Classifying easy-to-read texts without parsing

Johan Falkenjack, Arne Jönsson
Department of Information and Computer Science
Linköping University
581 83, Linköping, Sweden
johan.falkenjack@liu.se, arne.jonsson@liu.se

Abstract

Document classification using automated linguistic analysis and machine learning (ML) has been shown to be a viable road forward for readability assessment. The best models can be trained to decide if a text is easy to read or not with very high accuracy, e.g. a model using 117 parameters from shallow, lexical, morphological and syntactic analyses achieves 98,9% accuracy.

In this paper we compare models created by parameter optimization over subsets of that total model to find out to which extent different high-performing models tend to consist of the same parameters and if it is possible to find models that only use features not requiring parsing. We used a genetic algorithm to systematically optimize parameter sets of fixed sizes using accuracy of a Support Vector Machine classifier as fitness function.

Our results show that it is possible to find models almost as good as the currently best models while omitting parsing based features.

1 Introduction

The problem of readability assessment is the problem of mapping from a text to some unit representing the text's degree of readability. Measures of readability are mostly used to inform a reader how difficult a text is to read, either to give them a hint that they may try to find an easier to read text on the same topic or simply to inform them that a text may take some time to comprehend. Readability measures are mainly used to inform persons with reading disabilities on the complexity of a text, but can also be used to, for instance, assist teachers with assessing the reading ability of a student. By measuring the reading abilities of a person, it might also be possible to automatically find texts that fits that persons reading ability.

Since the early 2000s the speed and accuracy of text analysis tools such as lemmatizers, part-of-speech taggers and syntax parsers have made new text features available for readability assessment. By using machine learning a number of researchers have devised innovative ways of assessing readability. For instance, phrase grammar parsing has been used to find the average number of sub-clauses, verb phrases, noun phrases and average tree depth (Schwarm and Ostendorf, 2005).

The use of language models to assess the degree of readability was also introduced in the early 2000s (Collins-Thompson and Callan, 2004) and later combined with classification algorithms such as support vector machines to further increase accuracy (Petersen, 2007; Feng, 2010).

In this paper we investigate if it is possible to find a set of parameters for easy-to-read classification, on par with the best models used today, without using parsing based features. Finding such a set would facilitate portability and provide faster assessment of readability.

2 Method

To train and test our classifier we used one easy-to-read corpus and five corpora representing ordinary language in different text genres. The latter corpora is referred to as non-easy-to-read in this paper. For each category we used 700 texts.

Our source of easy-to-read material was the LäSBarT corpus (Mühlenbock, 2008). LäSBarT consists of manually created easy-to-read texts

Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR) @ EACL 2014, pages 114–122,
Gothenburg, Sweden, April 26-30 2014. ©2014 Association for Computational Linguistics

from a variety of sources and genres.

The non-easy-to-read material comprised texts from a variety of corpora. This material consisted of 215 news text articles from GP2007 (The Swedish news paper Göteborgs Posten), 34 whole issues of the Swedish popular science magazine Forskning och Framsteg, 214 articles from the professional news magazine Läkartidningen 05 (physician news articles), 214 public information notices from The Public Health Agency of Sweden (Smittskyddsinstitutet) and 23 full fiction novels from a Swedish book publisher (the Norstedts publishing house).

By using a corpus with such a variety of documents we got non-easy-to-read documents from different genres which is important as we want to be able to use the same model on all types of text. We also lowered the risk of genre classification rather than degree of readability classification.

The texts were preprocessed using the Korp corpus import tool (Borin et al., 2012). Steps in the preprocessing chain relevant for this study were tokenization, lemmatisation, part-of-speech tagging and dependency grammar parsing.

We used a large number of different text features proposed for readability assessment for both Swedish and English. We use both the term's feature (property of the text) and parameter (input to the ML-system). Some features consist of more than one parameter. In the paper we use the terms features and parameters somewhat interchangeably. However, technically, a feature is a property of the text, a parameter is input to the machine learning system. A few of the text features we use are represented as a combination of parameters and in these cases we select single parameters, not full features.

