Automatic thesaurus extraction for Icelandic

Frank Arthur Blöndahl Cassata
Reykjavík University
Ofanleiti 2, IS-203 Reykjavík Iceland
frank04@ru.is

Abstract

Thesauri are becoming a common resource used in various Natural Language Processing and Information Retrieval related tasks. Methods for automatic extraction of thesauri have just recently begun performing well enough for practical use. A method to automatically extract a thesaurus for Icelandic from a tagged and parsed corpus was implemented and evaluated. The method is based on extracting relational trigrams from the corpus and performing statistical analysis on them. As resources in the field of NLP are very limited for Icelandic, the effect of a small corpus on the final outcome is also examined. The results indicate that this method works well for Icelandic, but is highly dependent on the quality of its input. A by-product of this work is an evaluation of the *CST lemmatiser*.

1 Introduction

Natural Language Processing (NLP) is an interdisciplinary field, dealing with computer science (especially artificial intelligence), linguistics and statistics. The primary aim of most NLP related tasks is to build software that has the ability to use, and in some cases learn, a well-defined subset of human language, such as syntax or grammar, in order to help people in their day-to-day work and play.

Thesauri are a valuable resource, both for those who want to avoid awkward repetitions in their writings, and for expanding search results in domain specific Information Retrieval (IR) systems. Imagine for instance a doctor writing a case report about a patient with a specific type of cancer. He might refer to the cancer as a neoplasm, a tumor, or a malignancy. Another doctor doing research on the subject would want the same results regardless of which of these words he uses as input. Ever since the handmade Roget's thesaurus was put on a computer readable format in 1957 (Masterman, 1957), computer scientists have been looking for new ways to construct thesauri automatically. Many uses were quickly found for thesauri, primarily in the field of machine translation, a very popular subject of research during the Cold War.

When it comes to solving NLP tasks, working with Icelandic can be difficult, in part because of complex grammar. The current state is such that there is much to be done to catch up with the progress that has been made for English. The biggest limiting factor is the fact there are relatively few native speakers of the language. This makes it unprofitable to develop software to sell commercially, as the development cost would in all likelihood far exceed even the most optimistic income expectations. The situation is completely opposite when it comes to English. With its many speakers, both native and foreign, software to correct grammar, for example, can be very profitable. The resources used in NLP tasks for English are based on many decades of research, dating as far back as the 1950's. Today both researchers and software developers have various resources at their disposal. A few examples include word nets, large tagged and/or parsed corpora, general and domain-specific thesauri, etc.

In 1996, the Icelandic Ministry of Education, Science, and Culture published the booklet "By the power of information"[1] (Ministry of Education, 1996), in which they expressed their new-found interest in building a foundation of knowledge in this field. Two years later a committee was set up to investigate ways to build up the resources needed. As a result, funding was given to various research projects (Ólafsson et al., 1999). Even though a number of research projects have been carried out since then, much more is needed. An example is that the largest Icelandic hand-checked tagged corpus contains only about 600,000 tokens, while the size of many English corpora is in the millions or dozens of millions. Another example is that an Icelandic thesaurus has been constructed in computer-readable form, but it was done in such a way that using it as an NLP resource is almost impossible. It would be very labor intensive to change the form in which the thesaurus is stored. Therefore automatic ways to construct thesauri are needed (Ólafsson et al., 1999).

Despite the undisputed value of thesauri as an NLP resource, no research has been carried out to create a thesaurus automatically for Icelandic, to the author's best knowledge. A method will be implemented that has been used with success for English. The method is based on the theory that similar words often appear in similar context. In this work the focus is on noun synonimity. The method is described in detail in section 3.

The remainder of this paper is organized as follows: In section 2, previous work that has been done to automatically extract thesauri is discussed. The theory behind the method of choice is then described in section 3. The implementation is discussed in section 4. The results of this work are then detailed in section 5. Finally, in section 6, concluding remarks are given, and possible future work that can be done to improve upon this work is discussed.

2 Related work

A lot of work has been put into the research of automatic thesaurus creation for English. Three of these methods are discussed in the following sections, ending with the method of choice.

[1] Icelandic title is "Í krafti upplýsinga".

