A prototype of a grammar checker for Icelandic

Abstract

A grammar checker is one of the basic Natural Language Processing (NLP) tools for any language. The NLP field is relatively new in Iceland and a lot of tools have yet to be developed. One of these is a grammar checker. Methods used for grammar checking in other languages were examined and a prototype of a rule-based grammar checker for Icelandic developed. Resources in the field of NLP for Icelandic are limited. This, along with the complex grammar, presents some problems when developing a grammar checking system for the language. However, the results indicate that the approach chosen, to implement a rule-based system, is suitable.

Málfræðileiðréttingarforrit er eitt af grunntólum málvinnslu í hverju tungumáli. Málvinnslusviðið er tiltölulega nýtt hér á landi og því er mikið verk óunnið, meðal annars á eftir að þróa málfræðivilluleit og leiðréttingarforrit. Útfærsla slíkra leiðréttingarforrita fyrir önnur tungumál voru skoðaðar og frumgerð af íslenskri málfræðivilluleit þróuð. Þar sem íslensk málfræði er flókin og enn á eftir að útfæra margt af því sem nauðsynlegt er til að leiðrétta megi villur var áhersla lögð á villuleitina sjálfa og hvernig hentugast væri að inntak hennar væri. Útfærsluaðferðin sem valin var, byggð upp á reglum fyrir hverja gerð af mögulegum villum, virðist henta ágætlega fyrir íslenska villuleit.

1 Introduction

Natural language processing (NLP) is a subfield of computer science, with strong connections to artificial intelligence, linguistics and statistics among others. One area of NLP is concerned with creating proofing systems, such as spell checkers and grammar checkers. A grammar checker looks for grammatical errors and, in many cases, suggests possible corrections. Grammar checkers are also part of the fundamental tools needed for NLP in any language.

The best supported language in NLP, by far, is English. A lot of work has gone into developing sophisticated systems that have gone into widespread use, such as automatic translators and spell checkers. The progress made for the Nordic languages is nowhere near as much as for English. Icelandic, especially, is still in the early stages. In some ways this is understandable as the use of English is extremely widespread, with many native speakers as well as second language speakers.

Icelandic, on the other hand, has very few native speakers (around 300,000) and not a lot of people learn it as a second language. Therefore, the demand for NLP systems is little and so are the funds to develop them as they are usually not profitable.

In some ways, the problems that arise when dealing with Icelandic are very different from those faced when dealing with English. For example, the grammar is very complex, making it very hard to create grammar checkers and develop general understanding of the language. However, this perhaps makes a grammar checker an even more valuable part of the Icelandic NLP toolkit as it can help with language training at all levels, for both native and non-native speakers.

The Icelandic Ministry of Education, Science and Culture expressed an interest in supporting the NLP field in the booklet "Í Krafti Upplýsinga"[1] (Ministry of Education, 1996). A committee was formed in 1998 to look at what needed to be done in the field and decide upon the best ways to build up the necessary resources. In the committee's report "Tungutækni – skýrsla starfshóps"[2] (Ministry of Education, 1999) it was noted that in spite of the relatively high level of interest Icelanders have in their language, NLP was in its very early stages as a research field in Iceland. It was stated that a lot of work still had to be done and that the government's support was especially important regarding funding for the field due to the few number of speakers. It was maintained that what needed to be done next was, without any doubt, developing the necessary tools to correct written Icelandic and that work in that area should be done fast.

Since this report was published systems have been developed for Icelandic in several areas of NLP. As an example, the part-of-speech tagger and parser used in this project (*IceTagger* and *IceParser*, parts of the IceNLP toolbox for Icelandic (Loftsson and Rögnvaldsson, 2007b)) offer analysis of Icelandic text that is essential for further development of any kind of proofing system for the language. Other systems that research has focused on include spell checkers, speech synthesizers and machine translation systems.

Despite its potentially large role in the NLP toolkit, not a lot of research has been carried out to create an automatic grammar checker for Icelandic, to the author's best knowledge. The aim in this project is to implement a rule-based method for checking grammar in text.