2.1 Non-parsing features

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, Equation 1. This is the standard readability measure used for Swedish and can be considered baseline similar to the Flesch-Kincaid formula (Kincaid et al., 1975).

$$lix = \frac{n(w)}{n(s)} + (\frac{n(words > 6\,chars)}{n(w)} \times 100) \tag{1}$$

where $n(s)$ denotes the number of sentences and $n(w)$ the number of words.

OVIX Ordvariationsindex, word variation index, related to type-token ratio. Logarithms are used to cancel out type-token ratio problems with variable text length, Equation 2.

$$ovix = \frac{log(n(w))}{log(2 - \frac{log(n(uw))}{log(n(w))})} \tag{2}$$

where $n(w)$ denotes the number of words and $n(uw)$ the number of unique words.

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, Equation 3.

$$Nr = \frac{n(noun) + n(prep) + n(part)}{n(pro) + n(adv) + n(v)} \tag{3}$$

where $n(noun)$ denotes the number of nouns, $n(prep)$ the number of prepositions, $n(part)$ the number of participles, $n(pro)$ the number of pronouns, $n(adv)$ the number of adverbs, and $n(v)$ the number of verbs.

2.1.1 Shallow features

The shallow text features are the main features traditionally used for simple readability metrics. They occur in the "shallow" surface structure of the text and can be extracted after tokenization by simply counting words and characters. They include:

AWLC Average word length calculated as the average number of characters per word.

AWLS Average word length calculated as the average number of syllables per word. The number of syllables is approximated by counting the number of vowels.

ASL Average sentence length calculated as the average number of words per sentence.

Longer sentences, as well as longer words, tend to predict a more difficult text as exemplified by the performance of the LIX metric and related metrics for English. These types of features have been used in a number of readability studies based on machine learning (Feng, 2010) and as baseline when evaluating new features (Pitler and Nenkova, 2008).

2.1.2 Lexical features

Our lexical features are based on categorical word frequencies. The word frequencies are extracted after lemmatization and are calculated using the basic Swedish vocabulary SweVoc (Heimann Mühlenbock, 2013). SweVoc is 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, based on different categories in SweVoc:

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.

A high ratio of SweVoc words should indicate a more easy-to-read text. The Dale-Chall metric (Chall and Dale, 1995) has been used as a similar feature in a number of machine learning based studies of text readability for English (Feng, 2010; Pitler and Nenkova, 2008). The SweVoc metrics are also related to the language model features used in a number of studies (Schwarm and Ostendorf, 2005; Heilman et al., 2008).

2.1.3 The morpho-syntactic features

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

Part-of-speech tag ratio Unigram probabilities for the different parts-of-speech tags in the document, that is, the ratio of each part-of-speech, on a per token basis, as individual parameters. This is viewed as a single feature but is represented by 26 parameters, see Table 2. Such a 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) and for Swedish (Falkenjack et al., 2013).

RC The ratio of content words (nouns, verbs, adjectives and adverbs), on a per token basis, in the text. Such a metric has been used in a number of related studies (Alusio et al., 2010).

2.2 Parsing based features

These features are estimable after syntactic parsing of the text. The syntactic feature set is extracted after dependency parsing using the Maltparser (Nivre et al., 2006). Such parsers have been used for preprocessing texts for readability assessment for Italian (Dell'Orletta et al., 2011). 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 has 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, see Tables 4 and 5.

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). Such a language model was shown to be a good predictor for degree of readability in Swedish text (Falkenjack et al., 2013).

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 dependents 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 premodifiers per sentence.

PostM The average number of nominal postmodifiers per sentence.

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

Compound models We have also created a number of compound models, comprising metrics from sets of features; all traditional measures, all shallow features, all lexical features, all morpho-syntactic features, all syntactic features, and all features (Total), see Table 3. Falkenjack et al. (2013) also looked at incremental combinations of these same models.

