A Tool for Automatic Simplification of Swedish Texts

Evelina Rennes
SICS East Swedish ICT AB, Sweden
evelina.rennes@liu.se

Arne Jönsson
SICS East Swedish ICT AB, Sweden
arne.jonsson@liu.se

Abstract

We present a rule based automatic text simplification tool for Swedish. The tool is designed to facilitate experimentation with various simplification techniques. The architecture of the tool is inspired by and partly built on a previous text simplification tool for Swedish, CogFLUX. New functionality, new operation types, and new simplification operations were added.

1 Introduction

The task of automatic text simplification aims to reduce the overall complexity of a text, in order to enhance comprehension for a human reader, or to improve further processing performed by a computer program. The simplification of texts have previously been performed manually, but since this is a very time consuming and expensive task, the possibility to automatically create simplifications of texts would result in more accessible information, to a relatively low cost. Recent years' increase in computer power, and the availability of high quality linguistic resources and language processing tools enable faster, better, and more powerful tools for natural language processing needed for more advanced text simplification.

The group of people that might benefit from a text simplification tool is not homogeneous and therefore, it is important to account for the differences between these groups, and perhaps also look for the individual needs among the group members. The simplification tool can be used to study such individual simplication needs.

Although automatic summarization has been pointed out as a possible method of simplifying a text (Margarido et al., 2008; Smith and Jönsson, 2011), simplifications are not always shorter than the originals. For example, a simplification operation applied to a syntactically complex sentence might result in a longer sentence with a less complex grammatical structure, and some readers might benefit from more extensive explanations of terms or certain phenomena in order to gain full understanding of a text.

1.1 Text Simplification in Swedish

A study on simplification operations for Swedish was made by comparing the phrase structures of a text written in Standard Swedish to a manually simplified version of the text, and subsequently extracting simplification operations (Decker, 2003). This work resulted in 25 extracted simplification rules. The rules were grouped into two subsets: rules that delete or replace sub-phrases and rules that add new syntactical information to the text. The CogFLUX system (Rybing et al., 2010), implemented the first subset of this rule set. Abrahamsson (2011) developed the tool further by adding another subset of the rules, and an additional synonym replacement module.

Simplification through synonym replacement has been investigated by evaluating and comparing different methods for choosing alternative synonyms (Keskisärkkä and Jönsson, 2012). In that work, the success of the simplification used measures such as readability metrics, average word length, proportion of long words, and replacement error ratio. Synonym replacement was also the main interest in a study of simplification of Swedish medical texts (Abrahamsson et al., 2014), that replaced difficult medical terms with synonyms that were considered easier, by applying the two Swedish readability metrics LIX and OVIX to the texts. The difficulty of a given word was estimated by taking into account both the frequency of the word in a general corpus, but also the frequency of substrings of words. The result showed that the resulting text was slightly more difficult according to LIX, while being more readable according to OVIX.

2 The tool

The architecture of the text simplification tool presented in this paper is inspired by and partly built on CogFLUX (Rybing et al., 2010).

2.1 Layout

The applet, in its current state, consists of two main fields. An upper white field where the original text is inserted, and a lower white field contains the output of the simplification.

By the use of check boxes, the user decides what simplification operation to apply to the text.

The tool surrently supports the following simplifications: passive-to-active, quotation inversion, rearranging to straight word order, sentence split, synonym replacement, and the simplification rules extracted by Decker (2003). For our current experimental purposes, the first five are divided into three groups, corresponding to the estimated level of simplification.

- Low
  - Sentence Split
  - Quotation Inversion
- Medium
  - Low level operations +
  - Passive-to-active
  - Straight word order
- High
  - Medium level operations +
  - Synonym replacement

The user, or experimenter, can easily try either the pre-defined groups, or any combination of simplifications.

2.2 Linguistic Resources

The linguistic resources used in this project were the SUC3 corpus and the Synlex synonym lexicon.

The Swedish Language Bank (*Språkbanken*), has since the seventies developed and stored a large collection of Swedish text corpora. One of these is the Stockholm-Umeå Corpus (Ejerhed et al., 2006), which is a balanced corpus consisting of one million words, annotated with part-of-speech tags, morphological features and citation form

The synonym replacement module was built on the work of Abrahamsson (2011) using the Synlex lexicon (Kann, 2004), with an included frequency list. Synlex is a free linguistic resource containing about 80000 synonym pairs. The collection of synonym pairs was constructed in cooperation with voluntary Internet users, by giving suggestions of possible synonyms and giving the users the possibility to rate the correctness of the suggestion of a given synonym pair.

2.3 Preprocessing

For tagging we use Stagger (Östling, 2013), a fairly new part-of-speech tagger based on the averaged perceptron (Collins, 2002). It is currently the most accurate tagger for Swedish. Per-token accuracy is estimated to about 96.6 % (10-fold cross validation on SUC 3.0).

For syntactic analysis we use MaltParser 1.2 (Nivre et al., 2006) as the latest version, MaltParser 1.7.2, does not produce phrase structure trees, which in the current phase of the project is needed for interpretation of the rules. However, future functionality might benefit from dependency parsing, which can be turned on with a simple switch.

2.4 Simplification

The simplification rules were formalized to fit *X-rules*, the syntax notation script used in CogFLUX (Rybing et al., 2010). Originally, there were two different types of possible operations, replace (REPL) and delete (DEL). For this project, two additional operation types were created for the purpose of this study, SHIFT and SPLIT. After each operation type there is a target phrase, i.e. the type of phrase that is to be manipulated, followed by an arrow pointing towards the substitute phrase. In the REPL operation the substitute phrase consists of the replacement phrase structure, while the DEL operation completely removes the target phrase. The notation of the SHIFT operation is slightly different. Given a target phrase (to the left of the arrow), the second part of the rule indicates what part of the structure that will change position. This specific operation handles changes of word order, and in order to avoid erroneous rearrangement of words, this operation is only triggered by certain syntactic tags, for example the passive tense. The SPLIT operation simply splits a sentence into two when the condition to the left is fulfilled.

