”, but in this case prime does not yield a symbol at all.

An example of the other problem, when words were bound together with a dash, is the word “low-speed”, it does not yield any symbol, but the words “low” and “speed” gives a symbol each. The later problem is easier to solve, it would be possible to look for each of these words, and use those symbols. The MWU problem is more complicated to solve, one possibility would be to create a database over MWUs and look it up. An other, probably easier solution would be to use the Stanford parser, it has a grammatical relation for MWU, the relation bounds two words together that is an MWU. Meaning there would be a MWU relation between the word “living” and “room”.

5.3 Grammatical relations

The result of the grammatical relations is not as good as the Symbol server but I believe that it can be improved a lot, with a few simple means. Mainly by including more relations, instead of only include the nominal subject and the direct object of the root, all subjects and objects in the sentence could be incorporated. There also are some other relations that could be useful to include. For example the conjunctions to all subjects and objects, since the Stanford parser don't give them the subject/object relation. I also feel that negations would be useful to add, since it can change to meaning of the hole sentence.

5.4 Further work

The software developed in this thesis should be seen as prototype, it achieves a good result but there are room for improvements. First up, it needs to pass more articles, I expect that this will not be much of a problem, since the test articles were selected to include as much of the MediaWiki Markup as possible.

It also needs improvement regarding the content loss, images and other visual aids are important for a text to be easy to understand. Articles with missing pieces makes it harder to understand and give rise to confusion.

An other thing to implement is a symbol selector, which selects the right symbol when there are multiple ones suggested. This could possibly be done by looking at the POS, lemma and context of the word and the suggestions. In some cases the choice is obvious, it can be when there are an old and a new symbol for the same word, in this case it should just choose the new one. The grammatical relations could be improved, possibly by the suggestion in section 6.3, including more relations.

It would be very easy to change from Blissymbolcis to ARASAAC, since the CCF Symbol Server already supports it. However, if the results will be as good as for Blissymbolics is unknown.

Further, maybe most important, is to get an evaluation by people who are more familiar with Blissymbolics. It also needs to be evaluated by people from the target group. But for an evaluation by the target audience to work, a visualisation for said group needs to be implemented. Since the visualization made in this thesis is only to serve as a demonstration.

Bibliography

- [1] OECD. 2000. Literacy in the Information Age: Final Report of the International Adult Literacy Survey. *OECD Publications*, Paris.
- [2] Augmentative and alternative communication. Wikipedia. <http://en.wikipedia.org> (21 Jan. 2014)
- [3] Blissymbolics Communication International. <http://www.blissymbolics.org/> (9 Jan. 2014)
- [4] Lawton Henry, S. - McGee, L. & participants of the Education and Outreach Working Group. Accessibility. <http://www.w3.org/standards/webdesign/accessibility> (8 Jan. 2014)
- [5] Ranaldi, F. & Nisbet, P. 2010. Accessible Text: Guidelines for Good Practice. <http://www.callscotland.org.uk/Resources/Books/Accessible-Text/> (8 Jan. 2014)
- [6] Jennische, M. 2012. Characteristics of Blissymbolics. *Presentation at ISAAC research seminar*, Pittsburgh 2012.
- [7] Lundälv, M. - Derbring, S. - Brännström, A. - Farre, B. & Nordberg, L. (2013) Inclusive AAC - Multi-modal and Multilingual Language Support for All. *Assistive Technology Research Series*, vol. 33, pp. 332-339
- [8] WordNet. Princeton University. (2010) Princeton University "About WordNet." <http://wordnet.princeton.edu>
- [9] Basic English. Wikipedia. <http://en.wikipedia.org> (21 Jan. 2014)
- [10] Petrov, S. - Das, D. & McDonald, R. (2012) A Universal Part-of-Speech Tagset. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*.
- [11] Marcus, M. P. - Santorini, B. & Marcinkiewicz, M. A. (1993) Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, vol. 19, nr. 2, pp. 313-330.
- [12] Toutanova, K. - Klein, D. - Manning, C. & Singer, Y. (2003) Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. *In Proceedings of HLT-NAACL 2003*, pp. 252-259.
- [13] Toutanova, K & Manning, C. (2000) Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. *In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, pp. 63-70.

- [14] Finkel, J. R. - Grenager, T. & Manning, C. (2005) Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics* (ACL 2005), pp. 363-370.
- [15] Lemmatisation. Wikipedia. <http://en.wikipedia.org> (28 Dec. 2013)
- [16] de Marneffe, M.C. & Manning, C. (September 2008, Revised November 2013) Stanford typed dependencies manual. Stanford dependencies. <http://nlp.stanford.edu/software/stanford-dependencies.shtml> (2013-11-20)
- [17] de Marneffe, M.C. - MacCartney, B. & Manning, C. (2006) Generating Typed Dependency Parses from Phrase Structure Parses. *In LREC 2006*.
- [18] Klein, D. & Manning, C. (2003) Accurate Unlexicalized Parsing. *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pp. 423-430.
- [19] Coreference. Wikipedia. <http://en.wikipedia.org> (02 Jan. 2014)
- [20] Raghunathan, K. - Lee, H. - Rangarajan, S. - Chambers, N. - Surdeanu, M. - Jurafsky, D. & Manning, C. (2010) A Multi-Pass Sieve for Coreference Resolution. *EMNLP-2010*, Boston, USA, 2010.
- [21] Lee, H. - Peirsman, Y. - Chang, A. - Chambers, N. - Surdeanu, M. & Jurafsky, D. (2011) Stanford's Multi-Pass Sieve Coreference Resolution System at the CoNLL-2011 Shared Task. *In Proceedings of the CoNLL-2011 Shared Task*, 2011.
- [22] Lee, h. - Chang, a. - Peirsman, Y. - Chambers, N. - Surdeanu, M. & Jurafsky, D. (2013) Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics 39(4)*, 2013.
- [23] Recasens, M. - de Marneffe M.C. & Potts, C. (2013) The Life and Death of Discourse Entities: Identifying Singleton Mentions. *In Proceedings of NAACL 2013*.
- [24] Help:Wiki Markup. Wikipedia. <http://en.wikipedia.org> (10 Dec. 2013)

