Services for Text Simplification and Analysis

Johan Falkenjack, Evelina Rennes, Daniel Fahlborg, Vida Johansson, Arne Jönnson
Linköping University and RISE SICS East AB
Linköping, Sweden
johan.falkenjack@liu.se, evelina.rennes@liu.se, arne.jonsson@liu.se

Abstract

We present a language technology service for web editors’ work on making texts easier to understand, including tools for text complexity analysis, text simplification and text summarization. We also present a text analysis service focusing on measures of text complexity.

1 Introduction

Our research on digital inclusion has resulted in a variety of tools for making texts easier to understand, including an automatic text summarizer (Smith and Jönsson, 2011a; Smith and Jönsson, 2011b), syntactic (Rennes and Jönsson, 2015) and lexical (Keskisärkkä and Jönsson, 2013; Johansson and Rennes, 2016) simplification, and a number of measures of text complexity (Falkenjack et al., 2013; Falkenjack and Jönsson, 2014).

In the project TeCST¹ (Text Complexity and Simplification Toolkit) we developed a web service that integrated all these tools and made them easily available for producers of easy-to-read texts. The service can also be used by end users, i.e. anyone wanting to make a text easier to comprehend. Similar systems exist for other languages, such as Spanish (Saggion et al., 2015), Brazilian Portuguese (Scarton et al., 2010) and English (Lee et al., 2016).

The set of text complexity measures provided by the TeCST service is limited to a subset of features that is meant to be easily understandable by non-linguists. The complete set of features has instead been made available in the separate service SCREAM². All services presented in this paper can also be accessed through the SAPIS REST API (Fahlborg and Rennes, 2016). In what follows we will first present the features included in SCREAM and continue with the tools included in TeCST.

2 Text analysis

All tools in TeCST use Stagger (Östling, 2013) for tagging and MaltParser (Nivre et al., 2007) for parsing. We have also implemented support for the OpenNLP part-of-speech tagger (Morton et al., 2005) and older versions of MaltParser.

3 SCREAM – Text complexity features

SCREAM currently comprises 117 features.

3.1 Shallow features

Shallow text features are features that can be extracted after tokenization by simply counting words and characters. They include:

- **MWLC** Mean word length calculated as the average number of characters per word.
- **MWLS** Mean word length calculated as the average number of syllables per word. The number of syllables is approximated by counting the number of vowels.
- **MSL** Mean sentence length calculated as the average number of words per sentence.

3.2 Lexical features

Our lexical features are based on categorical word frequencies extracted after lemmatization and calculated using the basic Swedish vocabulary SweVoc (Heimann Mühlenbock, 2013). SweVoc is somewhat comparable to the list used in the classic Dale-Chall formula for English (Dale and Chall, 1949). Though developed for similar purposes, special sub-categories have been added (of which three are specifically considered). The following frequencies are calculated:
**SweVocC** SweVoc lemmas fundamental for communication (category C).

**SweVocD** SweVoc lemmas for everyday use (category D).

**SweVocH** SweVoc other highly frequent lemmas (category H).

**SweVocT** Unique, per lemma, SweVoc words (all categories, including some not mentioned above) per sentence.

### 3.3 Morpho-syntactic features

The morpho-syntactic features concern a morphology based analysis of text. The analysis relies on previously part-of-speech annotated text, which is investigated with regard to the following features:

**UnigramPOS** Unigram probabilities for 26 different parts-of-speech tags in the document, i.e. the ratio of each part-of-speech, on a per token basis, as individual attributes. Such a unigram language model based on part-of-speech, and similar metrics, has shown to be a relevant feature for readability assessment for English (Heilman et al., 2007; Petersen, 2007; Dell’Orletta et al., 2011). The tag set used is collected from the Stockholm-Umeå Corpus (Ejerhed et al., 2006).

**RatioContent** The ratio of content words (nouns, verbs, adjectives and adverbs), on a per token basis, in the text. Such a metric has been used by for instance Alusio et al. (2010).

### 3.4 Syntactic features

These features are estimable after syntactic parsing of the text. The dependency based features consist of:

**ADDD** The average dependency distance in the document on a per dependent basis. A longer average dependency distance could indicate a more complex text (Liu, 2008).

**ADDS** The average dependency distance in the document on a per sentence basis. A longer average total dependency distance per sentence could indicate a more complex text (Liu, 2008).

**RD** The ratio of right dependencies to total number of dependencies in the document. A high ratio of right dependencies could indicate a more complex text.

**SD** The average sentence depth. Sentences with deeper dependency trees could be indicative of a more complex text in the same way as phrase grammar trees have been shown to be (Petersen and Ostendorf, 2009).