This paper is organized as follows: Section 2 discusses the part-of-speech tagger and parser used for initial analysis of the input text as well as previous work that has been done in the area for

[1] "By the Power of Information"
[2] "Report on Language Technology in Iceland"

other languages. Section 3 describes the method chosen to implement the system, resources needed and the errors chosen as a focus for the system. Implementation of the system is described in section 4. Results of the project are discussed in section 5 and section 6 contains final conclusions, remarks and possible future work.

2 Related Work

A lot of work has gone into developing grammar checkers for languages, other than Icelandic. The most progress, by far, has been made for English. The earliest grammar checkers for English were developed in the 1970s and have gradually been improving over the last decades. Although there is still room for improvement their use is quite widespread as an English grammar checker is built into the most used word processor today, Microsoft Word. Some research has also been devoted to developing grammar checkers for the other Scandinavian languages. Swedish, in particular, has had several grammar checkers developed and a lot of research in the area, e.g. Granska (Carlberger, Domeij, Kann and Knutsson, 2000) and Grammatifix (Arppe, 2000) and a grammar checker for Norwegian has been developed and used in Microsoft Word (Hagen, Johannessen, Lane, 2001).

Before text is checked for grammatical errors, part-of-speech tagging and parsing need to be performed. The grammar checker used then starts by finding each sentence in the text, attempts to find any grammar errors in it and often suggests possible corrections. Most grammar checkers are rule-based, statistical or a hybrid of the two. Part-of-speech tagging and parsing as well as the aforementioned methods for grammar checking are described below.

2.1 Part-of-speech tagging and parsing

Before grammar checking can be performed on a text it needs to be run through a part-of-speech (POS) tagger and parser. This enables the grammar checker to recognise types of phrases within each sentence, syntactic roles and features of each word.

The text is first run through a POS tagger which generates a tag for each word in a sentence. The tag indicates the word’s class and morphological features (such as case, number, person and gender). Next, the text (with tags) is run through a parser which performs syntactic analysis on it, adding tags to parts of the sentence, marking phrases within it and syntactic roles.

An example of a sentence, before and after POS tagging and parsing:

- Original sentence:
  "Gamli maðurinn gekk í gegnum skóginn"
- After part-of-speech tagging:
  Gamli lkenvf maðurinn nkeng gekk sfg3eþ í aa gegnum ao skóginn nkeog . .
  Where the first letter in a tag indicates the word class of the preceding word and the other letters each indicate a morphological feature, such as case, number or person. E.g. the first word in this sentence (*Gamli*) is tagged as *lkenvf*. *l* indicates that it is an adjective, *k* that it is masculine, *e* that it is singular, *n* tells us it's in the nominative case, *v* indicates weak declension and *f* that the degree is positive.
- After parsing:
  *{*SUBJ> [NP [AP Gamli lkenvf AP] maðurinn nkeng NP] *SUBJ>}*
  [VP gekk sfg3eþ VP]
  [PP [MWE_PP í aa gegnum ao MWE_PP] [NP skóginn nkeog NP] PP]
  . .
  Where the words within each set of braces correspond to a syntactic function in the sentence (e.g. SUBJ indicates the subject of a sentence) and the words within each set of brackets corresponds to a sentence constituent (e.g. NP indicates the beginning and end of a noun phrase)

2.2 Statistical grammar checking

In the statistical approach the system is trained on a corpus to learn what is 'correct'. As an example, this can be done using trigram frequency information. A POS-tagged corpus is then used to construct a list of tag sequences (each sequence being the tags of three consecutive words in the sentence). The system would then be trained to assume that a sentence is grammatically incorrect if it contains a tag trigram that was not seen in training (and possibly if it occurs very rarely in the corpus).

This method has a few disadvantages. One of these is that it can be difficult to understand the error given by the system as there is not a specific error message. This also makes it more difficult to realise when a false positive is given (Naber, 2003). For Icelandic, however, the largest disadvantage is that the largest hand-checked POS tagged corpus is relatively small (around 600,000 tokens). This could result in a lot of false positives in the error detection as there would be a rather high ratio of unseen tag trigrams. However, when one has all the resources needed, the statistical approach to grammar checking can be a good choice as it does not require as much manual work. The system could therefore possibly cover a larger error set than with the rule-based approach, with the same amount of work.