A Appendix

A.1 Alan Turing: Full XML example

```
<page title="Alan Turing">
  <p>
    <s>
      <html tag="b">
        <w dep="" ne="PERSON" pos="NNP" sym="">Alan</w>
        <w dep="" ne="PERSON" pos="NNP" sym="">Mathison</w>
        <w dep="" ne="O" pos="NNP" sym="">Turing</w>
      </html>
      <link target="OBE">
        <w dep="" ne="O" pos="NNP" sym="">OBE</w>
      </link>
      <link target="FRS">
        <w dep="nsubj" ne="O" pos="NN" sym="">FRS</w>
      </link>
      <pun></pun>
      <link target="London">
        <w dep="" ne="LOCATION" pos="NNP" sym="">London</w>
      </link>
      <pun>,</pun>
      <w dep="" ne="DATE" pos="CD" sym="">23</w>
      <w dep="" ne="DATE" pos="NNP" sym="bliss/June.png">June</w>
      <w dep="" ne="DATE" pos="CD" sym="">1912</w>
      <pun>-</pun>
      <link target="Wilmslow">
        <w dep="" ne="LOCATION" pos="NNP" sym="">Wilmslow</w>
      </link>
      <pun>,</pun>
      <link target="Cheshire">
        <w dep="" ne="LOCATION" pos="NNP" sym="">Cheshire</w>
      </link>
      <pun>,</pun>
      <w dep="" ne="DATE" pos="CD" sym="">7</w>
      <w dep="" ne="DATE" pos="NNP" sym="bliss/June.png">June</w>
      <w dep="" ne="DATE" pos="CD" sym="">1954</w>
      <pun>)</pun>
      <w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
      <w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">an</w>
      <link target="English people">
        <w dep="" ne="MISC" pos="JJ" sym="bliss/English_(language).png">English</w>
      </link>
      <link target="mathematician">
        <w dep="root" ne="O" pos="NN"
          sym="bliss/mathematician_(academic).png">mathematician</w>
      </link>
      <w dep="" ne="O" pos="CC" sym="bliss/and,also,plus,too.png">and</w>
      <link target="computer scientist">
        <w dep="" ne="O" pos="NN" sym="bliss/computer.png">computer</w>
        <w dep="" ne="O" pos="NN" sym="bliss/scientist,researcher.png">scientist</w>
      </link>
      <pun>,</pun>
      <w dep="" ne="O" pos="VBN" sym="">born</w>
      <w dep="" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
      <link target="Maida Vale">
        <w dep="" ne="LOCATION" pos="NNP" sym="">Maida</w>
        <w dep="" ne="LOCATION" pos="NNP" sym="">Vale</w>
      </link>
      <pun>,</pun>
      <w dep="" ne="LOCATION" pos="NNP" sym="">London</w>
      <pun>.</pun>
    </s>
  </p>
  <p>
```

```
<s>
<w dep="nsubj" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
<w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
<w dep="root" ne="NUMBER" pos="CD" sym="bliss/one_(digit).png">one</w>
<w dep="" ne="O" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
<w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
<w dep="" ne="O" pos="NNS" sym="bliss/people,tribe,clan,folk.png">people</w>
<w dep="" ne="O" pos="WP" sym="bliss/who,whom,that-(relative).png">who</w>
<w dep="" ne="O" pos="VBD" sym="bliss/work-(to).png">worked</w>
<w dep="" ne="O" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">on</w>
<w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
<w dep="" ne="ORDINAL" pos="JJ" sym="bliss/first,primary.png">first</w>
<link target="computers">
  <w dep="" ne="O" pos="NNS" sym="bliss/computer.png">computers</w>
</link>
<pun>.</pun>
</s>
```

```
<s>
<w dep="nsubj" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
<w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
<w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
<w dep="" ne="ORDINAL" pos="JJ" sym="bliss/first,primary.png">first</w>
<w dep="root" ne="O" pos="NN"
sym="bliss/person,human_being,individual,human.png">person</w>
<w dep="" ne="O" pos="TO" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="O" pos="VB" sym="bliss/think,reason-(to).png">think</w>
<w dep="" ne="O" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
<w dep="" ne="O" pos="VBG" sym="bliss/use-(to).png">using</w>
<w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
<w dep="" ne="O" pos="NN" sym="bliss/computer.png">computer</w>
<w dep="" ne="O" pos="TO" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="O" pos="VB" sym="bliss/do,act-(to).png">do</w>
<w dep="" ne="O" pos="NNS" sym="bliss/thing,object.png">things</w>
<w dep="" ne="O" pos="WDT" sym="bliss/that_(there).png">that</w>
<w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">were</w>
<w dep="" ne="O" pos="RB" sym="bliss/and,also,plus,too.png">too</w>
<w dep="" ne="O" pos="JJ" sym="bliss/difficult,hard.png">hard</w>
<w dep="" ne="O" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
<w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
<w dep="" ne="O" pos="NN"
sym="bliss/person,human_being,individual,human.png">person</w>
<w dep="" ne="O" pos="TO" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="O" pos="VB" sym="bliss/do,act-(to).png">do</w>
<pun>.</pun>
</s>
```

```
<s>
<w dep="nsubj" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
<w dep="root" ne="O" pos="VBD" sym="bliss/create-(to).png">created</w>
<w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
<link target="Turing machine">
  <w dep="" ne="MISC" pos="JJ" sym="">Turing</w>
  <w dep="dobj" ne="O" pos="NN"
sym="bliss/machine,appliance,engine,motor.png">machine</w>
</link>
<w dep="" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
<w dep="" ne="DATE" pos="CD" sym="">1936</w>
<pun>.</pun>
</s>
```

```
<s>
<w dep="" ne="O" pos="DT" sym="bliss/the.png">The</w>
<w dep="nsubj" ne="O" pos="NN"
sym="bliss/machine,appliance,engine,motor.png">machine</w>
```

```

<w dep="" ne="0" pos="VBD" sym="bliss/be, am, are, is, exist-(to).png">was</w>
<link target="Imagination">
  <w dep="root" ne="0" pos="JJ" sym="bliss/make-
  believe, pretend, imaginary.png">imaginary</w>
</link>
<pun>, </pun>
<w dep="" ne="0" pos="CC" sym="bliss/but, except.png">but</w>
<w dep="" ne="0" pos="PRP" sym="bliss/it.png">it</w>
<w dep="" ne="0" pos="VBD" sym="bliss/include-(to).png">included</w>
<w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
<w dep="" ne="0" pos="NN" sym="bliss/idea, thought.png">idea</w>
<w dep="" ne="0" pos="IN"
sym="bliss/about, concerning, in_relation_to, of, on.png">of</w>
<w dep="" ne="0" pos="DT" sym="bliss/a, an, any.png">a</w>
<link target="Computer program">
  <w dep="" ne="0" pos="NN" sym="bliss/program, programme.png">program</w>
</link>
<pun>.</pun>
</s>
</p>
<p>
<s>
  <w dep="nsubj" ne="0" pos="VBG" sym="">Turing</w>
  <w dep="" ne="0" pos="VBD" sym="bliss/be, am, are, is, exist-(to).png">was</w>
  <w dep="root" ne="0" pos="JJ" sym="bliss/interesting, interested.png">interested</w>
  <w dep="" ne="0" pos="IN" sym="bliss/in, inside, interior, internal.png">in</w>
  <link target="artificial intelligence">
    <w dep="" ne="0" pos="JJ" sym="">artificial</w>
    <w dep="" ne="0" pos="NN" sym="">intelligence</w>
  </link>
  <pun>.</pun>
</s>
<s>
  <w dep="nsubj" ne="0" pos="PRP" sym="bliss/he, she, him, her, one.png">He</w>
  <w dep="root" ne="0" pos="VBD" sym="bliss/suggest, propose-(to).png">proposed</w>
  <w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
  <link target="Turing test">
    <w dep="" ne="MISC" pos="JJ" sym="">Turing</w>
    <w dep="dobj" ne="0" pos="NN" sym="bliss/test, assessment, exam.png">test</w>
  </link>
  <pun>, </pun>
  <w dep="" ne="0" pos="TO" sym="bliss/until, till, to.png">to</w>
  <w dep="" ne="0" pos="VB" sym="bliss/say, speak, talk, tell, express-(to).png">say</w>
  <w dep="" ne="0" pos="WRB" sym="bliss/when, what_time-(question).png">when</w>
  <w dep="" ne="0" pos="DT" sym="bliss/a, an, any.png">a</w>
  <w dep="" ne="0" pos="NN" sym="bliss/machine, appliance, engine, motor.png">machine</w>
  <w dep="" ne="0" pos="MD" sym="bliss/can, be_able-(to).png">could</w>
  <w dep="" ne="0" pos="VB" sym="bliss/be, am, are, is, exist-(to).png">be</w>
  <w dep="" ne="0" pos="VBN" sym="bliss/call, telephone, ring-(to).png">called</w>
  <pun>"</pun>
  <w dep="" ne="0" pos="JJ"
  sym="bliss/smart, bright, clever, intelligent.png">intelligent</w>
  <pun>"</pun>
  <pun>.</pun>
</s>
<s>
  <w dep="" ne="0" pos="DT" sym="bliss/a, an, any.png">A</w>
  <w dep="" ne="0" pos="NN" sym="bliss/computer.png">computer</w>
  <w dep="" ne="0" pos="MD" sym="bliss/can, be_able-(to).png">could</w>
  <w dep="" ne="0" pos="VB" sym="bliss/be, am, are, is, exist-(to).png">be</w>
  <w dep="root" ne="0" pos="VBN" sym="bliss/say, speak, talk, tell, express-
  (to).png">said</w>
  <w dep="" ne="0" pos="TO" sym="bliss/until, till, to.png">to</w>
  <pun>"</pun>
  <w dep="" ne="0" pos="VB" sym="bliss/think, reason-(to).png">think</w>

