A Comparison of Techniques to Automatically Identify Complex Words

Matthew Shardlow
School of Computer Science, University of Manchester
IT301, Kilburn Building, Manchester, M13 9PL, England
m.shardlow@cs.man.ac.uk

Abstract

Identifying complex words (CWs) is an important, yet often overlooked, task within lexical simplification (The process of automatically replacing CWs with simpler alternatives). If too many words are identified then substitutions may be made erroneously, leading to a loss of meaning. If too few words are identified then those which impede a user's understanding may be missed, resulting in a complex final text. This paper addresses the task of evaluating different methods for CW identification. A corpus of sentences with annotated CWs is mined from Simple Wikipedia edit histories, which is then used as the basis for several experiments.

Firstly, the corpus design is explained and the results of the validation experiments using human judges are reported. Experiments are carried out into the CW identification techniques of: simplifying everything, frequency thresholding and training a support vector machine. These are based upon previous approaches to the task and show that thresholding does not perform significantly differently to the more naïve technique of simplifying everything. The support vector machine achieves a slight increase in precision over the other two methods, but at the cost of a dramatic trade off in recall.

1 Introduction

Complex Word (CW) identification is an important task at the first stage of lexical simplification and errors introduced or avoided here will affect final results. This work looks at the process of automatically identifying difficult words for a lexical simplification system. Lexical simplification is the task of identifying and replacing CWs in a text to improve the overall understandability and readability. This is a difficult task which is computationally expensive and often inadequately accurate.

Lexical simplification is just one method of text simplification and is often deployed alongside other simplification methods (Carrol et al., 1998; Aluísio and Gasperin, 2010). Syntactic simplification, statistical machine translation and semantic simplification (or explanation generation) are all current methods of text simplification. Text simplification is typically deployed as an assistive technology (Devlin and Tait, 1998; Aluísio and Gasperin, 2010), although this is not always the case. It may also be used alongside other technologies such as summarisation to improve their final results.

Identifying CWs is a task which every lexical simplification system must perform, either explicitly or implicitly, before simplification can take place. CWs are difficult to define, which makes them difficult to identify. For example, take the following sentence:

> The four largest islands are Honshu, Hokkaido, Shikoku, and Kyushu, and there are approximately 3,000 smaller islands in the chain.

In the above sentence, we might identify the proper nouns (Honshu, Hokkaido, etc.) as complex (as they may be unfamiliar) or we may choose to discount them from our scheme altogether, as proper nouns are unlikely to have any valid replacements. If we discount the proper nouns then the other valid CW would be 'approximately'. At 13 characters it is more than twice the average of 5.7 characters per word and has more syllables than any other word. Further, CWs are often identified by their frequency (see Section 2.1) and here,

Proceedings of the ACL Student Research Workshop, pages 103–109,
Sofia, Bulgaria, August 4-9 2013.

'approximately' exhibits a much lower frequency than the other words.

There are many reasons to evaluate the identification of CWs. This research stems primarily from the discovery that no previous comparison of current techniques exists. It is hoped that by providing this, the community will be able to identify and evaluate new techniques using the methods proposed herein. If CW identification is not performed well, then potential candidates may be missed, and simple words may be falsely identified. This is dangerous as simplification will often result in a minor change in a text's semantics. For example, the sentence:

> The United Kingdom is a *state* in northwest Europe.

May be simplified to give:

> The United Kingdom is a *country* in northwest Europe.

In this example from the corpus used in this research, the word "state" is simplified to give "country". Whilst this is a valid synonym in the given context, state and country are not necessarily semantically identical. Broadly speaking, state refers to a political entity, whereas country refers to a physical space within a set of borders. This is an acceptable change and even necessary for simplification. However, if applied blindly, then too many modifications may be made, resulting in major deviations from the text's original semantics.

The contributions of this paper are as follows:

- A report on the corpus developed and used in the evaluation phase. Section 2.2.
- The implementation of a support vector machine for the classification of CWs. Section 2.6
- A comparison of common techniques on the same corpus. Section 4.
- An analysis of the features used in the support vector machine. Section 4.

2 Experimental Design

Several systems for detecting CWs were implemented and evaluated using the CW corpus. The two main techniques that exist in the literature are simplifying everything (Devlin and Tait, 1998) and frequency based thresholding (Zeng et al., 2005). These were implemented as well as a support vector machine classifier. This section describes the design decisions made during implementation.