2.3 Parameter optimization

The models for parameter optimization are created from various subsets of the text features using a genetic algorithm. Lau (2006) performed experiments on using genetic algorithms to select significant features that are useful when assessing readability for Chinese. Starting with 64 features, mainly various stroke features but also more traditional features, such as, measuring amount of familiar and common words, a genetic algorithm was used to find optimal feature subsets. Based on investigations of using three different fitness functions it was shown that a set of 15 features is sufficient and the best feature set for each fitness function is selected for further studies. These feature sets are then evaluated using SVR (Support Vector Regression) to train readability models and finally test them on the texts.

In our work we do not first select feature sets and then train the model on them. Instead feature sets, generated by genetic search, are used to train the readability model, using SVM, and then the models are tested.

We performed a number of trials based on different base sets of parameters. In each case the space we searched through had the size $\binom{|b|}{s}$, where b is the base set of parameters and s is the size of the model we were searching for.

We performed genetic searches through model spaces for 1000 generations. Each generation contained 10 chromosomes, i.e. models, 7 created by crossover and 3 randomly generated to avoid getting stuck in local maxima.

The crossover worked by randomly selecting parameters from the locally optimal parameter set of the prior generation. This locally optimal parameter set was created by taking the union of the best performing chromosomes until the size of the set exceeded the size of the target selection plus 4.

In the rare cases where the parameters in the total parent generation did not exceed this number all parameters from the parent generation were used.

The fitness function consisted of a 7-fold cross-validation test run of a Support Vector Machine trained by Sequential Minimal Optimization (Platt, 1998). For this we used the Waikato Environment for Knowledge Analysis, or Weka. The accuracy of a model was used as its fitness and used to order each generation from best to worst performing.

3 Results

We first present results from using only the single features and the compound models. We then present the results from the various models generated by our method.

We provide performance measures for single features for comparison in Tables 1 and 2. The performance for the 63 dependency types are presented in Tables 4 and 5.

		LäSBarT		Other	
Model	Accuracy	Prec.	Rec.	Prec.	Rec.
LIX	84.6 (1.9)	87.9	80.4	82.0	88.9
OVIX	85.6 (2.3)	86.8	84.4	84.9	86.9
NR	55.3 (9.1)	53.5	99.1	96.0	11.4
AWLC	79.6 (2.6)	82.3	75.7	77.4	83.4
AWLS	75.6 (2.6)	78.7	70.3	73.1	80.9
ASL	62.4 (8.1)	58.0	98.7	97.8	26.1
SweVocC	79.3 (0.8)	84.3	72.0	75.6	86.6
SweVocD	57.6 (3.8)	63.1	37.9	55.5	77.4
SweVocH	63.1 (4.5)	63.1	63.4	63.2	62.9
SweVocT	75.2 (1.4)	80.6	66.7	71.6	83.7
POS-tags	*96.8 (1.6)*	*96.9*	*96.7*	*96.7*	*96.9*
RC	50.4 (1.8)	50.4	52.7	50.4	48.1
ADDD	88.5 (2.0)	88.5	88.6	88.6	88.4
ADDS	53.9 (10.2)	52.8	99.7	28.1	8.1
RD	68.9 (2.1)	70.6	65.1	67.7	72.7
SD	75.1 (3.5)	79.1	68.4	72.2	81.9
Dep-tags	*97.9 (0.8)*	*97.7*	*98.0*	*98.0*	*97.7*
VR	72.6 (2.0)	77.0	64.6	69.5	80.6
AVA	63.4 (3.0)	64.9	58.4	62.3	68.4
UVA	*68.6 (1.7)*	*70.2*	*65.0*	*67.4*	*72.3*
TPC	71.4 (4.7)	64.2	98.6	97.0	44.3
PreM	83.4 (2.9)	78.1	93.0	91.3	73.9
PostM	57.4 (4.3)	54.1	99.9	98.4	15.0
PC	83.5 (3.5)	80.1	89.1	88.1	77.9

Table 1: Performance of the single feature models. The accuracy represents the average percentage of texts classified correctly, with the standard deviation within parentheses. Precision and Recall are also provided for both easy-to-read (LäSBarT) and non-easy-to-read (Other) sets. Italicized features consist of more than one parameter.