```

<pun></pun>
<w dep="" ne="0" pos="IN" sym="bliss/if.png">if</w>
<w dep="" ne="0" pos="DT" sym="bliss/a, an, any.png">a</w>
<w dep="" ne="0" pos="JJ"
sym="bliss/person, human_being, individual, human.png">human</w>
<w dep="" ne="0" pos="NN" sym="">interrogator</w>
<w dep="" ne="0" pos="MD" sym="bliss/can, be_able- (to).png">could</w>
<w dep="" ne="0" pos="RB" sym="bliss/not, negative, no, don't, doesn't.png">not</w>
<w dep="" ne="0" pos="VB" sym="bliss/say, speak, talk, tell, express- (to).png">tell</w>
<w dep="" ne="0" pos="PRP" sym="bliss/it.png">it</w>
<w dep="" ne="0" pos="RB" sym="">apart</w>
<pun>,</pun>
<w dep="" ne="0" pos="IN" sym="bliss/through.png">through</w>
<w dep="" ne="0" pos="NN"
sym="bliss/discussion, conversation, debate, chat.png">conversation</w>
<pun>,</pun>
<w dep="" ne="0" pos="IN" sym="bliss/from.png">from</w>
<w dep="" ne="0" pos="DT" sym="bliss/a, an, any.png">a</w>
<w dep="" ne="0" pos="JJ"
sym="bliss/person, human_being, individual, human.png">human</w>
<w dep="" ne="0" pos="NN" sym="bliss/creature, being.png">being</w>
<pun>.</pun>
</s>
</p>
<p>
<s>
<w dep="" ne="0" pos="IN" sym="bliss/during, while.png">During</w>
<link target="World War II">
<w dep="" ne="MISC" pos="NNP" sym="bliss/earth, globe, world.png">World</w>
<w dep="" ne="MISC" pos="NNP" sym="bliss/war.png">War</w>
<w dep="" ne="MISC" pos="NNP" sym="">II</w>
</link>
<pun>,</pun>
<w dep="nsubj" ne="PERSON" pos="NNP" sym="">Turing</w>
<w dep="root" ne="0" pos="VBD" sym="bliss/work- (to).png">worked</w>
<w dep="" ne="0" pos="TO" sym="bliss/until, till, to.png">to</w>
<w dep="" ne="0" pos="VB" sym="bliss/break, fracture.png">break</w>
<link target="Germany">
<w dep="" ne="MISC" pos="JJ" sym="bliss/German_ (language).png">German</w>
</link>
<link target="cipher">
<w dep="" ne="0" pos="NNS" sym="bliss/cipher.png">ciphers</w>
</link>
<pun>(</pun>
<w dep="" ne="0" pos="JJ" sym="bliss/secret.png">secret</w>
<w dep="" ne="0" pos="NNS" sym="bliss/message.png">messages</w>
<pun>)</pun>
<pun>.</pun>
</s>
<s>
<w dep="" ne="0" pos="VBG" sym="bliss/use- (to).png">Using</w>
<link target="cryptanalysis">
<w dep="" ne="0" pos="NN" sym="">cryptanalysis</w>
</link>
<w dep="nsubj" ne="0" pos="PRP" sym="bliss/he, she, him, her, one.png">he</w>
<w dep="root" ne="0" pos="VBD" sym="bliss/help, aid, assist, serve, support- (to).png">helped</w>
<w dep="" ne="0" pos="TO" sym="bliss/until, till, to.png">to</w>
<w dep="" ne="0" pos="VB" sym="bliss/break, fracture.png">break</w>
<w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
<link target="Enigma (machine)">
<w dep="" ne="0" pos="NN" sym="">Enigma</w>
</link>
<w dep="" ne="0" pos="NN" sym="bliss/machine, appliance, engine, motor.png">machine</w>
<pun>.</pun>

```

</s>
<s>
  <w dep="" ne="O" pos="IN" sym="bliss/after,behind.png">After</w>
  <w dep="" ne="O" pos="DT" sym="bliss/that_(there).png">that</w>
  <pun>,</pun>
  <w dep="nsubj" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">he</w>
  <w dep="root" ne="O" pos="VBD" sym="bliss/solve-(to).png">solved</w>
  <w dep="" ne="O" pos="JJ" sym="bliss/other,another_(thing).png">other</w>
  <w dep="" ne="MISC" pos="JJ" sym="bliss/German_(language).png">German</w>
  <w dep="dobj" ne="O" pos="NNS" sym="bliss/sign,signal.png">signals</w>
  <pun>.</pun>
</s>
</p>
<p>
  <s>
    <w dep="" ne="O" pos="IN" sym="bliss/from.png">From</w>
    <w dep="" ne="DATE" pos="CD" sym="">1945</w>
    <w dep="" ne="DATE" pos="TO" sym="bliss/until,till,to.png">to</w>
    <w dep="" ne="DATE" pos="CD" sym="">1947</w>
    <w dep="nsubj" ne="O" pos="NN" sym="">Turing</w>
    <w dep="root" ne="O" pos="VBD" sym="bliss/work-(to).png">worked</w>
    <w dep="" ne="O" pos="IN"
  sym="bliss/about,concerning,in_relation_to,of,on.png">on</w>
    <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
    <w dep="" ne="O" pos="NN" sym="bliss/plan,design,method,system.png">design</w>
    <w dep="" ne="O" pos="IN"
  sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
    <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
    <link target="ACE (computer)">
      <w dep="" ne="O" pos="NN" sym="">ACE</w>
    </link>
    <pun>(</pun>
    <w dep="" ne="O" pos="NNP" sym="">Automatic</w>
    <w dep="" ne="O" pos="NNP" sym="">Computing</w>
    <w dep="" ne="O" pos="NNP" sym="bliss/machine,appliance,engine,motor.png">Engine</w>
    <pun>)</pun>
    <w dep="" ne="O" pos="IN" sym="bliss/at.png">at</w>
    <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
    <link target="National Physical Laboratory">
      <w dep="" ne="O" pos="NNP" sym="bliss/national.png">National</w>
      <w dep="" ne="O" pos="NNP" sym="">Physical</w>
      <w dep="" ne="O" pos="NNP" sym="">Laboratory</w>
    </link>
    <pun>.</pun>
  </s>
  <s>
    <w dep="nsubj" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
    <w dep="root" ne="O" pos="VBD" sym="bliss/introduce,present-(to).png">presented</w>
    <w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
    <w dep="dobj" ne="O" pos="NN" sym="bliss/paper,card,page.png">paper</w>
    <w dep="" ne="O" pos="IN"
  sym="bliss/about,concerning,in_relation_to,of,on.png">on</w>
    <w dep="" ne="DATE" pos="CD" sym="">19</w>
    <w dep="" ne="DATE" pos="NNP" sym="bliss/February.png">February</w>
    <w dep="" ne="DATE" pos="CD" sym="">1946</w>
    <pun>,</pun>
    <w dep="" ne="O" pos="WDT" sym="bliss/which-(question).png">which</w>
    <w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
    <html tag="i">
      <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
      <w dep="" ne="ORDINAL" pos="JJ" sym="bliss/first,primary.png">first</w>
      <w dep="" ne="O" pos="JJ" sym="">detailed</w>
      <w dep="" ne="O" pos="NN" sym="bliss/plan,design,method,system.png">design</w>
      <w dep="" ne="O" pos="IN"
    sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
  </html>
  </s>