System	Score
SUBTLEX	0.3352
Wikipedia Baseline	0.3270
Kucera-Francis	0.3097
Random Baseline	0.0157

Table 1: The results of different experiments on the SemEval lexical simplification data. These show that SUBTLEX was the best word frequency measure for rating lexical complexity. The other entries correspond to alternative word frequency measures. The Google Web 1T data (Brants and Franz, 2006) has been shown to give a higher score, however this data was not available during the course of this research.

2.1 Lexical Complexity

All three of the implementations described in Sections 2.4, 2.5 and 2.6 require a word frequency measure as an indicator of lexical complexity. If a word occurs frequently in common language then it is more likely to be recognised (Rayner and Duffy, 1986).

The lexical simplification dataset from Task 1 at SemEval 2012 (De Belder and Moens, 2012) was used to compare several measures of word frequency as shown in Table 1. Candidate substitutions and sample sentences were provided by the task organisers, together with a gold standard ranking of the substitutes according to their simplicity. These sentences were ranked according to their frequency. Although the scores in Table 1 appear to be low, this is the kappa agreement for several categories and so should be expected. The inter-annotator agreement on the corpus was 0.488 (De Belder and Moens, 2012). The SUBTLEX dataset (Brysbaert and New, 2009) was the best available for rating word familiarity. This is a corpus of over 70,000 words collected from the subtitles of over 8,000 American English films.

2.2 CW Corpus

Simple Wikipedia edit histories were mined using techniques similar to those in Yatskar et al. (2010). This provided aligned pairs of sentences which had just one word simplified. Whereas Yatskar et al. (2010) used these pairs to learn probabilities of paraphrases, the research in this paper used them as instances of lexical simplification. The original simplifications were performed by editors trying to make documents as simple as possible. The CW is identified by comparison with the simplified sentence. Further information on the production of the corpus will be published in a future paper.

2.3 Negative Examples

The CW corpus provides a set of CWs in appropriate contexts. This is useful for evaluation as these words need to be identified. However, if only examples of CWs were available, it would be very easy for a technique to overfit — as it could just classify every single word as complex and get 100% accuracy. For example, in the case of thresholding, if only examples of CWs are available, the threshold could be set artificially high and still succeed for every case. When this is applied to genuine data it will classify every word it encounters as complex, leading to high recall but low precision.

To alleviate this effect, negative examples are needed. These are examples of simple words which do not require any further simplification. There are several methods for finding these, including: selecting words from a reference easy word list; selecting words with high frequencies according to some corpus or using the simplified words from the second sentences in the CW corpus. The chosen strategy picked a word at random from the sentence in which the CW occurs. Only one word was edited in this sentence and so the assumption may be made that none of the other words in the sentence require further simplification. Only one simple word per CW is chosen to enforce an even amount of positive and negative data. This gave a set of negative words which were reflective of the broad language which is expected when processing free text.

2.4 Simplify Everything

The first implementation involved simplifying everything, a brute force method, in which a simplification algorithm is applied to every word. This assumes that words which are already simple will not require any further simplification. A common variation is to limit the simplification to some combination of all the nouns, verbs and adjectives.

A standard baseline lexical simplification system was implemented following Devlin and Tait (1998). This algorithm generated a set of synonyms from WordNet and then used the SUBTLEX frequencies to find the most frequent synonym. If the synonym was more frequent than the original word then a substitution was made. This technique was applied to all the words. If a CW was changed, then it was considered a true positive; if a simple word was not changed, it was considered a true negative. Five trials were carried out and the average accuracy and standard deviation is reported in Figure 1 and Table 3.

2.5 Frequency Thresholding

The second technique is frequency thresholding. This relies on each word having an associated familiarity value provided by the SUBTLEX corpus. Whilst this corpus is large, it will never cover every possible word, and so words which are not encountered are considered to have a frequency of 0. This does not affect comparison as the infrequent words are likely to be the complex ones.

To distinguish between complex and simple words a threshold was implemented. This was learnt from the CW corpus by examining every possible threshold for a training set. Firstly, the training data was ordered by frequency, then the accuracy[1] of the algorithm was examined with the threshold placed in between the frequency of every adjacent pair of words in the ordered list. This was repeated by 5-fold cross validation and the mean threshold determined. The final accuracy of the algorithm was then determined on a separate set of testing data.

2.6 Support Vector Machine

Support vector machines (SVM) are statistical classifiers which use labelled training data to predict the class of unseen inputs. The training data consist of several features which the SVM uses to distinguish between classes. The SVM was chosen as it has been used elsewhere for similar tasks (Gasperin et al., 2009; Hancke et al., 2012; Jauhar and Specia, 2012). The use of many fea-

[1]The proportion of data that was correctly classified.

tures allows factors which may otherwise have been missed to be taken into account. One further advantage is that the features of an SVM can be analysed to determine their effect on the classification. This may give some indication for future feature classification schemes.