2.3 Rule-based grammar checking

Using the rule-based approach to grammar checking involves manually constructing error detection rules for the language. These rules are then used to find errors in text that has already been analysed, i.e. tagged with a part-of-speech tagger and parsed, so that the grammar checker recognises types of phrases within each sentence, syntactic roles and features of each word, such as case, number, person and gender.

These rules often contain suggestions on how to correct the error found in the text. For example, the grammar checker for English in the Microsoft Word word processor not only points out the errors it has detected but also suggests one or more corrections to the error. The user can then choose whether to ignore the ‘error‘ found (e.g. if it is a false positive or deliberate uncommon usage) or choose one of the correction suggestions from the grammar checker.

A good example of a rule-based grammar checking system is the Rule-based Style and Grammar Checker developed by Naber (Naber, 2003). The system checks text for certain grammatical errors and new rules can easily be added. For each rule in the system a description of the corresponding error and example sentences are provided for the user so as to make it easier to understand the problem and correct it.

2.4 Hybrid grammar checking systems

A good example of a hybrid system is the Swedish grammar checker Granska. Granska uses manually constructed error detection rules. However, in order to check each rule as seldom as possible it also uses the statistics of part-of-speech bigrams and words thus achieving higher efficiency with the same or better results than previous systems (Carlberger, Domeij, Kann and Knutsson, 2004).

3 Topic description

A subset of errors had to be decided upon, i.e. which errors the system should focus on detecting. A few very common errors, among native and non-native speakers, were chosen to start with. These were:

- Disagreement in case within a noun phrase
  Words within a noun phrase should all be in the same case
  Example of error:
  „Hún er góði kennara“
  where *‘góði‘* is in nominative case whereas *‘kennara‘* is in genitive, accusative or dative case
- Disagreement in number within a noun phrase
  Words within a noun phrase should either all be singular or all plural
  Example of error:
  „Þær eru góð vinkonur”
  where *‘góð’* is singular but *‘vinkonur’* is plural
- Disagreement in gender within a noun phrase
  Words within a noun phrase should all be of the same gender
  Example of error:
  „Hún er góð kennari“
  where *‘góð‘* is a feminine adjective whilst *‘kennari‘* is a masculine noun
- Disagreement between the subject and the complement of a sentence
  Words belonging to the subject and complement of a sentence should agree in gender and number
  Example of error:
  „Hún er góður“
  The subject of the sentence is the noun phrase *‘hún‘* and the complement is *‘góður‘*. *‘Hún‘* is a feminine personal pronoun and *‘góður‘* is a masculine adjective. Therefore the subject and the complement are not in agreement.
- Case disagreement in a prepositional phrase
  A preposition in Icelandic can govern a case, govern different cases based on the context or not govern a case at all. If a prepositional phrase includes a preposition that governs a certain case, the nested noun phrase should be in that case.
  Example of error:

> *„Hún hljóp í gegnum skóginum“*
> The prepositional phrase here is *'í gegnum'* and *'gegnum'* is tagged as governing the accusitive case. However, the nested noun phrase: *'skóginum'* is in the dative case and is therefore not in agreement with the preposition.
> This rule should only compare the noun phrase words with those words in the prepositional phrase that govern a case.

Due to the time limitation and ease of implementation of this project the author chose to take the rule-based approach. This approach is also good for developing small systems as it is easy to start with one rule and gradually build up to a larger system (Naber, 2003). For the initial analysis of the input text a parser and part-of-speech tagger are needed. The part-of-speech tagger generates a tag for each word in a sentence. The tag indicates the word's class and morphological features. The parser performs syntactic analysis, adding tags to parts of the sentence marking phrases within it and syntactic roles.

The output from these systems is then used as input to the grammar checker. Therefore, part of the goal for this project was examining whether the string generated in this initial analysis is suitable for detecting grammatical errors or if it would be better to somehow represent the string in a standard form, such as XML.

Using only the rule-based approach, each rule is separate from the others and manually constructed to check for agreement in certain parts of tags found in the relevant phrases/syntactic parts of a sentence. After the initial analysis of the input text the system then goes through the text finding the parts relevant for each rule. Whenever it finds such a part, it checks for compliance with the rule in question. If it decides that the phrase is in some way not in accordance with the relevant rule, an error is generated and stored for the output.