The results from using the full sets before parameter optimization are listed in Table 3. Using all features provides the best model with 98.9% accuracy which could be considered the target accuracy of our parameter optimization.

		LäSBarT		Other	
Model	Accuracy	Prec.	Rec.	Prec.	Rec.
VB	87.6 (1.7)	89.2	85.9	86.5	89.4
MAD	87.1 (0.9)	91.1	82.3	83.9	91.9
PAD	79.5 (1.6)	71.8	97.4	96.0	61.6
MID	76.6 (2.9)	78.6	73.3	74.9	79.9
PP	72.4 (3.8)	73.7	69.7	71.4	75.0
PN	72.1 (2.7)	79.2	60.4	67.9	83.9
NN	70.4 (2.6)	75.4	61.4	67.3	79.4
DT	67.7 (3.3)	67.9	67.6	67.6	67.9
PL	65.6 (2.5)	70.4	53.9	62.8	77.4
JJ	64.1 (4.3)	63.6	65.7	64.7	62.4
HA	62.4 (1.1)	66.5	49.9	59.9	74.9
SN	59.4 (3.7)	64.7	42.1	57.0	76.7
UO	58.2 (8.2)	55.1	98.4	94.6	18.0
KN	56.6 (3.0)	57.9	48.9	55.7	64.4
AB	56.0 (3.2)	58.4	43.0	54.7	69.0
IN	53.0 (5.1)	60.0	78.7	16.1	27.3
IE	52.6 (2.4)	61.5	19.0	51.5	86.1
PS	52.6 (1.4)	59.4	17.7	51.5	87.4
HP	52.5 (5.4)	69.9	24.0	47.2	81.0
HS	52.4 (2.0)	51.2	99.7	89.3	5.0
RG	51.6 (3.5)	51.1	96.9	69.6	6.4
HD	50.4 (0.7)	50.2	31.7	35.9	69.1
PLQS	50.0 (0.0)	50.0	100.0	0.0	0.0
RO	49.7 (0.9)	49.8	89.3	48.8	10.1
PM	49.7 (1.3)	49.8	95.0	54.9	4.4

Table 2: Performance of the POS-tag ratio parameters ordered by performance. The various models are tags used in the SUC corpus (Ejerhed et al., 2006), normally part of speech tags, e.g. VB is verb, with some extensions, but the tags comprise other features as well e.g. MAD comprises sentence terminating delimiters, PAD pair-wise delimiters such as parentheses and MID other delimiters such as comma and semicolon. Measures as described in Table 1.

		LäSBarT		Other	
Model	Acc.	Pre.	Rec.	Pre.	Rec.
TradComb	91.4 (3.0)	92.0	91.0	91.1	91.9
Shallow	81.6 (2.7)	83.3	79.4	80.3	83.9
Lexical	78.4 (2.2)	81.8	73.0	75.6	83.7
Morpho	96.7 (1.6)	96.8	96.7	96.7	96.7
Syntactic	98.0 (1.1)	97.9	98.1	98.1	97.9
Total	98.9 (1.0)	98.9	98.9	98.9	98.9

Table 3: Performance of the full feature sets. Measures as described in Table 1.

3.1 POS-ratio features

The first trial we performed was a search through the parameter space containing ratios of part-of-speech unigrams. As our data contained 26 different POS-tags (additional morphological data was ignored in this search) the size of the spaces were $\binom{26}{s}$ where s is the size of the model we were optimizing. For 3-parameter models this is no larger than $\binom{26}{3} = 2600$ while the maximum size is $\binom{26}{13} = 10400600$. We searched for optimal subsets of sizes from 1 to 25. The best models are presented in Table 6 and the performance results in Table 8. Models comprising more than 10 features are omitted as no significant performance improvement is measured beyond this point. See Table 7 for sizes.