2.1 Dictionary lookup

Bilingual dictionaries can be used to extract thesauri, and the method is quite simple (Scannell, 2003). This method has been used with some success, especially when dealing with minority languages where NLP resources are limited, as is the case with Icelandic. To implement the method for Icelandic, an English thesaurus and an Icelandic – English dictionary, both in computer-readable form, could for example be used. Extracting the thesaurus would then only require an automatic translation of the entries in the English thesaurus with the dictionary. It soon becomes clear that this method has a few shortcomings that need to be addressed, such as choosing headwords. The meaning of a word may also be regional. For instance, the word *pint* would be understood as *beer* by most native English speakers, while it would translate to *hálfpottur* (lit. *half-pot*) in Icelandic, a word that many Icelanders associate with milk or cream. Lastly, the input resources pose limitations, for instance in the definition of similarity for two words. One might want to define the similarity between two words differently (for example more liberally).

2.2 Wordnets

Wordnets are not proper thesauri since they contain more information about the relationship between words, such as hypo- and hypernyms. They can be visualized as trees with words for nodes, where the words near the root contain abstract concepts and things. As one moves down the tree, the meaning of words becomes narrower and more specific. Every node is a hypernym of all words below it, and one might conclude that all words belonging to the same parent are synonyms. The method of constructing such a tree is often based on finding patterns in text (Curran and Moens, 2002; Hearst, 1992). The patterns must be found manually, but finding good patterns is not difficult. Consider the following pattern:

$$NP_1, NP_2, ..., NP_n \textit{ and other } NP_k{}^2$$

For every (NP_i, NP_k) pair, where $0 < i \leq n$, it can be said that NP_i is a hyponym of NP_k. The sentence *Iceland, Norway, Denmark and other countries [...]* would fit this pattern. To show that this method is not particularly good for finding synonyms, consider the sentence *Keys, carburetors and other metal objects [...]*. It is clear that *key* and *carburetor* are both metal objects, but in no way would they be considered synonyms.

This method relies the least on other NLP resources, as the only thing needed is a corpus. Since the patterns used only occur in a small fraction of the sentences in the input, an extremely large input corpus would be needed.

2.3 Context of words

This method is based on the theory that related words often appear in similar context (Caraballo, 1999; Hindle, 1990; Lin, 1997; Uramoto, 1996) and is the method chosen for the present work. The

[2] NP stands for "Noun Phrase".

method sounds simple; the context in which a word appears dictates its meaning. If a computer could understand or analyze the context in which a word stands, and group together words that appear in similar context, it might be assumed that a good method of finding synonyms has been found. The downside of this method is the fact that it relies heavily on other NLP resources, but all the resources needed exist for Icelandic. This method is described in detail in the next section.

3 Topic description

The theory behind the method of choice is very simple, and becomes clear by looking at a simple example presented by Lin (1998) (slightly modified):

1. There is a can of Achel on the table.
2. Everyone likes Achel.
3. You become drunk if you drink Achel.
4. Achel is made from hop and barley.

The context in which the word *Achel* appears indicates that it is an alcoholic beverage, probably some kind of beer. If a computer could analyze the context that the word appears in, it might be assumed that a good method of finding synonyms has been found.

When this method is used, the primary input resource is a large corpus that has been tagged and parsed. To the author's knowledge, the smallest corpus that has been used with this method consists of 64 million words, and is constructed from texts from *AP Newswire* (19 million words), *Wall Street Journal* (24 million words), and *San Jose Mercury* (21 million words) (Lin, 1998). From the corpus, relational trigrams are extracted. They describe how words in the sentence relate to each other syntactically, with respect to verb subjects and objects, as well as noun modifiers. The trigrams are of the form *(w, r, w')*, where *w* and *w'* are two words, and *r* represents their relation. In the sentence *I have a brown dog* the following trigrams could be found:

- (*I, subject-of, have*)
- (*brown, modifier-of, dog*)

When these trigrams have been extracted, the frequency of each trigram is defined with ||*w, r, w'* ||. For wildcards, *** can be used instead of *w*, *r* or *w'* to count the frequency of the trigrams matching the resulting pattern. ||*brown, modifier-of, **|| would then give the total number of trigrams where *w* is *brown*, and *r* is *modifier-of.* This notation has nothing to do with the actual implementation; it is only used to facilitate the presentation of this theory.