To begin with, the system was meant to detect these errors in a text and somehow indicate that there was an error/errors. The development of a sophisticated user interface was not a priority but it was decided that a very simple interface would be made simply to demonstrate the system in action.

4 Implementation

The resources used in the project are discussed in section 4.1. The rules implemented in this prototype are described in section 4.2. And finally, section 4.3 provides an overview of the system, its output and discusses how the detected errors are communicated to the user.

4.1 Resources

For the initial analysis of the text, the input is tagged with part-of-speech tags using *IceTagger* (Loftsson, 2008) and parsed with a shallow finite state parser, *IceParser* (Loftsson and Rögnvaldsson, 2007a). The grammar checking system then goes through each sentence of the text and inserts a number as the first letter in each tag. This number is the word's position in its original sentence.

An example of this, using the same sentence as in section 2.1:

- After POS tagging and parsing:
  *{*SUBJ> [NP [AP Gamli lkenvf AP] maðurinn nkeng NP] *SUBJ>}*
  [VP gekk sfg3eþ VP]
  [PP [MWE_PP í aa gegnum ao MWE_PP] [NP skóginn nkeog NP] PP]
  . .
- After inserting original location into each tag:
  *{*SUBJ> [NP [AP Gamli 0lkenvf AP] maðurinn 1nkeng NP] *SUBJ>}*
  [VP gekk 2sfg3eþ VP]
  [PP [MWE_PP í 3aa gegnum 4ao MWE_PP] [NP skóginn 5nkeog NP] PP]
  . 6.

4.2 Rules

Five rules were implemented in this prototype. Three are for internal agreement within a noun phrase, one for agreement between a sentence's subject and complement and the last one is for agreement between a governing preposition and a noun phrase within a prepositional phrase.

Case agreement within a noun phrase

This rule searches for the syntactic tags *[NP* and *NP]*, indicating the beginning and end of each noun phrase. It gathers all the noun phrases in a sentence. It then looks at one noun phrase at a time, checks the letter in each tag that indicates case and compares them. As soon as it finds a tag that does not indicate the same case as the noun phrase‘s first word it generates a new error. The error contains an error message, indicating which type of error has been made as well as the location of the two erroneous words in the original sentence. The original location is found by checking the first letter of the tag for the number previously inserted.

Number agreement within a noun phrase

Works exactly like the above rule except it compares the letter in each tag that indicates a word‘s number. The error message is slightly different, indicating that the error has to do with number agreement in the phrase.

Gender agreement within a noun phrase

Works like the above rules except it compares the letter in each tag that indicates a word‘s number. Only the error message is different, indicating gender disagreement within the phrase.

Gender and number agreement between a sentence‘s subject and complement

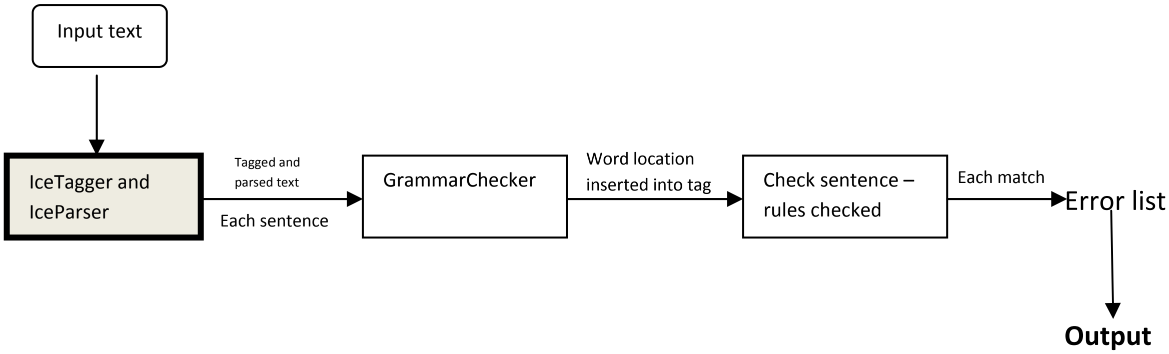
If a sentence includes both a subject and a complement this rule starts by extracting both parts of the sentence. It then finds the subject‘s noun phrase and the complement‘s adjective phrase. It then compares the tags from both phrases to check if they are in agreement for both gender and number. If this is not the case a new error is generated as in the above rules, with the erroneous words‘ original locations in the sentence and a relevant error message.