		LäSBarT		Other	
#	Accuracy	Prec.	Rec.	Prec.	Rec.
IP	89.4 (1.7)	92.9	85.3	86.5	93.4
SS	87.4 (2.9)	88.2	86.4	86.7	88.3
ROOT	83.0 (2.4)	88.0	76.4	79.2	89.6
AT	78.1 (4.0)	75.9	82.9	81.0	73.3
ET	77.7 (2.4)	79.6	74.7	76.3	80.7
JR	76.4 (6.4)	69.0	97.7	96.0	55.0
AN	76.2 (2.5)	72.3	85.6	82.4	66.9
IQ	73.1 (2.1)	67.0	90.7	85.9	55.4
IK	72.5 (2.5)	75.0	67.9	70.6	77.1
OO	72.2 (5.3)	74.4	67.4	70.4	77.0
IR	72.1 (3.4)	64.7	97.9	95.6	46.3
DT	70.4 (1.4)	73.4	64.4	68.3	76.4
VG	70.0 (2.4)	81.1	52.1	64.8	87.9
PL	66.8 (2.7)	70.8	57.7	64.3	75.9
JC	64.8 (4.3)	59.1	97.4	92.4	32.1
CJ	64.0 (3.6)	62.2	71.7	66.6	56.3
HD	62.5 (2.7)	59.0	84.7	73.2	40.3
IC	61.3 (4.3)	56.8	97.1	90.8	25.4
OA	61.0 (3.4)	66.9	43.3	58.2	78.7
SP	60.7 (2.0)	67.4	42.4	57.9	79.0
I?	60.6 (1.3)	78.4	29.3	56.5	91.9
+A	60.1 (2.3)	58.6	68.9	62.4	51.4
TA	59.8 (2.5)	63.9	46.0	57.7	73.6
AG	59.7 (2.2)	57.0	81.6	68.4	37.9
NA	59.5 (3.5)	63.3	45.0	57.5	74.0
+F	59.0 (3.3)	64.4	40.4	56.6	77.6
UA	58.6 (3.9)	63.7	41.1	56.3	76.1
VA	58.2 (6.1)	56.2	85.3	67.1	31.1
MS	57.5 (1.8)	62.5	38.3	55.4	76.7
KA	57.5 (3.6)	75.6	35.4	47.3	79.6

Table 4: Performance of the Dependency type ratio attributes ordered by performance. Measures as described in Table 1 Continued in table 5.

3.2 Non-syntactic features

The second trial we performed was a search through the parameter space of all non-syntactic features. As our data contained 37 such parameters the size of the spaces were $\binom{37}{s}$ where s is the size of the model we were optimizing. For 3-parameter models this is no larger than $\binom{37}{3} = 7770$ while the maximum size is $\binom{37}{19} = 17672631900$. We searched for optimal subsets of sizes from 1 to 25. The best models are presented in Table 9 and the performance results in Table 10. Models larger than 8 are omitted as no significant performance improvement is measured beyond this point.