To calculate the similarity between individual words, statistical analysis can then be performed on the set of trigrams. The methods for calculating similarity are described in the next section, where the implementation is discussed.

4 Implementation

In section 4.1 the NLP resources used for this work are presented. Then in section 4.2 the similarity measures used to process the dependency trigrams are discussed. Finally in section 4.3 an overview of the implementation is given.

4.1 Resources

The input text must both be tagged with part-of-speech (POS) tags, and parsed to identify language constituents. This is done with *IceTagger* (Loftsson, 2006, 2007), a part-of-speech tagger, and *IceParser* (Loftsson and Rögnvaldsson, 2007), a shallow finite state parser. Furthermore, words need to be reduced to their base forms, called lemmas. For example, it must be clear that *hestur* (the nominative form of *horse*) and *hest* (the accusative) represent the same word. To achieve this, the *CST lemmatiser* is used (Jongejan and Haltrup, 2005), a tool that can be trained to create rules to reduce words to their base form (lemma). For Icelandic the lemmas are the nominative singular form for nouns, and the infinitive form for verbs.

For the input text the largest tagged Icelandic corpus *Íslensk Orðtíðnibók* (OTB) was used (Pind et al., 1991). The OTB consists of approximately 600,000 tokens, and contains a POS tag for each word, as well as the word's lemma, and has been hand-checked for errors. Therefore it is not necessary to POS tag this corpus, or use the lemmatiser on it, but it still needs to be parsed with *IceParser*. As stated before the smallest corpus previously used with this method (to the author's knowledge), contained over 60 million words. Since the method relies on analyzing the context in which words appear, it follows that as more text is used for input, the more information can be extracted for each word. Therefore, the OTB corpus can probably not be used to create a usable thesaurus. The first phase of this work involves making preliminary evaluations of the method chosen when used with the OTB corpus. In the second phase, a larger corpus will be created with texts from *Morgunblaðið* (MBL), an Icelandic national newspaper. This corpus will be tagged and parsed with *IceTagger* and *IceParser*, respectively, and finally lemmatized with the *CST lemmatiser*.

4.2 Calculating similarity

To calculate the similarity between two words, a method described by (Lin, 1998) is used. To describe a word w, all trigrams matching the pattern (w, *, *) are found. From this, the similarity of two words can be calculated by looking at the trigrams that describe both of the words in the same way. These calculations can be performed in the following manner. First the amount of information contained in each trigrams is calculated. The formula for this is derived in a few steps. First observe that an occurrence of a relational trigram (w, r, w') can be looked at as the co-occurrence of three events:

A: a randomly selected word is *w*.

B: a randomly selected dependency type is *r*.

C: a randomly selected word is *w'*.

The probability of A, B and C co-occurring is estimated by:

$$P_{MLE}(B)P_{MLE}(A\,|B)P_{MLE}(C\,|B) \tag{1}$$

where P_{MLE} is the maximum likelihood estimation of a probability distribution and:

$$P_{MLE}(B) = \frac{\|*, r, *\|}{\|*, *, *\|} \tag{2}$$

$$P_{MLE}(A\,|B) = \frac{\|w, r, *\|}{\|*, r, *\|} \tag{3}$$

$$P_{MLE}(B\,|C) = \frac{\|*, r, w'\|}{\|*, r, *\|} \tag{4}$$

If the value of $||w, r, w'||$ is known, $P_{MLE}(A, B, C)$ can be computed directly thus:

$$P_{MLE}(A, B, C) = \frac{\|w, r, w'\|}{\|*, *, *\|} \tag{5}$$

The information in each trigram is now defined as $I(w, r, w')$ with the following formula:

$$I(w, r, w') = -\log(P_{MLE}(B)P_{MLE}(A\,|B\,)P_{MLE}(C\,|B\,)) - (-\log P_{MLE}(A, B, C)) \tag{6}$$

Or more simply as:

$$I(w, r, w') = \log \frac{\|w, r, w'\| \times \|*, r, *\|}{\|*, *, *\| \times \|*, r, w'\|} \tag{7}$$