```

```
<w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
<link target="stored-program computer">
  <w dep="" ne="O" pos="JJ" sym="">stored-program</w>
  <w dep="" ne="O" pos="NN" sym="bliss/computer.png">computer</w>
</link>
</html>
<pun>.</pun>
</s>
<s>
  <w dep="" ne="O" pos="IN" sym="bliss/although.png">Although</w>
  <w dep="" ne="O" pos="NN" sym="">ACE</w>
  <w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
  <w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
  <w dep="" ne="O" pos="JJ" sym="">feasible</w>
  <w dep="" ne="O" pos="NN" sym="bliss/plan,design,method,system.png">design</w>
<pun>,</pun>
  <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
  <w dep="nsubj" ne="O" pos="NN" sym="">secrecy</w>
  <w dep="" ne="O" pos="VBG" sym="">surrounding</w>
  <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
  <w dep="" ne="O" pos="NN" sym="">wartime</w>
  <w dep="" ne="O" pos="NN" sym="bliss/work,employment,job.png">work</w>
  <w dep="" ne="O" pos="IN" sym="bliss/at.png">at</w>
  <link target="Bletchley Park">
    <w dep="" ne="LOCATION" pos="NNP" sym="">Bletchley</w>
    <w dep="" ne="LOCATION" pos="NNP" sym="bliss/park.png">Park</w>
  </link>
  <w dep="root" ne="O" pos="VBD" sym="bliss/lead,direct,guide-(to).png">led</w>
  <w dep="" ne="O" pos="TO" sym="bliss/until,till,to.png">to</w>
  <w dep="" ne="O" pos="NNS" sym="">delays</w>
  <w dep="" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
  <w dep="" ne="O" pos="VBG" sym="bliss/begin,start-(to).png">starting</w>
  <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
  <w dep="" ne="O" pos="NN" sym="">project</w>
  <w dep="" ne="O" pos="CC" sym="bliss/and,also,plus,too.png">and</w>
  <w dep="" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">he</w>
  <w dep="" ne="O" pos="VBD" sym="bliss/become-(to).png">became</w>
  <w dep="" ne="O" pos="JJ" sym="">disillusioned</w>
<pun>.</pun>
</s>
<s>
  <w dep="" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">In</w>
  <w dep="" ne="DATE" pos="JJ" sym="bliss/late.png">late</w>
  <w dep="" ne="DATE" pos="CD" sym="">1947</w>
  <w dep="nsubj" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">he</w>
  <w dep="root" ne="O" pos="VBD" sym="bliss/return,come_back,reverse-(to).png">returned</w>
  <w dep="" ne="O" pos="TO" sym="bliss/until,till,to.png">to</w>
  <w dep="" ne="LOCATION" pos="NNP" sym="">Cambridge</w>
  <w dep="" ne="O" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
  <w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
  <w dep="" ne="O" pos="JJ" sym="">sabbatical</w>
  <w dep="" ne="DURATION" pos="NN" sym="bliss/year.png">year</w>
<pun>.</pun>
</s>
<s>
  <w dep="" ne="O" pos="IN" sym="bliss/during,while.png">While</w>
  <w dep="" ne="O" pos="PRP" sym="bliss/he,she,him,her,one.png">he</w>
  <w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
  <w dep="" ne="O" pos="IN" sym="bliss/at.png">at</w>
  <w dep="" ne="LOCATION" pos="NNP" sym="">Cambridge</w>
<pun>,</pun>
  <w dep="" ne="O" pos="DT" sym="bliss/the.png">the</w>
  <link target="Pilot ACE">
    <w dep="" ne="O" pos="NN" sym="bliss/pilot.png">Pilot</w>

```

<w dep="" ne="O" pos="NN" sym="">ACE</w>
</link>
<w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
<w dep="root" ne="O" pos="VBN" sym="bliss/build,construct-(to).png">built</w>
<w dep="" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
<w dep="" ne="O" pos="PRP\$" sym="bliss/his,her,hers,one's.png">his</w>
<w dep="" ne="O" pos="NN" sym="">absence</w>
<pun>.</pun>
</s>
<s>
<w dep="nsubj" ne="O" pos="PRP" sym="bliss/it.png">It</w>
<w dep="root" ne="O" pos="VBD" sym="">executed</w>
<w dep="" ne="O" pos="PRP\$" sym="bliss/its.png">its</w>
<w dep="" ne="ORDINAL" pos="JJ" sym="bliss/first,primary.png">first</w>
<w dep="dobj" ne="O" pos="NN" sym="bliss/program,programme.png">program</w>
<w dep="" ne="O" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">on</w>
<w dep="" ne="DATE" pos="CD" sym="">10</w>
<w dep="" ne="DATE" pos="NNP" sym="bliss/May.png">May</w>
<w dep="" ne="DATE" pos="CD" sym="">1950</w>
<pun>.</pun>
</s>
</p>
<p>
<head size="2">Private life</head>
<s>
<w dep="nsubj" ne="PERSON" pos="VBG" sym="">Turing</w>
<w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
<w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
<link target="gay">
<w dep="" ne="O" pos="JJ" sym="bliss/homosexual.png">gay</w>
</link>
<w dep="root" ne="O" pos="NN" sym="bliss/man,male.png">man</w>
<pun>.</pun>
</s>
<s>
<w dep="root" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">In</w>
<w dep="" ne="DATE" pos="CD" sym="">1952</w>
<pun>,</pun>
<w dep="" ne="PERSON" pos="VBG" sym="">Turing</w>
<w dep="" ne="O" pos="VBN" sym="">admitted</w>
<w dep="" ne="O" pos="VBG" sym="bliss/have-(to).png">having</w>
<w dep="" ne="O" pos="VBD" sym="bliss/have-(to).png">had</w>
<w dep="" ne="O" pos="NN" sym="bliss/gender,sex.png">sex</w>
<w dep="" ne="O" pos="IN" sym="bliss/with.png">with</w>
<w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
<w dep="" ne="O" pos="NN" sym="bliss/man,male.png">man</w>
<w dep="" ne="O" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
<link target="England">
<w dep="" ne="LOCATION" pos="NNP" sym="bliss/England.png">England</w>
</link>
<pun>.</pun>
</s>
<s>
<w dep="" ne="O" pos="IN" sym="bliss/at.png">At</w>
<w dep="" ne="O" pos="DT" sym="bliss/that_(there).png">that</w>
<w dep="" ne="O" pos="NN" sym="bliss/time.png">time</w>
<pun>,</pun>
<w dep="nsubj" ne="O" pos="PRP" sym="bliss/it.png">it</w>
<w dep="" ne="O" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
<w dep="" ne="O" pos="DT" sym="bliss/a,an,any.png">a</w>
<w dep="root" ne="O" pos="NN" sym="bliss/crime.png">crime</w>
<w dep="" ne="O" pos="TO" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="O" pos="VB" sym="bliss/take,bring,carry,move-(to).png">take</w>
<w dep="" ne="O" pos="NN" sym="bliss/part,bit,piece,portion,part_of.png">part</w>