		LäSBarT		Other	
#	Accuracy	Prec.	Rec.	Prec.	Rec.
IT	56.5 (1.8)	54.1	86.7	66.6	26.3
PT	55.7 (2.9)	53.6	85.0	63.7	26.4
IS	55.6 (5.9)	53.1	99.9	85.0	11.3
JT	55.5 (3.8)	53.0	99.6	94.0	11.4
AA	55.4 (3.1)	57.4	42.1	54.3	68.7
IG	55.4 (2.8)	52.9	99.4	97.0	11.3
IU	55.1 (2.4)	82.4	26.1	45.6	84.0
RA	54.8 (2.5)	65.7	31.4	53.8	78.1
IO	54.4 (2.3)	63.6	33.4	45.5	75.4
MA	54.3 (3.3)	68.4	18.0	52.4	90.6
FS	53.8 (2.3)	72.9	12.0	52.1	95.6
CA	53.6 (3.9)	53.2	60.3	54.1	46.9
XX	53.0 (1.6)	69.4	24.7	44.5	81.3
ES	52.9 (1.7)	77.0	22.1	44.4	83.7
EF	52.4 (4.4)	52.4	75.4	41.4	29.4
++	52.3 (1.7)	51.3	93.6	65.0	11.0
XA	52.1 (1.7)	51.1	97.6	65.4	6.7
XT	52.1 (2.2)	51.2	97.0	50.9	7.3
EO	51.8 (2.4)	36.7	70.4	60.4	33.1
IF	51.2 (2.3)	55.4	39.7	48.1	62.7
FP	51.0 (1.3)	61.3	60.1	22.0	41.9
JG	51.0 (1.7)	29.1	57.0	48.6	45.0
DB	50.6 (0.9)	63.5	48.7	28.9	52.6
IV	50.5 (0.5)	75.0	44.0	28.8	57.0
OP	50.4 (0.9)	36.0	65.3	21.8	35.4
FO	50.2 (0.3)	57.1	29.0	35.8	71.4
VS	50.1 (0.4)	43.8	72.7	14.4	27.6
YY	50.0 (0.0)	50.0	100.0	0.0	0.0
XF	49.9 (0.2)	50.0	85.1	14.1	14.7
FV	49.8 (1.0)	55.6	57.9	21.3	41.7
VO	49.8 (3.3)	52.9	73.3	15.6	26.3

Table 5: Performance of the Dependency type ratio attributes ordered by performance. Measures as described in Table 1. Continued from table 4.

#	Set
2	VB, MAD
3	MAD, VB, MID
4	VB, PAD, MID, MAD
5	MAD, VB, MID, PAD, PM
6	MID, VB, HA, PAD, AB, MAD
7	PAD, JJ, PN, VB, MAD, KN, MID
8	PAD, HD, PM, MID, PN, VB, PL, MAD
9	PAD, SN, PLQS, MAD, DT, VB, RG, PM, MID
10	MAD, PM, PAD, KN, MID, PLQS, IE, VB, HA, DT

Table 6: Features in the best performing sets found for each size by the genetic search through the POS-ratio space.

4 Discussion

From the models using POS-ratio features, Tables 6 and 8, we see that it is possible to find models

#	Size
1 and 25	26
2 and 24	325
3 and 23	2 600
4 and 22	14 950
5 and 21	65 780
6 and 20	230 230
7 and 19	657 800
8 and 18	1 562 275
9 and 17	3 124 550
10 and 16	5 311 735
11 and 15	7 726 160
12 and 14	9 657 700
13	10 400 600

Table 7: Sizes of model space based on number of attributes in the target model.

		LäSBarT		Other	
Model	Accuracy	Prec.	Rec.	Prec.	Rec.
2	95.4 (1.5)	94.7	96.3	96.2	94.6
3	96.4 (0.9)	96.2	96.7	96.7	96.1
4	96.9 (1.0)	97.0	96.9	96.9	97.0
5	97.0 (1.1)	97.0	97.0	97.0	97.0
6	97.0 (1.2)	97.6	96.4	96.5	97.6
7	97.0 (1.1)	96.8	97.3	97.3	96.7
8	96.9 (1.1)	96.9	97.0	97.0	96.9
9	96.9 (1.3)	96.8	97.1	97.1	96.7
10	97.4 (1.1)	97.6	97.1	97.2	97.6
All(26)	96.8 (1.6)	96.9	96.7	96.7	96.9

Table 8: Performance of the feature sets selected from the set of POS-tag ratio features ordered by number of parameters. Measures as described in Table 1.

#	Set
2	OVIX, MAD
3	OVIX, MAD, MID
4	MID, PAD, MAD, OVIX
5	MAD, OVIX, VB, SN, SweVocT
6	MAD, HD, MID, PL, OVIX, SweVocC
7	MAD, AB, PP, HD, MID, OVIX, DT
8	MID, AB, PAD, OVIX, MAD, SweVocH, HS, RG

Table 9: Features in the best performing sets found for each size by the genetic search through the non-syntactic space.

that outperform most single feature models. We have in Table 8 included the performance of the full, 26 feature, model which shows that performance might be increased slightly by filtering out confusing features.