If $T(w)$ is defined as the pairs of (r, w') where $I(w, r, w')$ is a positive number, the similarity between two words, w_1 and w_2, can be measured in a number of ways. The following formula shows how the cosine similarity between the two words can be measured:

$$sim(w_1, w_2) = \frac{|T(w_1) \cap T(w_2)|}{\sqrt{|T(w_1)| \times |T(w_2)|}} \tag{8}$$

Formula 9 shows the similarity measure used in this work, proposed by Lin (1998).

$$sim(w_1, w_2) = \frac{\sum_{(r,w) \in T(w_1) \cap T(w_2)} I(w_1, r, w) + I(w_2, r, w))}{\sum_{(r,w) \in T(w_1)} I(w_1, r, w) + \sum_{(r,w) \in T(w_2)} I(w_2, r, w)} \tag{9}$$

What $sim(w_1, w_2)$ gives is a similarity measure of the two words as number between 0 and 1, where 1 would mean that the two words share the exact same (r, w') set, and 0 would then mean that they share no common (r, w') pairs.

These formulas are from (Lin, 1998). Many more formulas to calculate the similarity between words using the trigrams can be found in (Curran and Moens, 2002; Dagan et al., 1994). Most similarity measures can also be adapted to this method.

Using formula 7, combined with either formula 8 or 9 (or other adapted measures of similarity), a thesaurus can now be extracted. For each word, a thesaurus entry might be created containing the first n words that are most similar to it. Then a cut-off point must be chosen for $sim(w_1, w_2)$ somewhere between 0 and 1 (depending on how loosely synonyms are defined). Lastly a way to extract appropriate headwords for the thesaurus must be found (that is however not within the scope of this work). This method was described by Lin (1998).

4.3 Implementation

Two separate Java programs were developed to extract a thesaurus from a corpus. First the *TrigramExtractor*, that parses the corpus and extracts relational trigrams from it. Figure 1 shows the flow of the program.

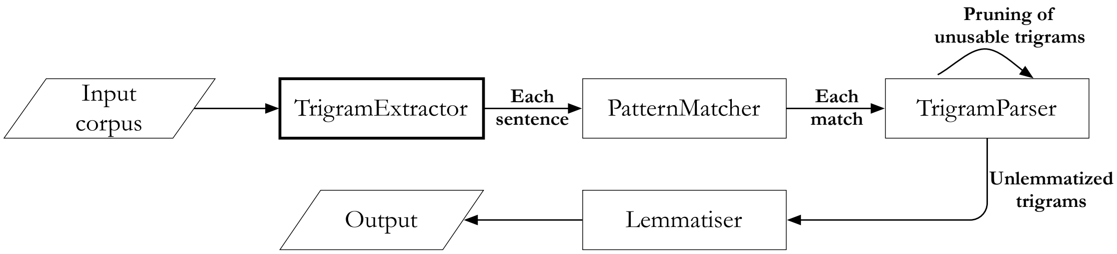

Figure 1: TrigramExtractor flow

The main class, *TrigramExtractor*, parses the input corpus and passes each sentence to a pattern matcher. The pattern matcher finds predefined patterns, defined with regular expressions, known to contain specific trigram candidates. Figure 2 shows how a parsed sentence might look, a pattern matching it, as well as the resulting trigram. Each of these matches is then sent to the trigram parser, which extracts the actual trigrams from the matched pattern. Unusable trigrams are also pruned from the final output. They include trigrams containing proper names and pronouns, as well as a set of "meaningless" verbs. These are verbs that do not contribute to a better understanding of the words context. For example, almost all nouns can *be* something, and most nouns can be *had*, so *have* and *be* are two of the "meaningless" verbs filtered out. The trigram parser contains a list of such verbs, and discards all trigrams containing them, as well as trigrams containing pronouns and proper nouns. Every trigram is then passed to a lemmatiser, and finally written to disk. The program is capable of extracting roughly 5000 trigrams per second[3], but this can probably be improved.

```
{*SUBJ> [NP blístrið nheng NP] *SUBJ>} [VP hljómaði sfg3eþVP]
{*SUBJ .*? \[NP (.*?) NP\] .*? *SUBJ>} .*? \[VP[a-z]+ (.*?)VP[a-z]+\]
(blístrið_nheng, subject-of, hljómaði_sfg3eþ) → lemmatiser →(blístur, subject-of, hljóma)
```