The SVM was trained using the LIBSVM package (Chang and Lin, 2011) in Matlab. the RBF kernel was selected and a grid search was performed to select values for the 2 parameters C and γ. Training and testing was performed on a held-out data-set using 5-fold cross validation.

To implement the SVM a set of features was determined for the classification scheme. Several external libraries were used to extract these as detailed below:

Frequency The SUBTLEX frequency of each word was used as previously described in Section 2.1.

CD Count Also from the SUBTLEX corpus. The number of films in which a word appeared, ranging from $0 - 8,388$.

Length The word length in number of characters was taken into account. It is often the case that longer words are more difficult to process and so may be considered 'complex'.

Syllable Count The number of syllables contained in a word is also a good estimate of its complexity. This was computed using a library from the morphadorner package[2].

Sense Count A count of the number of ways in which a word can be interpreted - showing how ambiguous a word is. This measure is taken from WordNet (Fellbaum, 1998).

Synonym Count Also taken from WordNet, this is the number of potential synonyms with which a word could be replaced. This again may give some indication of a word's degree of ambiguity.

3 Results

The results of the experiments in identifying CWs are shown in Figure 1 and the values are given in Table 3. The values presented are the mean of 5 trials and the error bars represent the standard deviation.

[2] http://morphadorner.northwestern.edu/

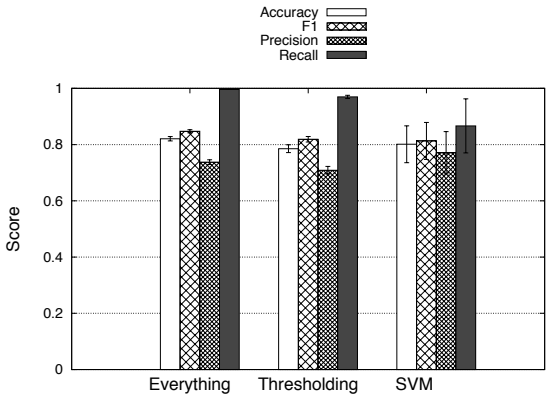

Figure 1: A bar chart with error bars showing the results of the CW identification experiments. Accuracy, F1 Score, Precision and Recall are reported for each measure.

Feature	Coefficient
Frequency	0.3973
CD Count	0.5847
Length	−0.5661
Syllables	−0.4414
Senses	−0.0859
Synonyms	−0.2882

Table 2: The correlation coefficients for each feature. These show the correlation against the language's simplicity and so a positive correlation indicates that if that feature is higher then the word will be simpler.

To analyse the features of the SVM, the correlation coefficient between each feature vector and the vector of feature labels was calculated. This is a measure which can be used to show the relation between two distributions. The adopted labelling scheme assigned CWs as 0 and simple words as 1 and so the correlation of the features is notionally against the simplicity of the words.[3] The results are reported in Table 2.

4 Discussion

It is clear from these results that there is a fairly high accuracy from all the methods. This shows that they perform well at the task in hand, reflecting the methods which have been previously applied. These methods all have a higher recall than

[3] i.e. A positive correlation indicates that if the value of that feature is higher, the word will be simpler.

System	Accuracy	F1	Precision	Recall
Simplify Everything	0.8207 ± 0.0077	0.8474 ± 0.0056	0.7375 ± 0.0084	0.9960 ± 0
Thresholding	0.7854 ± 0.0138	0.8189 ± 0.0098	0.7088 ± 0.0136	0.9697 ± 0.0056
SVM	0.8012 ± 0.0656	0.8130 ± 0.0658	0.7709 ± 0.0752	0.8665 ± 0.0961

Table 3: The results of classification experiments for the three systems.

precision, which indicates that they are good at identifying the CWs, but also that they often identify simple words as CWs. This is particularly noticeable in the 'simplify everything' method, where the recall is very high, yet the precision is comparatively low. This indicates that many of the simple words which are falsely identified as complex are also replaced with an alternate substitution, which may result in a change in sense.

A paired t-test showed the difference between the thresholding method and the 'simplify everything' method was not statistically significant ($p > 0.8$). Thresholding takes more data about the words into account and would appear to be a less naïve strategy than blindly simplifying everything. However, this data shows there is little difference between the results of the two methods. The thresholding here may be limited by the resources, and a corpus using a larger word count may yield an improved result.