		LäSBarT		Other	
Model	Accuracy	Prec.	Rec.	Prec.	Rec.
2	96.6 (1.0)	95.5	98.0	98.0	95.3
3	97.4 (1.3)	97.3	97.4	97.5	97.3
4	98.2 (1.3)	97.8	98.7	98.7	97.7
5	97.9 (1.2)	97.1	98.9	98.8	97.0
6	98.0 (1.0)	97.2	98.9	98.8	97.1
7	97.8 (1.3)	97.1	98.6	98.6	97.0
8	98.5 (1.0)	97.9	99.1	99.1	97.9
All (37)	98.3 (1.0)	97.4	99.3	99.3	97.3

Table 10: Performance of the feature sets selected from the set of all non- syntactic features ordered by number of parameters. Measures as described in Table 1.

We can also see that the sets beyond 4 parameters do not fully correlate to the best performing single parameters in the parameter space. This implies that combinations of some features may be better predictors than the individual features.

When we search through all non-syntactic features we get results similar to the POS-ratio space search. While the first generated sets seem to consist of the best performing single parameters, larger models seem to be more "exotic" using low performing single parameters to create stronger combination effects, see Table 9.

The most interesting result here is that a model with 8 non-syntactic parameters, model 8 in Table 10, performs almost as well (-0.4 pp) as the 117 parameter total model, see Table 3.

Another interesting result is that the ratio of verbs (VB in Table 2) has an accuracy of 87.6%, only outperformed by the syntactic feature ADDD.

Even more interesting is the fact that the ratio of sentence terminating delimiters (MAD in Table 2) has such high performance. Especially as the average sentence length (ASL) is not a very good predictor of readability, see Table 3 and Falkenjack et al. (2013).

Theoretically, the ratio of MADs is the inverse of the ASL and as such their performance should align. However, the two metrics are calculated differently, sentence length is based on parsing data and MAD ratio is based on POS-tagging data. While a sentence should contain exactly one MAD there are instances where more than one (informal language, transcribed spoken language, misidentified ellipsis, quotations etc.) or less than one (bullet points, tables etc.) might occur in the actual text. It should be noted that if the aforementioned is true MAD might rather be a style predictor than a direct readability predictor. However, in that case style and readability appears to correlate which is not surprising.

We further note how much accuracy can be improved by combining very few measures. For instance, OVIX gives an accuracy of only 85.6% and MAD gives 87.1%, but combined they give 96.6%, set 2 in Table 10

5 Conclusion

In this paper we introduced and evaluated a method for finding optimal subsets of text features for readability based document classification. The method uses genetic search to systematically generate models using various sets of text features. As fitness function for the genetic algorithm we used SVM created models that were 7-fold cross validated on one easy-to-read corpus and one corpus of regular texts.

Our results show that, at least for Swedish, it is possible to find models almost as good the currently best models while omitting parsing based features. Our algorithm found a model of 8 non-syntactic parameters which predicted readability with an accuracy of 98.5%. This is almost as accurate as a 117 parameter model, including parsing based features, with an accuracy of 98.9%

Our study was conducted for Swedish texts but only a few of the metrics used are specific to Swedish and the optimization method itself is language independent, thus, the method can easily be applied to other languages. The method can be used for optimization of readability assessment systems as well as for basic linguistic research into readability.

Acknowledgments

We would like to thank Stiftelsen Marcus och Amalia Wallenbergs Minnesfond and SICS East Swedish ICT AB for funding this research as well as the staff members at Språkbanken who created and let us use the Korp corpus import tool.

References

Sandra Alusio, Lucia Specia, Caroline Gasperin, and Carolina Scarton. 2010. Readability assessment for text simplification. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–9.

Lars Borin, Markus Forsberg, and Johan Roxendal. 2012. Korp – the corpus infrastructure of Språkbanken. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*.