<w dep="" ne="0" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
 <link target="homosexual">
 <w dep="" ne="0" pos="JJ" sym="bliss/homosexual.png">homosexual</w>
 </link>
 <w dep="" ne="0" pos="NNS" sym="bliss/action,demonstration_(in_favour_of).png">acts</w>
 <pun>.</pun>
 </s>
 <s>
 <w dep="" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
 <w dep="" ne="0" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
 <w dep="root" ne="0" pos="VBN" sym="">convicted</w>
 <w dep="" ne="0" pos="CC" sym="bliss/and,also,plus,too.png">and</w>
 <w dep="" ne="0" pos="VBN" sym="bliss/forced,compelled,obliged.png">forced</w>
 <w dep="" ne="0" pos="TO" sym="bliss/until,till,to.png">to</w>
 <w dep="" ne="0" pos="VB" sym="bliss/make,manufacture,produce-(to).png">make</w>
 <w dep="" ne="0" pos="DT" sym="bliss/a,an,any.png">a</w>
 <w dep="" ne="0" pos="NN" sym="bliss/choice,selection.png">choice</w>
 <pun>.</pun>
 </s>
 <s>
 <w dep="nsubj" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
 <w dep="" ne="0" pos="MD" sym="bliss/can,be_able-(to).png">could</w>
 <w dep="root" ne="0" pos="VB" sym="bliss/choose,pick,select-(to).png">choose</w>
 <w dep="" ne="0" pos="IN" sym="bliss/between.png">between</w>
 <w dep="" ne="0" pos="VBG" sym="bliss/go,depart,leave-(to).png">going</w>
 <w dep="" ne="0" pos="TO" sym="bliss/until,till,to.png">to</w>
 <link target="jail">
 <w dep="" ne="0" pos="NN" sym="bliss/prison,jail.png">jail</w>
 </link>
 <w dep="" ne="0" pos="CC" sym="bliss/or.png">or</w>
 <w dep="" ne="0" pos="VB" sym="bliss/take,bring,carry,move-(to).png">take</w>
 <link target="hormone">
 <w dep="" ne="0" pos="NNS" sym="">hormones</w>
 </link>
 <w dep="" ne="0" pos="IN" sym="bliss/like-(to).png">like</w>
 <link target="oestrogen">
 <w dep="" ne="0" pos="NN" sym="">estrogen</w>
 </link>
 <w dep="" ne="0" pos="TO" sym="bliss/until,till,to.png">to</w>
 <w dep="" ne="0" pos="VB" sym="bliss/lower-(to).png">lower</w>
 <w dep="" ne="0" pos="PRP\$" sym="bliss/his,her,hers,one's.png">his</w>
 <w dep="" ne="0" pos="NN" sym="bliss/gender,sex.png">sex</w>
 <w dep="" ne="0" pos="NN" sym="bliss/drive-(to).png">drive</w>
 <pun>.</pun>
 </s>
 <s>
 <w dep="nsubj" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
 <w dep="root" ne="0" pos="VBD" sym="bliss/decide-(to).png">decided</w>
 <w dep="" ne="0" pos="TO" sym="bliss/until,till,to.png">to</w>
 <w dep="" ne="0" pos="VB" sym="bliss/choose,pick,select-(to).png">choose</w>
 <w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
 <w dep="" ne="0" pos="NNS" sym="">hormones</w>
 <pun>.</pun>
 </s>
 <s>
 <w dep="" ne="0" pos="IN" sym="bliss/after,behind.png">After</w>
 <w dep="" ne="0" pos="PRP\$" sym="bliss/his,her,hers,one's.png">his</w>
 <w dep="" ne="0" pos="NN" sym="">punishment</w>
 <pun>,</pun>
 <w dep="nsubj" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">he</w>
 <w dep="root" ne="0" pos="VBD" sym="bliss/become-(to).png">became</w>
 <w dep="" ne="0" pos="JJ" sym="">impotent</w>
 <w dep="" ne="0" pos="CC" sym="bliss/and,also,plus,too.png">and</w>
 <w dep="" ne="0" pos="MD" sym="bliss/can,be_able-(to).png">could</w>

```
<w dep="" ne="0" pos="RB" sym="bliss/not,negative,no,don't,doesn't.png">not</w>
<w dep="" ne="0" pos="VB" sym="bliss/get,acquire,receive-(to).png">get</w>
<w dep="" ne="0" pos="DT" sym="bliss/a,an,any.png">an</w>
<link target="erection">
  <w dep="" ne="0" pos="NN" sym="bliss/erection,erect_penis.png">erection</w>
</link>
<pun>.</pun>
</s>
<s>
  <w dep="nsubj" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">He</w>
  <w dep="" ne="0" pos="RB" sym="bliss/and,also,plus,too.png">also</w>
  <w dep="root" ne="0" pos="VBD" sym="bliss/grow-(to).png">grew</w>
  <link target="breast">
    <w dep="dobj" ne="0" pos="NNS" sym="bliss/breasts.png">breasts</w>
  </link>
  <pun>.</pun>
</s>
</p>
<p>
  <head size="2">Death</head>
  <s>
    <w dep="" ne="0" pos="IN" sym="bliss/in,inside,interior,internal.png">In</w>
    <w dep="" ne="DATE" pos="CD" sym="">1954</w>
    <pun>,</pun>
    <w dep="" ne="0" pos="IN" sym="bliss/after,behind.png">after</w>
    <w dep="" ne="0" pos="VBG" sym="bliss/pain,suffering.png">suffering</w>
    <w dep="" ne="0" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
    <w dep="" ne="DURATION" pos="CD" sym="bliss/two_(digit).png">two</w>
    <w dep="" ne="NUMBER" pos="NNS" sym="bliss/year.png">years</w>
    <pun>,</pun>
    <w dep="nsubj" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">he</w>
    <w dep="root" ne="0" pos="VBD" sym="bliss/croak_(frog)-(to).png">died</w>
    <w dep="" ne="0" pos="IN" sym="bliss/after,behind.png">after</w>
    <w dep="" ne="0" pos="VBG" sym="bliss/eat-(to).png">eating</w>
    <w dep="" ne="0" pos="DT" sym="bliss/a,an,any.png">an</w>
    <link target="apple">
      <w dep="" ne="0" pos="NN" sym="bliss/apple.png">apple</w>
    </link>
    <w dep="" ne="0" pos="WDT" sym="bliss/which-(question).png">which</w>
    <w dep="" ne="0" pos="VBD" sym="bliss/be,am,are,is,exist-(to).png">was</w>
    <w dep="" ne="0" pos="VBN" sym="">poisoned</w>
    <w dep="" ne="0" pos="IN" sym="bliss/with.png">with</w>
    <link target="cyanide">
      <w dep="" ne="0" pos="NN" sym="">cyanide</w>
    </link>
    <pun>,</pun>
    <w dep="" ne="0" pos="CC" sym="bliss/or.png">or</w>
    <w dep="" ne="0" pos="VBG" sym="bliss/drink-(to).png">drinking</w>
    <w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
    <w dep="" ne="0" pos="NN" sym="">cyanide</w>
    <w dep="" ne="0" pos="IN" sym="bliss/in,inside,interior,internal.png">in</w>
    <w dep="" ne="0" pos="NN" sym="bliss/water,fluid,liquid.png">water</w>
    <pun>.</pun>
  </s>
  <s>
    <w dep="" ne="0" pos="DT" sym="bliss/the.png">The</w>
    <w dep="nsubj" ne="0" pos="NN" sym="bliss/mind,intellect,reason.png">reason</w>
    <w dep="" ne="0" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
    <w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
    <w dep="" ne="0" pos="NN" sym="">confusion</w>
    <w dep="root" ne="0" pos="VBZ" sym="bliss/be,am,are,is,exist-(to).png">is</w>
    <w dep="" ne="0" pos="IN" sym="bliss/that_(there).png">that</w>
    <w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
    <w dep="" ne="0" pos="NN" sym="bliss/police_force,police.png">police</w>
    <w dep="" ne="0" pos="RB" sym="bliss/never.png">never</w>
  </s>
</p>
```