Whilst the thresholding and simplify everything methods were not significantly different from each other, the SVM method was significantly different from the other two ($p < 0.001$). This can be seen in the slightly lower recall, yet higher precision attained by the SVM. This indicates that the SVM was better at distinguishing between complex and simple words, but also wrongly identified many CWs. The results for the SVM have a wide standard deviation (shown in the wide error bars in Figure 1) indicating a higher variability than the other methods. With more data for training the model, this variability may be reduced.

One important factor in the increased precision observed in the SVM is that it used many more features than the other methods, and so took more information into account. Table 2 shows that these features had varying degrees of correlation with the data label (i.e. whether the word was simple or not) and hence that they had varying degrees of effect on the classification scheme.

Frequency and CD count are moderately positively correlated as may be expected. This indicates that higher frequency words are likely to be simple. Surprisingly, CD Count has a higher correlation than frequency itself, indicating that this is a better measure of word familiarity than the frequency measure. However, further investigation is necessary to confirm this.

Word length and number of syllables are moderately negatively correlated, indicating that the longer and more polysyllabic a word is, the less simple it becomes. This is not true in every case. For example, 'finger' and 'digit' can be used in the same sense (as a noun meaning an appendage of the hand). Whilst 'finger' is more commonly used than 'digit'[4], digit is one letter shorter.

The number of senses was very weakly negatively correlated with word simplicity. This indicates that it is not a strong indicative factor in determining whether a word is simple or not. The total number of synonyms was a stronger indicator than the number of senses, but still only exhibited weak correlation.

One area that has not been explored in this study is the use of contextual features. Each target word occurs in a sentence and it may be the case that those words surrounding the target give extra information as to its complexity. It has been suggested that language is produced at an even level of complexity (Specia et al., 2012), and so simple words will occur in the presence of other simple words, whereas CWs will occur in the presence of other CWs. As well as lexical contextual information, the surrounding syntax may offer some information on word difficulty. Factors such as a very long sentence or a complex grammatical structure can make a word more difficult to understand. These could be used to modify the familiarity score in the thresholding method, or they could be used as features in the SVM classifier.

5 Related Work

This research will be used for lexical simplification. The related work in this field is also generally

[4]in the SUBTLEX corpus 'finger' has a frequency of 1870, whereas 'digit' has a frequency of 30.

used as a precursor to lexical simplification. This section will explain how these previous methods have handled the task of identifying CWs and how these fit into the research presented in this paper.

The simplest way to identify CWs in a sentence is to blindly assume that every word is complex, as described earlier in Section 2.4. This was first used in Devlin's seminal work on lexical simplification (Devlin and Tait, 1998). This method is somewhat naïve as it does not mitigate the possibility of words being simplified in error. Devlin and Tait indicate that they believe less frequent words will not be subject to meaning change. However, further work into lexical simplification has refuted this (Lal and Rüger, 2002). This method is still used, for example Thomas and Anderson (2012) simplify all nouns and verbs. This corresponds to the 'Everything' method.

Another method of identifying CWs is to use frequency based thresholding over word familiarity scores, as described in Section 2.5 and corresponding to the 'Frequency' method in this paper. This has been applied to the medical domain (Zeng et al., 2005; Elhadad, 2006) for predicting which words lay readers will find difficult. This has been correlated with word difficulty via questionnaires (Zeng et al., 2005; Zeng-Treitler et al., 2008) and via the analysis of low-level readability corpora (Elhadad, 2006). In both these cases, a familiarity score is used to determine how likely a subject is to understand a term. More recently, Bott et al. (2012) use a threshold of 1% corpus frequency, along with other checks, to ensure that simple words are not erroneously simplified.

Support vector machines are powerful statistical classifiers, as employed in the 'SVM' method of this paper. A Support Vector Machine is used to predict the familiarity of CWs in Zeng et al. (2005). It takes features of term frequency and word length and is correlated against the familiarity scores which are already obtained. This proves to have very poor performance, something which the authors attribute to a lack of suitable training data. An SVM has also been trained for the ranking of words according to their complexity (Jauhar and Specia, 2012). This was done for the SemEval lexical simplification task (Specia et al., 2012). Although this system is designed for synonym ranking, it could also be used for the CW identification task. Machine learning has also been applied to the task of determining whether an entire sentence requires simplification (Gasperin et al., 2009; Hancke et al., 2012). These approaches use a wide array of morphological features which are suited to sentence level classification.

6 Future Work

This work is intended as an initial study of methods for identifying CWs for simplification. The methods compared, whilst typical of current CW identification methods, are not an exhaustive set and variations exist. One further way of expanding this research would be to take into account word context. This could be done using thresholding (Zeng-Treitler et al., 2008) or an SVM (Gasperin et al., 2009; Jauhar and Specia, 2012).