Case agreement in a prepositional phrase

This rule collects all tags in a prepositional phrase, removes all *‘aa‘* tags as those indicate an adverb (as they do not govern a case) and finally compares the other tags, checking for case agreement. If a discrepancy is found an error is generated as in the above rules and with a relevant error message. There is, however, a slight difference as this rule also checks if the preposition in the phrase is a multiword preposition (such as *‘inn í‘* and *‘út úr‘*). When a preposition is a multiword preposition, one of the words can govern a case and the other be non-governing. However, the grammar checker considers them as one whole when an error is generated, so as to better describe the error for the user in the output. The error thus contains the location of both preposition words, not only the one with the case disagreement according to the tags.

4.3 System overview

The input is run through *IceTagger* and *IceParser*, resulting in a POS tagged string with annotations indicating phrase structures (such as noun phrases and prepositional phrases) and syntactic functions (such as the subject and object of a sentence). Each sentence is then passed, with POS annotation, to the grammar checker where the original word location is added to each POS tag. The sentence is then checked for compliance with each rule of the system. There is no hierarchy for the rules, the system simply goes through them in the order they appear in the code.



If there is more than one type of error within an erroneous sentence part (e.g. case and number disagreement in a noun phrase or gender and number disagreement between a subject and complement) both errors are indicated. An example of this is shown below. Lines 1-3 showing the POS tagged, parsed string with the original word location number inserted in the POS tag. The fourth line showing the two sentence parts extracted (subject and complement), the fifth showing the tags compared and their disagreement. Finally, line 6 shows the contents of the error generated by the disagreement in the tags.

*{*SUBJ> [NP Hún 0fpven NP] *SUBJ>}*
[VPb er 1sfg3en VPb]
*{*COMP< [NP [AP góð 2lvensf AP] kennari 3nken NP] *COMP<}*

[NP Hún 0fpven NP] [NP [AP góð 2lvensf AP] kennari 3nken NP]

*0fp**v**en 2l**v**ensf 3n**k**en* (the first two words are feminine, the third masculine)

Original word locations 0 and 3 and the error message *"Ósamræmi er í kyni frumlags og sagnfyllingar"*

The output prints out one line per detected error in the system. Each line consists of the relevant error message and the original input sentence with the erroneous words in boldface. If the system finds more than one error in the same sentence it prints out an error line for each error as opposed to indicating all the errors in one output sentence. It was considered easier for the user to see where the error was made if only the words making up one error are indicated in each line.

5 Results

Section 5.1 describes two error types that were originally thought of for error detection but weren't included as the necessary resources for detection aren't available. Section 5.2 discusses the testing done, including the test data used. Section 5.3 describes the results these tests provided, including precision and recall of the system. An error in the system found during last-minute testing is discussed in section 5.4.

5.1 Errors the system cannot detect with current resources

'Dative sickness/tendency' (þágufallssýki/-hneigð)

One type of error the author originally intended the system to detect was the growing tendency, especially among children, to use the dative case for a sentence's subject with verbs that should take subjects in the accusative (such as *'langar'*) or even with verbs that should take the nominative (such as *'kvíða'*). This tendency results in an increasing number of Icelandic speakers writing: *"Mér langar"*, *"honum langar"* and *"mig kvíður fyrir"* instead of the correct *"mig langar"*, *"hann langar"* and *"ég kvíði fyrir"*.

However, this type of error cannot be found with the current resources as it requires subcategorization information for verbs for the language – stating which case a subject should be in according to the verb.

Certain case disagreements within a prepositional phrase

Certain case disagreements within a prepositional phrase cannot be found. An example of this error is: *"Ég fór á hest til Akureyrar"* whereas it should be *"Ég fór á hest**i** til Akureyrar"*. '*Á'* can govern either the accusative or the dative case, therefore it is impossible to detect that there is an error when a user writes '*á hest'* as '*hest*' is in the accusative. To detect this kind of error the system would need to possess 'world knowledge', some sort of understanding of what is being written.