**Dependency type tag ratio** Unigram probabilities for the dependency type tags resulting from the dependency parsing, on a per token basis, as individual parameters. This is viewed as a single feature but is represented by 63 parameters. These parameters make up a unigram language model and is comparable to the phrase type rate based on phrase grammar parsing used in earlier research (Nenkova et al., 2010).

**VR** The ratio of sentences with a verbal root, that is, the ratio of sentences where the root word is a verb to the total number of sentences (Dell’Orletta et al., 2011).

**AVA** The average arity of verbs in the document, calculated as the average number of dependants per verb (Dell’Orletta et al., 2011).

**UVA** The ratio of verbs with an arity of 0-7 as distinct features (Dell’Orletta et al., 2011). This is viewed as a single feature but is represented by 8 parameters.

**TPC** The average number of tokens per clause in the document. This is related to the shallow feature average number of tokens per sentence.

**PreM** The average number of nominal pre-modifiers per sentence.

**PostM** The average number of nominal post-modifiers per sentence.

**PC** The average number of prepositional complements per sentence in the document.

**TokensPerClause** The average number of tokens per clause in the document. This is related to the shallow feature average number of tokens per sentence.

**PreModifiers** The average number of nominal pre-modifiers per sentence.

**PostModifiers** The average number of nominal post-modifiers per sentence.
PrepComp The average number of prepositional complements per sentence in the document.

3.5 Text quality metrics
The three most used traditional text quality metrics used to measure readability for Swedish are:

LIX Läsbarhetsindex, readability index. Ratio of words longer than 6 characters coupled with average sentence length.

OVIX Ordvariationsindex, word variation index, which is essentially a reformulation of type-token ratio less sensitive to text length.

NR Nominal ratio, the ratio of nominal word, used to measure formality of text rather than readability, however, this is traditionally assumed to correlate to readability.

4 TecST
TecST consists of a subset of the features from SCREAM, the text simplifier StILLET, and the text summarizer FRIENDLYREADER.

4.1 StILLET
StILLET is a rule-based tool for automatic text simplification in Swedish. StILLET was originally developed as an extension of CogFlux (Rybing et al., 2010) that included a set of text rewriting operations (Decker, 2003). The first version of StILLET (Rennes and Jönsson, 2015) was extended to support additional rules; rewriting from passive to active tense, quotation inversion, rearrangement to straight word order, and sentence split. Due to the inefficiency of phrase based parsers, a new version of StILLET was developed, now relying on dependencies, providing faster simplification. We are currently working on methods for the automatic extraction of simplification operations based on an aligned corpus of simplified and regular texts (Rennes and Jönsson, 2016; Albertsson et al., 2016). The automatically harvested rules will eventually be included in addition to the existing rule sets.

4.2 FRIENDLYREADER
FRIENDLYREADER (Smith and Jönsson, 2011a; Smith and Jönsson, 2011b) is the automatic text summarizer used in TecST that extracts the most important sentences in a text based on distributional semantics. It uses a word space model, in this case Random Indexing (RI) (Hassel, 2007; Hassel, 2011) with pre-trained word vectors. Furthermore, to handle long sentences with many words, the mean document vector is subtracted from each of the sentence’s word vectors before summarizing the vectors (Higgins and Burstein, 2007). FRIENDLYREADER does not directly use a vector distance metric to select sentences, instead it uses the Weighted PageRank algorithm to rank the sentences (Chatterjee and Mohan, 2007). In this case each vertex depicts a unit of text and the edges between the units represent a connection between the corresponding text units, c.f. TextRank (Mihalcea, 2004). Thus, the importance of a vertex within a graph considers global information from the entire graph, not only the local context of the vertices, as ranks are recursively computed so that the rank of a vertex depends on all the vertices’ ranks.

5 SAPIS
SAPIS³ (StILlet SCREAM API Service) is a back-end solution providing the calculation of text complexity features (SCREAM) and the application of simplification operations (StILLET) on a remote server. SAPIS is able to present simplification feedback on a sentence level by identifying sentences where any of the rules in StILLET is applicable. A textual feedback is returned for each sentence that matches any of the patterns given in the simplification rule sets.

SAPIS also provides simple access to part-of-speech tagging and dependency parsing.

6 Conclusions
We have presented a service that integrates a variety of tools aiming to make texts easier to understand. Current work focuses on corpus collection for StILLET, and interaction design to improve usability and make the measures of text complexity easier to interpret.

Acknowledgments
This research is financed by Internetfonden, Vinnova and SweClarin.

References
Sarah Albertsson, Evelina Rennes, and Arne Jönsson. 2016. Similarity-based alignment of monolingual