Jeanne S. Chall and Edgar Dale. 1995. *Readability revisited: The new Dale–Chall readability formula*. Brookline Books, Cambride, MA.

Kevyn Collins-Thompson and Jamie Callan. 2004. A Language Modeling Approach to Predicting Reading Difficulty. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

Edgar Dale and Jeanne S. Chall. 1949. The concept of readability. *Elementary English*, 26(23).

Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2011. READ-IT: Assessing Readability of Italian Texts with a View to Text Simplification. In *Proceedings of the 2nd Workshop on Speech and Language Processing for Assistive Technologies*, pages 73–83, July.

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

Johan Falkenjack, Katarina Heimann Mühlenbock, and Arne Jönsson. 2013. Features indicating readability in Swedish text. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NoDaLiDa-2013), Oslo, Norway*, NEALT Proceedings Series 16.

Lijun Feng. 2010. *Automatic Readability Assessment*. Ph.D. thesis, City University of New York.

Michael J. Heilman, Kevyn Collins-Thompson, Jamie Callan, and Maxine Eskenazi. 2007. Combining Lexical and Grammatical Features to Improve Readability Measures for First and Second Language Texts. In *Proceedings of NAACL HLT 2007*, pages 460–467.

Michael J. Heilman, Kevyn Collins-Thompson, and Maxine Eskenazi. 2008. An Analysis of Statistical Models and Features for Reading Difficulty Prediction. In *Proceedings of the Third ACL Workshop on Innovative Use of NLP for Building Educational Applications*, pages 71–79, June.

Katarina Heimann Mühlenbock. 2013. *I see what you mean. Assessing readability for specific target groups*. Dissertation, Språkbanken, Dept of Swedish, University of Gothenburg.

J. P. Kincaid, R. P. Fishburne, R. L. Rogers, and B. S. Chissom. 1975. Derivation of new readability formulas (Automated Readability Index, Fog Count, and Flesch Reading Ease Formula) for Navy enlisted personnel. Technical report, U.S. Naval Air Station, Millington, TN.

Tak Pang Lau. 2006. Chinese readability analysis and its applications on the internet. Master's thesis, The Chinese University of Hong Kong.

Haitao Liu. 2008. Dependency distance as a metric of language comprehension difficulty. *Journal of Cognitive Science*, 9(2):169–191.

Katarina Mühlenbock. 2008. Readable, Legible or Plain Words – Presentation of an easy-to-read Swedish corpus. In Anju Saxena and Åke Viberg, editors, *Multilingualism: Proceedings of the 23rd Scandinavian Conference of Linguistics*, volume 8 of *Acta Universitatis Upsaliensis*, pages 327–329, Uppsala, Sweden. Acta Universitatis Upsaliensis.

Ani Nenkova, Jieun Chae, Annie Louis, and Emily Pitler. 2010. Structural Features for Predicting the Linguistic Quality of Text Applications to Machine Translation, Automatic Summarization and Human–Authored Text. In E. Krahmer and M. Theune, editors, *Empirical Methods in NLG*, pages 222–241. Springer-Verlag.

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 Evaluation (LREC2006)*, pages 2216–2219, May.

Sarah Petersen and Mari Ostendorf. 2009. A machine learning approach toreading level assessment. *Computer Speech and Language*, 23:89–106.

Sarah Petersen. 2007. *Natural language processing tools for reading level assessment and text simplification for bilingual education*. Ph.D. thesis, University of Washington, Seattle, WA.

Emily Pitler and Ani Nenkova. 2008. Revisiting Readability: A Unified Framework for Predicting Text Quality. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 186–195, Honolulu, HI, October.

John C. Platt. 1998. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. Technical Report MSR-TR-98-14, Microsoft Research, April.

Sarah E. Schwarm and Mari Ostendorf. 2005. Reading level assessment using support vector machines and statistical language models. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*.

Johan Sjöholm. 2012. Probability as readability: A new machine learning approach to readability assessment for written Swedish. Master's thesis, Linköping University.