5.2 Testing and test data

Very limited testing was performed on the system. The reason for this was twofold. Firstly, procuring the right kind of test data proved harder than originally expected. Secondly, the time that went into manually checking text and getting users to insert errors into it restricted the amount of data that was possible to go through during the development if this prototype.

The input text had to be varied, correctly spelled and preferably POS tagged and parsed, with the tagging and parsing hand-checked, so as to avoid errors in the grammar checking due to incorrect tagging/parsing. However, the text also had to include grammatical errors for the grammar checker to detect. Not a lot of these texts are available.

Originally, it was thought best to get some sort of text from non-native Icelandic speakers as it was assumed that students learning the language might not have the same 'feel for the language' as native speakers and therefore have more grammatical errors in their texts. The Reykjavik University (RU) International Office and the University of Iceland's Department of Icelandic for Foreign Students were therefore contacted. After help from the International Office at RU and contacting several people at the University of Iceland a group of students was contacted and asked for any texts (essays, stories, letters etc.) they might have written as part of their studies. Two students kindly sent their essays to use as test data.

It was obvious that more data was needed to test the system so a story (Little Red Riding Hood) and a few short articles from a newspaper website (www.mbl.is) were split into shorter parts and sent to six different users, ranging in age and background although all are native Icelandic speakers. The users added grammatical errors into the text they received. The texts were gathered again and used for testing.

Finally, the above data (essays, news and story) were tagged and parsed using *IceTagger* and *IceParser* and then run through the grammar checker for errors. For each type of error rule the number of errors in the texts were counted, number of errors found were counted as well as the number of false positives generated by the grammar checker. These numbers were then used to calculate the precision and recall of the system.

5.3 Testing results

The system's output includes an error message for each error detected in a phrase, e.g. if both case and number agreement are detected in the same noun phrase both will be indicated in the output. When counting the detected errors each of these is therefore counted as separate. The table below shows the precision and recall rates for all the errors in the texts as well as the corresponding rates for each type of error. The rates are found as follows:

$$Precision = \frac{number\ of\ correctly\ flagged\ errors}{total\ number\ of\ errors\ flagged}$$

$$Recall = \frac{number\ of\ correctly\ flagged\ errors}{total\ number\ of\ errors\ that\ occur\ in\ text}$$

	Noun phrase errors	**Subject-complement errors**	**Prepositional phrase errors**	**All erros**
Precision:	88,89%	60%	57,13%	84,21%
Recall:	72,72%	30%	30,77%	71,64%

Most of the errors in the texts, by far, were internal noun phrase disagreements (case, number or gender). These errors also have the highest precision and recall rates of all the error types.

The low precision and recall rates of the prepositional phrase rule can partly be explained by the fact that some prepositions in Icelandic (such as *'á'*, *'í'* and *'við'*) can govern two different cases, depending on the context. It is therefore extremely hard for the tagger used to choose the correct tag for these prepositions. As the only thing the grammar checker has to go by is the tag given to each word and the syntactical annotation, these prepositions are rather difficult to handle when checking for grammar errors.

An example of this kind of error is: "*Hún fór á hestbak"* and *"Hún fór á hestbaki"*. The first sentence is correct whereas the second one is incorrect as the noun is in the dative case (when it should be in the accusative). *'Á'* can govern either the accusative or the dative case, depending on the context. '*Hestbak*' is in the accusative and '*Hestbaki*' is in the dative. After being run through the tagger and parser the prepositional phrases are annotated as follows:

*[PP á a**o** [NP hestbak nhe**o** NP] PP]*
*[PP á a**þ** [NP hestbaki nhe**þ** NP] PP]*

In order for the grammar checker to detect a case error the noun in a prepositional phrase would have to be tagged as being in another case than the one the preposition governs (the letter in each tag indicating case is in boldface). In the first sentence, both are tagged as being in the accusative case. In the latter, they are both tagged as being in the dative case. It is therefore impossible for the grammar checker to detect the error being made in the latter sentence.