Figure 2: A parsed sentence, a pattern matching it, and the resulting trigram

The second part of the implementation is the trigram similarity calculator, called *TriSim*. *TriSim* reads the output of *TrigramExtractor* and parses it. Then for each word it calculates its similarity with every word found in the trigram set, using formula 8. Then the first n similar words for each word are written to disk, for this work $n = 15$ is used. The performance of *TriSim* is highly dependent on both the number of trigrams and the number of unique nouns in the input.

5 Results

In section 5.1 the results when using the small OTB corpus are presented. The results when using the two larger MBL corpora are then given in sections 5.2 and 5.3. Section 5.4 gives an evaluation of the thesauri extracted from the MBL corpora. Finally in section 5.5 an evaluation of the *CST lemmatiser* is presented.

5.1 Extracting a thesaurus from OTB

The OTB corpus is not well-suited for this method for two main reasons. First, it is very small, only around 600,000 tokens, or around 0.9% of the corpus used by Lin (1998). Second, the OTB corpus is made up of texts from Icelandic novels, so it contains a lot of 1^{st} or 3^{rd} person narratives. This yields a low ratio of usable trigrams because of proper nouns and pronouns, as well as

[3] Assuming a 2GHz CPU and 2GB RAM.

the "meaningless" verbs mentioned before. It must also be mentioned that the trigram extraction process relies on the correct syntactic parsing of the corpora. An evaluation presented in (Loftsson and Rögnvaldsson, 2007) shows that the accuracy of *IceParser* for phrases and syntactic functions is 96.7% and 84.3%, respectively, when parsing a correctly tagged corpus. Since *TrigramExtractor* works by finding patterns of phrases and syntactic functions, this means not all trigrams in the text will be found. Furthermore, incorrect trigrams will be present in the trigram set. In total roughly 70,000 trigrams were extracted from OTB. Of those more than 44,000 were discarded due to proper nouns, pronouns and "meaningless" verbs. This can be seen clearly in figure 3.

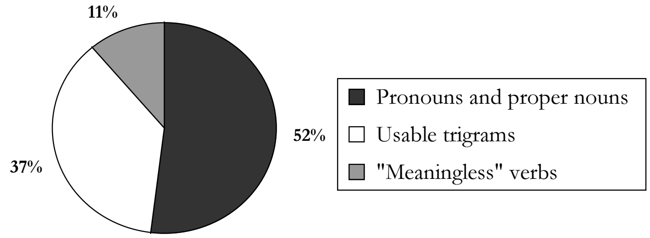

Figure 3: Usable trigram ratio in OTB

The noun frequency of the OTB corpus is also much skewed towards lower frequencies. Of the 6951 unique nouns found in the corpus, 6523 of them appear less than 10 times (over 3000 only once), yielding little information about their context. The frequency distribution can be seen in figure 4 (note that the y-axis is logarithmic).

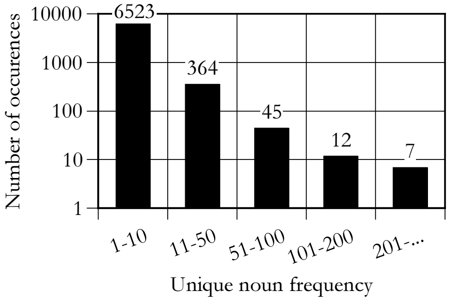

Figure 4: Noun frequency distribution in OTB

Although the OTB corpus is not ideal for this method it does give some promising results. The highly skewed noun frequency is the factor that affects these results the most. With a method that is based on analysing the context in which nouns appear it is clear that each noun must occur at least a few times for it to be possible to make any assumptions about it. A few test runs were done where nouns appearing less than 50 times were cut from the input. Below are three sample entries from the output, that should show that the method is performing as it should. One thing to note is that the first entry does not contain synonyms *per se*, but is still an indication that the method works. A literal English translation of each word has been inserted in parentheses behind each word:

blístur (tootle):	skarkali (racket): 0.591, bjölluhljómur (ding): 0.576, [...]
skúr (shed):	bygging (building) 0.951, hús (house) 0.632, [...]
birta (brightness):	bygging (building) 0.310, ljós (light) 0.310, [...]