Another way to increase the accuracy of the frequency count method may be to use a larger corpus. Whilst the corpus used in this paper performed well in the preliminary testing section, other research has shown the Google Web1T corpus (a n-gram count of over a trillion words) to be more effective (De Belder and Moens, 2012). The Web 1T data was not available during the course of this research.

The large variability in accuracy shown in the SVM method indicates that there was insufficient training data. With more data, the SVM would have more information about the classification task and would provide more consistent results.

CW identification is the first step in the process of lexical simplification. This research will be integrated in a future system which will simplify natural language for end users. It is also hoped that other lexical simplification systems will take account of this work and will use the evaluation technique proposed herein to improve their identification of CWs.

7 Conclusion

This paper has provided an insight into the challenges associated with evaluating the identification of CWs. This is a non-obvious task, which may seem intuitively easy, but in reality is quite difficult and rarely performed. It is hoped that new research in this field will evaluate the techniques used, rather than using inadequate techniques blindly and naïvely. This research has also shown that the current state of the art methods have much room for improvement. Low precision is a constant factor in all techniques and future research should aim to address this.

Acknowledgment

This work is supported by EPSRC grant EP/I028099/1. Thanks go to the anonymous reviewers for their helpful suggestions.

References

Sandra Maria Aluísio and Caroline Gasperin. 2010. Fostering digital inclusion and accessibility: the PorSimples project for simplification of Portuguese texts. In *Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas*, YIWCALA '10, pages 46–53, Stroudsburg, PA, USA. Association for Computational Linguistics.

Stefan Bott, Luz Rello, Biljana Drndarevix, and Horacio Saggion. 2012. Can spanish be simpler? lexsis: Lexical simplification for spanish. In *Coling 2012: The 24th International Conference on Computational Linguistics*.

Thorsten Brants and Alex Franz. 2006. Web 1T 5-gram Version 1.

Marc Brysbaert and Boris New. 2009. Moving beyond Kucera and Francis : a critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English. *Behav Res Methods*.

John Carrol, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. 1998. Practical simplification of english newspaper text to assist aphasic readers. In *AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Jan De Belder and Marie-Francine Moens. 2012. A dataset for the evaluation of lexical simplification. In *Computational Linguistics and Intelligent Text Processing*, volume 7182 of *Lecture Notes in Computer Science*, pages 426–437. Springer Berlin / Heidelberg.

Siobhan Devlin and John Tait. 1998. The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases*, pages 161–173.

Noémie Elhadad. 2006. Comprehending technical texts: Predicting and defining unfamiliar terms. In *AMIA Annual Symposium proceedings*, volume 2006, page 239. American Medical Informatics Association.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Caroline Gasperin, Lucia Specia, Tiago Pereira, and Sandra M. Aluísio. 2009. Learning when to simplify sentences for natural text simplification. In *Encontro Nacional de Inteligência Artificial*.

Julia Hancke, Sowmya Vajjala, and Detmar Meurers. 2012. Readability classification for German using lexical, syntactic, and morphological features. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 1063–1080, Mumbai, India.

Sujay Kumar Jauhar and Lucia Specia. 2012. Uowshef: Simplex – lexical simplicity ranking based on contextual and psycholinguistic features. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 477–481, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Partha Lal and Stefan Rüger. 2002. Extract-based summarization with simplification. In *Proceedings of the ACL*.

Keith Rayner and Susan Duffy. 1986. Lexical complexity and fixation times in reading: Effects of word frequency, verb complexity, and lexical ambiguity. *Memory & Cognition*, 14:191–201.

Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. 2012. Semeval-2012 task 1: English lexical simplification. In *First Joint Conference on Lexical and Computational Semantics*.

S. Rebecca Thomas and Sven Anderson. 2012. Wordnet-based lexical simplification of a document. In Jeremy Jancsary, editor, *Proceedings of KONVENS 2012*, pages 80–88. ÖGAI, September. Main track: oral presentations.

Mark Yatskar, Bo Pang, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2010. For the sake of simplicity: Unsupervised extraction of lexical simplifications from Wikipedia. In *Proceedings of the NAACL*.

Qing Zeng, Eunjung Kim, Jon Crowell, and Tony Tse. 2005. A text corpora-based estimation of the familiarity of health terminology. *Biological and Medical Data Analysis*, pages 184–192.

Qing Zeng-Treitler, Sergey Goryachev, Tony Tse, Alla Keselman, and Aziz Boxwala. 2008. Estimating consumer familiarity with health terminology: a context-based approach. *Journal of the American Medical Informatics Association*, 15(3):349–356.