The fact that most of the errors in the texts were internal noun phrase errors and that these errors have the highest precision and recall rates account for the relatively high total precision and recall rates for the system. It should be noted however that the rates (especially recall rates) for both prepositional phrase and subject-complement rules are very low.

5.4 Known errors in the system

During last-minute testing an error was found in the grammar checking system. Whenever a text including a number representing a year (e.g. 2008) is part of the input an OutOfBounds exception is thrown by the system. This is caused by the year being tagged with the tag *'ta'* by *IceTagger*. This tag is not found in the tagset for Icelandic used when developing the error rules (the tagset used in the Icelandic Frequency Dictionary) and was therefore not accounted for in the rules. Unfortunately this error was not detected early enough for it to be corrected.

6 Conclusion and possible future work

The performance of a grammar checker is extremely dependent on the input, both the original text and the tagging and parsing performed on it. Spelling errors in the input text can create problems in the tagging process so running some sort of spell checker on the text before tagging could be wise.

Representing the input text in XML or other standard form is also probably preferable to using the output string from the parser directly. Using the tagged and parsed string presented a number of trivial problems such as difficulties with finding a word's original location in a sentence after parts of the sentence had been extracted and checked for errors. Problems of this kind might have been avoided had the text been represented in a structured form.

A grammar checker is not of much use if it cannot either suggest corrections or give a good hint as to why it assumes something to be erroneous. The user can then make his or her own decision whether to change that part of the text or not. The current system only prints out a general error message for each error detected and shows the user where the error is to be found by printing the erroneous words in boldface. Therefore, future work could include developing more detailed error messages and/or correction suggestions (when the necessary resources become available).

Furthermore, stylistic checking could be added. This could include checking for deviations from conventional punctuation (such as doubled punctuation, a common typo), number/date formatting or repeated use of the same word (using a thesaurus).

Finally, were the work on a rule-based grammar checker to be continued, some sort of optimization might become necessary so as to improve the speed with which it checks the text. With a growing number of rules in the system it would become increasingly time consuming to check each error rule for a match at every position in the text. Statistical optimization, such as that used to improve the performance of the Swedish grammar checker Granska (Carlberger, Domeij, Kann and Knutsson, 2004) could for instance be implemented. The optimization used in Granska checks which tag combinations are needed to match each rule. Then, instead of trying to match each rule at every position in the text, it only checks the relevant portions of the texts for each rule.

Acknowledgements

The author would like to thank Hrafn Loftsson for all his invaluable input and the use of *IceTagger* and *IceParser*. Sigríður Kristinsdóttir at the Reykjavik University International Office and María Anna Garðarsdóttir who teaches Icelandic for foreign students at the University of Iceland receive thanks for their help with procuring test data. The two students who kindly sent their texts, Branislav Bédi and Cynthia Trililani, receive the author‘s thanks for their help. And finally, the users who helped by inserting errors into the supplemental texts used: thank you all.

References

Arppe, A. (2000). Developing a grammar checker for Swedish. In *Proc. 12th Nordic Conference in Computational Linguistics, Nodalida-99*. Trondheim, Norway.

Carlberger, J., Domeij, R., Kann, V., Knutsson, O. (2004). The development and performance of a grammar checker for Swedish: A language engineering perspective.

Carlberger, J., Domeij, R., Kann, V., Knutsson, O. (2000). A Swedish Grammar Checker.

Hagen, K., Johannessen, J. B. and Lane, P. (2001). Some problems related to the development of a grammar checker. Paper presented at NODALIDA '01, the 2001 Nordic Conference in Computational Linguistics, May 21--22, 2001.

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

Loftsson, H. and Rögnvaldsson, E. (2007b). IceNLP: A Natural Language Processing Toolkit for Icelandic. In *Proceedings of InterSpeech 2007, Special session: "Speech and language technology for less-resourced languages"*. Antwerp, Belgium.

Loftsson, H. (2008). Tagging Icelandic text: A linguistic rule-based approach. Appeared in a revised form, subsequent to editorial input by Cambridge University Press, in Nordic Journal of Linguistics, 31(1), 47-72. © 2008 Cambridge University Press.

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

Naber, D. (2003). A Rule-Based Style and Grammar Checker. Diplomarbeit. Technische Fakultät Bielefeld.

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

Contents