This corpus is clearly not suited to be used as input when extracting a thesaurus, but is a clear indication that the method works.

5.2 Extracting a thesaurus from MBL

Two tagged and parsed corpora were created from texts from the newspaper *Morgunblaðið*. One 4 million word corpus, and one 15 million word corpus (MBL4 and MBL15, respectively). The main difference between the MBL corpora and the OTB corpus, other than size, is that while the OTB corpus has been hand-checked for errors, some errors are introduced into the MBL corpora when it is tagged with *IceTagger*. According to an evaluation presented in (Loftsson, 2007), the accuracy of *IceTagger* is 91.5%. The accuracy of *IceParser* for phrases and syntactic functions also drops to 91.9% and 75.3%, respectively, when parsing the output of *IceTagger* (Loftsson and Rögnvaldsson, 2007). Another factor that introduces even more errors into the final results is the *CST lemmatiser*. Even with these errors in the input it was quite surprising that the results from the MBL4 corpus were not much better than the OTB corpus results. Some improvement could however be seen when using the MBL15 corpus.

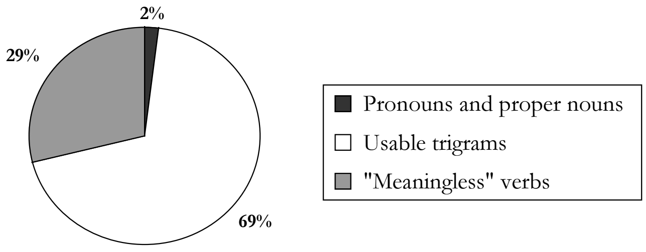

Figure 5: Usable trigram ratio in MBL4 and MBL15

label First item

label Second item

A much higher percentage of usable trigrams was obtained from the MBL corpora, as can be seen in figure 5. Of about 280,000 trigrams extracted from the MBL4 corpus (852,000 in MBL15), 69%, or over 200,000 (594,000 in MBL15), were usable trigrams, as opposed to only 37% in the OTB corpus. Most noticeable is the drop in trigrams containing pronouns and proper nouns, since newspaper do not generally contain a lot of 1^{st} and 3^{rd} person narratives. However, it was surprising to see the percentage of trigrams discarded due to "meaningless" verbs grow almost 3-fold. Explanation for this increase was not investigated.

The errors introduced in tagging and parsing the text should not affect the results much. The reason is that this method is not overly sensitive to errors in the input. If a word has 50 trigrams and 2 or 3 of them are wrong, it should not change the results substantially since these few wrong trigrams do not weigh much compared to the many correct ones in the calculations. However, it is apparent that the lemmatiser is a big factor in the quality of the results. Every word incorrectly lemmatised is added into the set of unique words, and this results in unusually many words appearing very infrequently. This can be seen in the noun frequency distribution in MBL4 in figure 6.

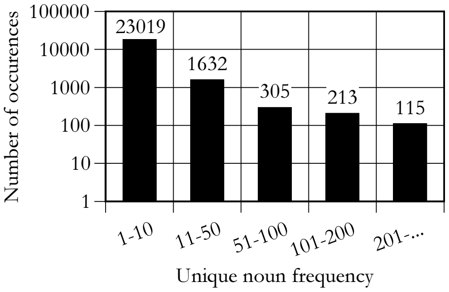

Figure 6: Noun frequency distribution in MBL4

Of the 25,284 unique nouns in the MBL4 corpus, over 23,019 appear less than 10 times, and over 12,742 appear only once. When words with lower frequencies are cut out, as was done with the OTB corpus, the results look promising. However, as a result there are too few words left to get a usable thesaurus.

As can be seen in figure 7 there are 60,281 unique nouns in the MBL15 corpus, of which 55,178 appear less than 10 times, and 32,802 appear only once. In MBL15, as opposed to the OTB and MBL4 corpora, words with a frequency over 200 exceed the number of words appearing 101-200 times. Again the words with the lowest frequencies were cut out, since they skew the results highly.