<w dep="" ne="0" pos="VBD" sym="">tested</w>
<w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
<w dep="" ne="0" pos="NN" sym="bliss/apple.png">apple</w>
<w dep="" ne="0" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
<w dep="" ne="0" pos="NN" sym="">cyanide</w>
<pun>.</pun>
</s>
</p>
<p>
<s>
<w dep="" ne="0" pos="DT" sym="bliss/the.png">The</w>
<w dep="" ne="0" pos="NN" sym="bliss/treatment.png">treatment</w>
<w dep="" ne="0" pos="VBN" sym="bliss/forced,compelled,obliged.png">forced</w>
<w dep="" ne="0" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">on</w>
<w dep="" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">him</w>
<w dep="" ne="0" pos="VBZ" sym="bliss/be,am,are,is,exist-(to).png">is</w>
<w dep="" ne="DATE" pos="RB" sym="bliss/now.png">now</w>
<w dep="root" ne="0" pos="VBN" sym="bliss/believe-(to).png">believed</w>
<w dep="" ne="0" pos="TO" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="0" pos="VB" sym="bliss/be,am,are,is,exist-(to).png">be</w>
<w dep="" ne="0" pos="RB" sym="bliss/much,many,very.png">very</w>
<w dep="" ne="0" pos="JJ" sym="bliss/incorrect,bad,inaccurate,wrong.png">wrong</w>
<pun>,</pun>
<w dep="" ne="0" pos="VBG" sym="bliss/go,depart,leave-(to).png">going</w>
<w dep="" ne="0" pos="IN" sym="bliss/against,opposed_to.png">against</w>
<link target="medical ethics">
<w dep="" ne="0" pos="JJ" sym="bliss/medical,medically.png">medical</w>
<w dep="" ne="0" pos="NNS" sym="">ethics</w>
</link>
<w dep="" ne="0" pos="CC" sym="bliss/and,also,plus,too.png">and</w>
<w dep="" ne="0" pos="JJ" sym="bliss/international.png">international</w>
<w dep="" ne="0" pos="NNS" sym="bliss/judgement,law_(in_combinations).png">laws</w>
<w dep="" ne="0" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
<link target="human rights">
<w dep="" ne="0" pos="JJ"
sym="bliss/person,human_being,individual,human.png">human</w>
<w dep="" ne="0" pos="NNS" sym="bliss/right.png">rights</w>
</link>
<pun>.</pun>
</s>
<s>
<w dep="" ne="0" pos="IN" sym="bliss/in,inside,interior,internal.png">In</w>
<w dep="" ne="DATE" pos="NNP" sym="bliss/August.png">August</w>
<w dep="" ne="DATE" pos="CD" sym="">2009</w>
<pun>,</pun>
<w dep="" ne="0" pos="DT" sym="bliss/a,an,any.png">a</w>
<link target="petition">
<w dep="nsubj" ne="0" pos="NN" sym="">petition</w>
</link>
<w dep="root" ne="0" pos="VBD" sym="bliss/apologize,apologise,regret_(express)-
(to).png">apologised</w>
<w dep="" ne="0" pos="TO" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="0" pos="VBG" sym="">Turing</w>
<w dep="" ne="0" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
<w dep="" ne="0" pos="VBG" sym="">punishing</w>
<w dep="" ne="0" pos="PRP" sym="bliss/he,she,him,her,one.png">him</w>
<w dep="" ne="0" pos="IN" sym="bliss/for_(in_exchange_for).png">for</w>
<w dep="" ne="0" pos="VBG" sym="bliss/creature,being.png">being</w>
<w dep="" ne="0" pos="DT" sym="bliss/a,an,any.png">a</w>
<w dep="" ne="0" pos="NN" sym="bliss/homosexual.png">homosexual</w>
<pun>.</pun>
</s>
<s>

```

<w dep="" ne="0" pos="DT" sym="bliss/the.png">The</w>
<w dep="nsubj" ne="0" pos="NN" sym="">petition</w>
<w dep="root" ne="0" pos="VBD" sym="bliss/get,acquire,receive-(to).png">received</w>
<w dep="dobj" ne="0" pos="NNS" sym="">thousands</w>
<w dep="" ne="0" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
<w dep="" ne="0" pos="NNS" sym="bliss/signature.png">signatures</w>
<pun>.</pun>
</s>
<s>
<w dep="" ne="0" pos="NNP" sym="">Prime</w>
<w dep="" ne="0" pos="NNP"
sym="bliss/minister,pastor,preacher,priest,rabbi.png">Minister</w>
<link target="Gordon Brown">
  <w dep="" ne="PERSON" pos="NNP" sym="">Gordon</w>
  <w dep="nsubj" ne="PERSON" pos="NNP" sym="bliss/brown_(bci).png">Brown</w>
</link>
<w dep="root" ne="0" pos="VBD" sym="">acknowledged</w>
<w dep="" ne="0" pos="DT" sym="bliss/the.png">the</w>
<w dep="" ne="0" pos="NN" sym="">petition</w>
<pun>,</pun>
<w dep="" ne="0" pos="CC" sym="bliss/and,also,plus,too.png">and</w>
<w dep="" ne="0" pos="VBN" sym="bliss/call,telephone,ring-(to).png">called</w>
<w dep="" ne="0" pos="NNP" sym="">Turing</w>
<w dep="" ne="0" pos="POS" sym="bliss/be,am,are,is,exist-(to).png">'s</w>
<w dep="" ne="0" pos="NN" sym="bliss/treatment.png">treatment</w>
<pun>"</pun>
<w dep="" ne="0" pos="JJ" sym="">appalling</w>
<pun>"</pun>
<pun>:</pun>
</s>
</p>
<p>
<head size="2">References</head>
<ref>
<s>
  <w dep="" ne="" pos="" sym="">Newman</w>
  <w dep="" ne="" pos="" sym="">M</w>
  <pun>.</pun>
  <w dep="" ne="" pos="" sym="">H</w>
  <pun>.</pun>
  <w dep="" ne="" pos="" sym="bliss/a,an,any.png">A</w>
  <pun>.</pun>
  <w dep="" ne="" pos="" sym="">1955</w>
  <pun>.</pun>
  <w dep="" ne="" pos="" sym="">Alan</w>
  <w dep="" ne="" pos="" sym="">Mathison</w>
  <w dep="" ne="" pos="" sym="">Turing</w>
  <pun>.</pun>
  <w dep="" ne="" pos="" sym="">1912</w>
  <pun>-</pun>
  <w dep="" ne="" pos="" sym="">1954</w>
  <pun>.</pun>
  <html tag="i">
    <w dep="" ne="" pos="" sym="">Biographical</w>
    <w dep="" ne="" pos="" sym="">Memoirs</w>
    <w dep="" ne="" pos=""
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
    <w dep="" ne="" pos="" sym="">Fellows</w>
    <w dep="" ne="" pos=""
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
    <w dep="" ne="" pos="" sym="bliss/the.png">the</w>
    <w dep="" ne="" pos="" sym="bliss/royal.png">Royal</w>
    <w dep="" ne="" pos="" sym="bliss/society.png">Society</w>
  </html>