A functionality that was added to the *X-rules* script is the possibility to add dependency tags to the part-of-speech tags. This was made in order to make full use of the information provided by the parser. Another additional development was the introduction of the "?" tag, which is able to represent one, many or no tags of any sort.

All the syntactic rules that were previously applied in the CogFLUX project were included in this project.

This work did not take into account all the simplification operations suggested by the previously conducted literature overview, but limited the applied operations to 4 rules. Other operations were not included in this first iteration due to the nature of the actions: they are either too complex for the tool in its current state, or they cannot be easily included without a foregoing text analysis. The aim is, however, to continue the development of the simplification tool, and eventually apply all the proposed operations.

The syntactic simplification applied in this project consisted of 4 separate rules:

- Changing from passive to active voice

  To transform a sentence from passive to active voice in Swedish, the subject of the passive sentence must become the object of the active sentence. The s-ending must be deleted and the preposition *av* (*by*, for example *the cookie was eaten* ***by*** *the boy*) must be removed. In order to perform this, a sequence of operations were applied when a sentence of passive voice was detected:

  SHIFT//S-NP(SS) VB/VP ? PP(AG) → S-PP NP &P(#)

  input: *The huge cookie was eaten by both Kalle and Stina in the dining room.*
  output: *Both Kalle and Stina ate the huge cookie in the dining room.*

- Quotation Inversion

  The quotation inversion changes the place of the speaker in a quotation, from *[quotation], said X* to *X said: [quotation]*) (Bott et al., 2012).

  The quotation inversion operation is triggered by quotation marks followed by specific words from a lexicon that might indicate a quotation (such as *said, exclaimed, whispered*, etc. and the quotation (specified by the quotation marks) switches place with the verb phrase and the noun phrase, such as:

  input: *"Go to bed!" said Kalle.*
  output: *Kalle said: "Go to bed!"*

- Rearranging to straight word order

  This rule shifts the word order of clauses initiated with adverbs or adjectives.

  SHIFT//S-AVP/AP VB/VP NP(SS) ? → S-NP(SS) AVP &P(#) (1)
  SHIFT//S-AVP/AP VB/VP NP(SS) ? → S-AVP ? &P(#) (2)

  The application of this simplification operation might result in the following example:

  input: *Yesterday bought Kalle a new car.**
  output: *Kalle bought a new car yesterday.*

- Sentence split

  The SPLIT operation splits a sentence in two. In the example rule below, a clause that is consisting of two clauses joined with a conjunction, is split at the word marked as a conjunction and two separate sentences are created, inserting a full stop as an end of sentence marker.

  SPLIT//S-S KN S → KN &P(#)

  The rule type is dynamic and the breaking point can easily be changed by changing the second part of the rule.

3 Conclusion

This report described a framework for syntactical and lexical text simplification for Swedish. The architecture of the tool is inspired by and partly built on a previous text simplification tool for Swedish, CogFLUX, but has been modified with new functionality. Two new operation types were added, SHIFT and SPLIT, and four new simplification operations were applied: changing from passive to active word order, quotation inversion, rearranging to straight word order, and sentence split.

The tool is mainly intended to be used for experiments on rule based text simplification techniques.

Acknowledgments

This research was supported by SICS East Swedish ICT AB.

References

Emil Abrahamsson, Timothy Forni, Maria Skeppstedt, and Maria Kvist. 2014. Medical text simplification using synonym replacement: Adapting assessment of word difficulty to a compounding language. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)@ EACL*.

Peder Abrahamsson. 2011. Mer lättläst - påbyggnad av ett automatiskt omskrivningsverktyg till lätt svenska. Bachelor's thesis, Linköping University.

Stefan Bott, Horacio Saggion, and Simon Mille. 2012. Text simplification tools for spanish. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA. Association for Computational Linguistics.

Anna Decker. 2003. Towards automatic grammatical simplification of swedish text. Master's thesis, Stockholm University.

Eva Ejerhed, Gunnel Källgren, and Benny Brodda. 2006. Stockholm Umeå Corpus version 2.0.

Viggo Kann. 2004. Folkets användning av lexin – en resurs. Technical report, KTH Nada.

Robin Keskisärkkä and Arne Jönsson. 2012. Automatic text simplification via synonym replacement. In *Proceedings of The Fourth Swedish Language Technology Conference*.

Paulo R. A. Margarido, Thiago A. S. Pardo, Gabriel M. Antonio, Vinícius B. Fuentes, Rachel Aires, Sandra M. Aluísio, and Renata P. M. Fortes. 2008. Automatic summarization for text simplification: Evaluating text understanding by poor readers. In *Companion Proceedings of the XIV Brazilian Symposium on Multimedia and the Web*.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of the Fifth International Conference on Language Resources and Evalutation*, pages 2216–2219.

Robert Östling. 2013. Stagger: an open-source part of speech tagger for swedish. *Northen Europea Journal of Language Technology*, 3.

Jonas Rybing, Christian Smith, and Annika Silvervarg. 2010. Towards a rule based system for automatic simplification of texts. *Proceedings of SLTC*.

Christian Smith and Arne Jönsson. 2011. Automatic Summarization As Means of Simplifying Texts, an Evaluation for Swedish. In *Proceedings of the 18th Nordic Conference of Computational Linguistics (NoDaLiDa-2010)*.