Below are four entries from the MBL4 corpus, both good and bad ones, with literal English translations in parentheses:

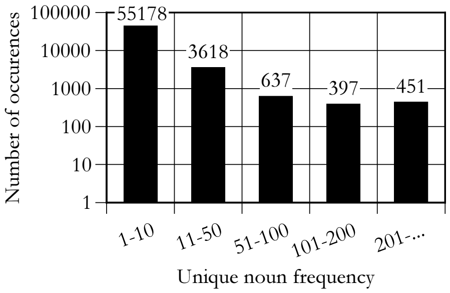

Figure 7: Noun frequency distribution in MBL15

háskólagráða (university degree):	doktorsnám (doctoral studies) = 0.699,
	stúdentspróf (GCSE) = 0.666, [...]
körfubolti (basketball):	landsleikur (national game) = 0.694,
	fótbolti (football) = 0.656, [...]
keppni (competition):	útför (funeral) = 0.521,
	ganga (hike) = 0.509, [...]
taug (nerve/rope):	vitni (witness) = 0.607,
	búskapur (farming) = 0.477, [...]

Even though the MBL15 results were somewhat better than the MBL4 and OTB results, they are still not good enough for this method. Results from both of the MBL corpora are however a good indication that a larger corpus does give better results. A corpus even larger than MBL15 should give much stronger results. Below are four entries from the MBL15 thesaurus:

vist (card game):	briddsdeild (bridge division) = 0.618,
	bridds (bridge) = 0.607, [...]
atvinna (profession):	sjómennska (seamanship) = 0.518,
	framhaldsnám (graduate studies) = 0.394, [...]
athugasemd (comment):	fyrirspurn (enquiry) = 0.532,
	tölvupóstur (e-mail) = 0.352, [...]
meðferð (treatment):	fótaaðgerð (foot-operation) = 0.368,
	meðhöndlun (treatment) = 0.218, [...]

5.3 Evaluation of the MBL4 and MBL15 results

Evaluating the quality of a thesaurus is difficult, since the definition of a synonym is highly subjective. As an attempt to evaluate the quality of the MBL4 and MBL15 thesauri 10 random entries were chosen from both thesauri. They were printed out, and 10 people were asked to evaluate them. They were given two different color markers and instructed to mark true synonyms with one color, and words in some way related to the headword with the other. The results of the evaluation are presented in tables 1 and 2.

Person	Synonyms	Related
1	2.67%	6%
2	2.67%	4.33%
3	1.4%	5.2%
4	0%	6%
5	1.8%	4.67%
6	2.67%	3.2%
7	2.33%	3.78%
8	1.67%	4.5%
9	1.32%	6.13%
10	0.77%	4.23%
avg	**1.73%**	**4.8%**

Table 1: MBL4 evaluation

Person	Synonyms	Related
1	11.33%	18%
2	10.6%	17.2%
3	11.2%	17.33%
4	9.67%	15.67%
5	7.56%	19.27%
6	8.55%	14.89%
7	7.71%	16.2%
8	10.27%	18.6%
9	11%	15.67%
10	8.46%	13.45%
avg	**9.64%**	**16.63%**

Table 2: MBL15 evaluation

While this evaluation method is quite rudimentary and highly susceptible to human bias, it seems to work quite well. It clearly shows the advantage of a bigger corpus, since the average ratio of synonyms grows more than 5-fold, and the ratio of related words grows by a factor of 3.5. From this it can be conjectured that the results would be much stronger with a larger corpus. It can however not be concluded from this data alone that the ratio of synonyms would continue to grow linearly with larger corpora, that conclusion can only be made with further testing.

5.4 Lemmatiser evaluation

The *CST lemmatiser* was trained on the 600,000 tokens from the OTB corpus. It was then evaluated by hand-checking 600 random words from its output. The results were quite good, with 90% correct lemmas. A specific pattern in its wrong output was not apparent. It did seem to perform slightly worse with words that had vowel mutations between the form being lemmatised and the lemma itself. Since *IceTagger* has an accuracy of 91.5%, most words with incorrect POS tags were also incorrectly lemmatised.