```

```
<html tag="b">
  <w dep="" ne="" pos="" sym="">1</w>
</html>
<pun>:</pun>
<w dep="" ne="" pos="" sym="">253</w>
<pun>.</pun>
<link target="http://rsbm.royalsocietypublishing.org/content/1/253"/>
</s>
</ref>
<ref>
<s>
  <w dep="" ne="" pos="" sym="">Harnad</w>
<pun>,</pun>
  <w dep="" ne="" pos="" sym="">Stevan</w>
  <w dep="" ne="" pos="" sym="">2008</w>
<pun>.</pun>
<link target="http://eprints.ecs.soton.ac.uk/7741">
  <w dep="" ne="" pos="" sym="bliss/the.png">The</w>
  <w dep="" ne="" pos="" sym="">Annotation</w>
  <w dep="" ne="" pos="" sym="bliss/game.png">game</w>
<pun>:</pun>
  <w dep="" ne="" pos="" sym="">Turing</w>
<pun>( </pun>
  <w dep="" ne="" pos="" sym="">1950</w>
<pun>)</pun>
  <w dep="" ne="" pos=""
  sym="bliss/about,concerning,in_relation_to,of,on.png">on</w>
  <w dep="" ne="" pos="" sym="">Computing</w>
<pun>,</pun>
  <w dep="" ne="" pos="" sym="">machinery</w>
  <w dep="" ne="" pos="" sym="bliss/and,also,plus,too.png">and</w>
  <w dep="" ne="" pos="" sym="">intelligence</w>
</link>
<pun>.</pun>
  <w dep="" ne="" pos="" sym="bliss/in,inside,interior,internal.png">In</w>
<pun>:</pun>
  <w dep="" ne="" pos="" sym="">Epstein</w>
<pun>,</pun>
  <w dep="" ne="" pos="" sym="">Robert</w>
<pun>&</pun>
  <w dep="" ne="" pos="" sym="">Peters</w>
<pun>,</pun>
  <w dep="" ne="" pos="" sym="bliss/grace.png">Grace</w>
<pun>( </pun>
  <w dep="" ne="" pos="" sym="">eds</w>
<pun>)</pun>
<html tag="i">
  <w dep="" ne="" pos="" sym="">Parsing</w>
  <w dep="" ne="" pos="" sym="bliss/the.png">the</w>
  <w dep="" ne="" pos="" sym="">Turing</w>
  <w dep="" ne="" pos="" sym="bliss/test,assessment,exam.png">Test</w>
<pun>:</pun>
  <w dep="" ne="" pos="" sym="">philosophical</w>
  <w dep="" ne="" pos="" sym="bliss/and,also,plus,too.png">and</w>
  <w dep="" ne="" pos="" sym="">methodological</w>
  <w dep="" ne="" pos="" sym="">issues</w>
  <w dep="" ne="" pos="" sym="bliss/in,inside,interior,internal.png">in</w>
  <w dep="" ne="" pos="" sym="bliss/the.png">the</w>
  <w dep="" ne="" pos="" sym="">quest</w>
  <w dep="" ne="" pos="" sym="bliss/for_(in_exchange_for).png">for</w>
  <w dep="" ne="" pos="" sym="bliss/the.png">the</w>
  <w dep="" ne="" pos="" sym="bliss/think,reason-(to).png">thinking</w>
  <w dep="" ne="" pos="" sym="bliss/computer.png">computer</w>
</html>
<pun>.</pun>
```

```
<w dep="" ne="" pos="" sym="">Springer</w>
</s>
</ref>
<ref>
<s>
<w dep="" ne="" pos="" sym="">Copeland</w>
<pun>,</pun>
<w dep="" ne="" pos="" sym="">B</w>
<pun>.</pun>
<w dep="" ne="" pos="" sym="">Jack</w>
<w dep="" ne="" pos="" sym="">2006</w>
<pun>.</pun>
<html tag="i">
<w dep="" ne="" pos="" sym="">Colossus</w>
<pun>:</pun>
<w dep="" ne="" pos="" sym="bliss/the.png">The</w>
<w dep="" ne="" pos="" sym="">secrets</w>
<w dep="" ne="" pos=""
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
<w dep="" ne="" pos="" sym="">Bletchley</w>
<w dep="" ne="" pos="" sym="bliss/park.png">Park</w>
<w dep="" ne="" pos="" sym="bliss/be,am,are,is,exist-(to).png">'s</w>
<w dep="" ne="" pos="" sym="">code-breaking</w>
<w dep="" ne="" pos="" sym="bliss/computer.png">computers</w>
</html>
<pun>.</pun>
<w dep="" ne="" pos="" sym="">Oxford</w>
<w dep="" ne="" pos="" sym="bliss/university.png">University</w>
<w dep="" ne="" pos="" sym="bliss/bump,press,pressing.png">Press</w>
<pun>.</pun>
<w dep="" ne="" pos="" sym="">p108</w>
<w dep="" ne="" pos="" sym="">ISBN</w>
<w dep="" ne="" pos="" sym="">978</w>
<pun>-</pun>
<w dep="" ne="" pos="" sym="">0-19-284055-4</w>
</s>
</ref>
<ref>
<s>
<w dep="" ne="" pos="" sym="">Hodges</w>
<pun>,</pun>
<w dep="" ne="" pos="" sym="">Andrew</w>
<w dep="" ne="" pos="" sym="">1983</w>
<pun>.</pun>
<html tag="i">
<w dep="" ne="" pos="" sym="">Alan</w>
<w dep="" ne="" pos="" sym="">Turing</w>
<pun>:</pun>
<w dep="" ne="" pos="" sym="bliss/the.png">the</w>
<w dep="" ne="" pos="" sym="">enigma</w>
</html>
<pun>.</pun>
<w dep="" ne="" pos="" sym="">London</w>
<pun>:</pun>
<w dep="" ne="" pos="" sym="">Burnett</w>
<w dep="" ne="" pos="" sym="bliss/book.png">Books</w>
<pun>,</pun>
<w dep="" ne="" pos="" sym="">p488</w>
<pun>.</pun>
<w dep="" ne="" pos="" sym="">ISBN</w>
<w dep="" ne="" pos="" sym="">0-04-510060-8</w>
</s>
</ref>
<ref>
<s>
```

```
<w dep="" ne="" pos="" sym="bliss/the.png">The</w>
<w dep="" ne="" pos="" sym="">petition</w>
<w dep="" ne="" pos="" sym="bliss/be,am,are,is,exist-(to).png">was</w>
<w dep="" ne="" pos="" sym="bliss/alone,just,only,solitary.png">only</w>
<w dep="" ne="" pos="" sym="bliss/open-(to).png">open</w>
<w dep="" ne="" pos="" sym="bliss/until,till,to.png">to</w>
<w dep="" ne="" pos="" sym="">UK</w>
<w dep="" ne="" pos="" sym="bliss/citizen.png">citizens</w>
<pun>.</pun>
</s>
</ref>
</p>
<p>
<head size="2">Other websites</head>
<li type="dot">
<s>
<w dep="" ne="PERSON" pos="NNP" sym="">Jack</w>
<w dep="root" ne="PERSON" pos="NNP" sym="">Copeland</w>
<w dep="" ne="DATE" pos="CD" sym="">2012</w>
<pun>.</pun>
</s>
<s>
<w dep="" ne="PERSON" pos="NNP" sym="">Alan</w>
<w dep="root" ne="PERSON" pos="NNP" sym="">Turing</w>
<pun>:</pun>
<w dep="" ne="O" pos="DT" sym="bliss/the.png">The</w>
<w dep="" ne="O" pos="NN" sym="">codebreaker</w>
<w dep="" ne="O" pos="WP" sym="bliss/who,whom,that-(relative).png">who</w>
<w dep="" ne="O" pos="VBD" sym="bliss/save_(computer)-(to).png">saved</w>
<pun>'</pun>
<w dep="" ne="O" pos="NNS" sym="">millions</w>
<w dep="" ne="O" pos="IN"
sym="bliss/about,concerning,in_relation_to,of,on.png">of</w>
<w dep="" ne="O" pos="NNS" sym="bliss/life.png">lives</w>
<pun>'</pun>
<pun>.</pun>
</s>
<s>
<w dep="" ne="ORGANIZATION" pos="NNP" sym="">BBC</w>
<w dep="" ne="ORGANIZATION" pos="NNP" sym="bliss/news.png">News</w>
<pun>/</pun>
<w dep="root" ne="O" pos="NNP" sym="bliss/technology.png">Technology</w>
</s>
<s>
<link target="http://www.bbc.co.uk/news/technology-18419691"/>
</s>
</li>
</p>
</page>
```