Although 90% sounds very good, it must be noted that it is probably too low for this work. Many of the people who evaluated the MBL corpora noted that some of the words in their lists were not correct Icelandic words, a result of incorrect lemmatisation. A part of the reason why words with lower frequencies were cut out when extracting thesauri from the MBL corpora was to avoid the incorrect lemmas (most of which occur infrequently), but some of them do appear too frequently to be avoided with such measures.

6 Conclusion and future work

Extracting a thesaurus automatically is both difficult and requires a number of different resources. The method chosen for implementation for extracting thesauri from Icelandic text works well, but is highly dependent on its input, especially the size of the corpus, as well as the accuracy of the tools used in the trigram extraction process. Two questions remain unanswered:

1. Will a much larger corpus yield better results?
2. Are other methods of lemmatisation better suited for this method?

The author hopes to answer these questions with future research.

Acknowledgements

The author would like to thank Hrafn Loftsson for his input in this work, as well as supplying the primary resources used in it (*IceTagger*, *IceParser*, the OTB corpus and the untagged and unparsed MBL corpus). Bart Jongejan also receives thanks for his help in using the *CST lemmatiser*, as well as Kristinn R. Þórisson for his comments and suggestions.

References

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 120–126, Morristown, NJ, USA. Association for Computational Linguistics.

Curran, J. R. and Moens, M. (2002). Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 59–66, Morristown, NJ, USA. Association for Computational Linguistics.

Dagan, I., Pereira, F., and Lee, L. (1994). Similarity-based estimation of word cooccurrence probabilities. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 272–278, Morristown, NJ, USA. Association for Computational Linguistics.

Hearst, M. A. (1992). Automatic Acquisition of Hyponyms. Technical report, Berkeley, CA, USA.

Hindle, D. (1990). Noun classification from predicate-argument structures. In *Proceedings of the 28th annual meeting on Association for Computational Linguistics*, pages 268–275, Morristown, NJ, USA. Association for Computational Linguistics.

Jongejan, B. and Haltrup, D. (2005). *The CST Lemmatiser*. http://www.cst.dk/download/cstlemma/current/doc/cstlemma.pdf.

Lin, D. (1997). Using syntactic dependency as local context to resolve word sense ambiguity. In *Proceedings of the 35th annual meeting on Association for Computational Linguistics*, pages 64–71, Morristown, NJ, USA. Association for Computational Linguistics.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics*, pages 768–774, Morristown, NJ, USA. Association for Computational Linguistics.

Loftsson, H. (2006). Tagging a morphologically complex language using heuristics. In *T. Salakoski, F. Ginter, S. Pyysalo and T. Pahikkala (eds.), Advances in Natural Language Processing, 5th International Conference on NLP, FinTAL 2006, Proceedings*, Turku, Finland.

Loftsson, H. (2007). Tagging Icelandic Text using a Linguistic and a Statistical Tagger. In *Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the ACL*, Rochester, NY, USA.

Loftsson, H. and Rögnvaldsson, E. (2007). IceParser: An Incremental Finite-State Parser for Icelandic. In *Proceedings of NoDaLiDa 2007*, Tartu, Estonia.

Masterman, M. (1957). *The thesaurus in syntax and semantics*, chapter 4, pages 1–2.

Ministry of Education (1996). Í krafti upplýsinga. Tillögur menntamálaráðuneytisins um menntun, menningu og upplýsingatækni 1996-1999.

Ólafsson, R., Rögnvaldsson, E., and Sigurðsson, Þ. (1999). Tungutækni - Skýrsla starfshóps. Menntamálaráðuneytið.

Pind, J., Magnússon, F., and Briem, S. (1991). Íslensk orðtíðnibók. Reykjavík: Orðabók háskólans.

Scannell, K. (2003). Automatic thesaurus generation for minority languages: an Irish example. *Atelier TALN'03, vol.2*, pages 203–212.

Uramoto, N. (1996). Positioning unknown words in a thesaurus by using information extracted from a corpus. In *Proceedings of the 16th conference on Computational linguistics*, pages 956–961, Morristown, NJ, USA. Association for Computational Linguistics.