A.2 Alan Turing: Full Visualization example

Alan Turing

Alan Mathison Turing OBE **FRS** (London, 23 June 1912 –
 Wilmslow, Cheshire, 7 June 1954) was an English

mathematician and computer scientist, born in Maida Vale, London.

He was **one** of the people who worked on the first

computers.

He was the first **person** to think of using a computer

to do things that were too hard for a person to do.

He **created** the Turing **machine** in 1936.

The machine was imaginary, but it included the idea of

a program.

Turing was interested in artificial intelligence.

He proposed the Turing test, to say when a

machine could be called "intelligent".

A computer could be said to "think" if a human

interrogator could not tell it apart, through conversation, from

a human being.

During World War II, Turing worked to break German

ciphers (secret messages).

Using cryptanalysis he helped to break the Enigma machine.

After that, he solved other German signals.

From 1945 to 1947 Turing worked on the design of the

ACE (Automatic Computing Engine) at the National Physical Laboratory.

He presented a paper on 19 February 1946, which was

the first detailed design of a stored-program computer.

Although ACE was a feasible design, the **secretcy** surrounding the wartime work at Bletchley Park **led** to delays in starting the project and he became disillusioned .

In late 1947 **he returned** to Cambridge for a

sabbatical year .

While he was at Cambridge , the Pilot ACE was **built**

in his absence .

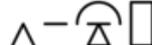
It executed its first **program** on 10 May 1950 .

Private life

 was a   **man**.

 In 1952, Turing admitted having had sex with a man

 in England .

> .   ,   was a  to take part

 in homosexual acts .

 He was **convicted** and forced to make a choice .

 He could **choose** between going to jail or take

hormones  like estrogen to lower his sex drive .

  to  the  .

After  punishment ,   impotent and 

not  an  .

 also   .

Death

 1954 ,  suffering  two  , 

 after  an  which  poisoned 

cyanide ,  drinking  cyanide  water .

The  for  confusion  that  police 

tested the apple for cyanide .

The treatment forced on him is now **believed** to be

very wrong , going against medical ethics and international laws

of human rights .

In August 2009 , a **petition** **apologised** to Turing for

punishing him for being a homosexual .

The **petition** **received** **thousands** of signatures .

Prime Minister Gordon **Brown** **acknowledged** the petition ,

+ and called Turing's treatment "appalling":

References

- Newman M. H. A. 1955. Alan Mathison Turing. 1912 – 1954. *Biographical Memoirs of Fellows of the Royal Society* 1: 253 [0]
- Harnad, Stevan 2008. The Annotation game: Turing (1950) on Computing, machinery and intelligence. In: Epstein, Robert & Peters, Grace (eds) *Parsing the Turing Test: philosophical and methodological issues*

       Springer
in the quest for the thinking computer .

3. Copeland, B. Jack 2006 . *Colossus : The secrets of*

    Oxford
Bletchley Park 's code-breaking computers .

   p108 ISBN 978 - 0-19-284055-4

4. Hodges, Andrew 1983 . *Alan Turing : the enigma .* London :

  p488 . ISBN 0-04-510060-8

      
5. The petition was only open to UK citizens .

Other websites

-
- Jack **Copeland** 2012 .

Alan **Turing** : The codebreaker who saved ' millions of



lives ' .



BBC

News /



Technology