The CW Corpus: A New Resource for Evaluating the Identification of Complex Words

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

Abstract

The task of identifying complex words (CWs) is important for lexical simplification, however it is often carried out with no evaluation of success. There is no basis for comparison of current techniques and, prior to this work, there has been no standard corpus or evaluation technique for the CW identification task. This paper addresses these shortcomings with a new corpus for evaluating a system's performance in identifying CWs. Simple Wikipedia edit histories were mined for instances of single word lexical simplifications. The corpus contains 731 sentences, each with one annotated CW. This paper describes the method used to produce the CW corpus and presents the results of evaluation, showing its validity.

1 Introduction

CW identification techniques are typically implemented as a preliminary step in a lexical simplification system. The evaluation of the identification of CWs is an often forgotten task. Omitting this can cause a loss of accuracy at this stage which will adversely affect the following processes and hence the user's understanding of the resulting text.

Previous approaches to the CW identification task (see Section 5) have generally omitted an evaluation of their method. This gap in the literature highlights the need for evaluation, for which gold standard data is needed. This research proposes the CW corpus, a dataset of 731 examples of sentences with exactly one annotated CW per sentence.

A CW is defined as one which causes a sentence to be more difficult for a user to read. For example, in the following sentence:

'The cat reposed on the mat'

The presence of the word 'reposed' would reduce the understandability for some readers. It would be difficult for some readers to work out the sentence's meaning, and if the reader is unfamiliar with the word 'reposed', they will have to infer its meaning from the surrounding context. Replacing this word with a more familiar alternative, such as 'sat', improves the understandability of the sentence, whilst retaining the majority of the original semantics.

Retention of meaning is an important factor during lexical simplification. If the word 'reposed' is changed to 'sat', then the specific meaning of the sentence will be modified (generally speaking, reposed may indicate a state of relaxation, whereas sat indicates a body position) although the broad meaning is still the same (a cat is on a mat in both scenarios). Semantic shift should be kept to a minimum during lexical simplification. Recent work (Biran et al., 2011; Bott et al., 2012) has employed distributional semantics to ensure simplifications are of sufficient semantic similarity.

Word complexity is affected by many factors such as familiarity, context, morphology and length. Furthermore, these factors change from person to person and context to context. The same word, in a different sentence, may be perceived as being of a different level of difficulty. The same word in the same sentence, but read by a different person, may also be perceived as different in difficulty. For example, a person who speaks English as a second language will struggle with unfamiliar words depending on their native tongue. Conversely, the reader who has a low reading ability will struggle with long and obscure words. Whilst there will be some crossover in the language

Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations, pages 69–77,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics

these two groups find difficult, this will not be exactly the same. This subjectivity makes the automation and evaluation of CW identification difficult.

Subjectivity makes the task of natural language generation difficult and rules out automatically generating annotated complex sentences. Instead, our CW discovery process (presented in Section 2) mines simplifications from Simple Wikipedia[1] edit histories. Simple Wikipedia is well suited to this task as it is a website where language is collaboratively and iteratively simplified by a team of editors. These editors follow a set of strict guidelines and accountability is enforced by the self policing community. Simple Wikipedia is aimed at readers with a low English reading ability such as children or people with English as a second language. The type of simplifications found in Wikipedia and thus mined for use in our corpus are therefore appropriate for people with low English proficiency. By capturing these simplifications, we produce a set of genuine examples of sentences which can be used to evaluate the performance of CW identification systems. It should be noted that although these simplifications are best suited to low English proficiency users, the CW identification techniques that will be evaluated using the corpus can be trained and applied for a variety of user groups.

The contributions of this paper are as follows:

- A description of the method used to create the CW corpus. Section 2.
- An analysis of the corpus combining results from 6 human annotators. Section 3.
- A discussion on the practicalities surrounding the use of the CW corpus for the evaluation of a CW identification system. Section 4.

Related and future work are also presented in Sections 5 and 6 respectively.

2 Design

Our corpus contains examples of simplifications which have been made by human editors during their revisions of Simple Wikipedia articles. These are in the form of sentences with one word which has been identified as requiring simplification.[2] These examples can be used to evaluate the output of a CW identification system (see Section 6). To make the discovery and evaluation task easier, we limit the discovered simplifications to one word per sentence. So, if an edited sentence differs from its original by more than one word, we do not include it in our corpus. This also promotes uniformity in the corpus, reducing the complexity of the evaluation task.

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

Table 1: The results of different experiments on the SemEval lexical simplification data (de Belder and Moens, 2012), showing the SUBTLEX data's superior performance over several baselines. Each baseline gave a familiarity value to a set of words based on their frequency of occurrence. These values were used to produce a ranking over the data which was compared with a gold standard ranking using kappa agreement to give the scores shown here. A baseline using the Google Web 1T dataset was shown to give a higher score than SUBTLEX, however this dataset was not available during the course of this research.

2.1 Preliminaries

SUBTLEX

The SUBTLEX dataset (Brysbaert and New, 2009) is used as a familiarity dictionary. Its primary function is to associate words with their frequencies of occurrence, assuming that words which occur more frequently are simpler. SUBTLEX is also used as a dictionary for testing word existence: if a word does not occur in the dataset, it is not considered for simplification. This may occur in the case of very infrequent words or proper nouns. The

[1] http://simple.wikipedia.org/

[2] We also record the simplification suggested by the original Simple Wikipedia editor.

SUBTLEX data is chosen over the more conventional Kučera-Francis frequency (Kučera and Francis, 1967) and over a baseline produced from Wikipedia frequencies due to a previous experiment using a lexical simplification dataset from task 1 of SemEval 2012 (de Belder and Moens, 2012). See Table 1.

Word Sense

Homonymy is the phenomenon of a wordform having 2 distinct meanings as in the classic case: '*Bank* of England' vs. 'River *bank*'. In each case, the word *bank* is referring to a different semantic entity. This presents a problem when calculating word frequency as the frequencies for homonyms will be combined. Word sense disambiguation is an unsolved problem and was not addressed whilst creating the CW corpus. The role of word sense in lexical simplification will be investigated at a later stage of this research.

Yatskar et al. (2010)

The CW corpus was built following the work of Yatskar et al. (2010) in identifying paraphrases from Simple Wikipedia edit histories. Their method extracts lexical edits from aligned sentences in adjacent revisions of a Simple Wikipedia article. These lexical edits are then processed to determine their likelihood of being a true simplification. Two methods for determining this probability are presented, the first uses conditional probability to determine whether a lexical edit represents a simplification and the second uses metadata from comments to generate a set of trusted revisions, from which simplifications can be detected using pointwise mutual information. Our method (further explained in Section 2.2) differs from their work in several ways. Firstly, we seek to discover only single word lexical edits. Secondly, we use both article metadata and a series of strict checks against a lexicon, a thesaurus and a simplification dictionary to ensure that the extracted lexical edits are true simplifications. Thirdly, we retain the original context of the simplification as lexical complexity is thought to be influenced by context (Biran et al., 2011; Bott et al., 2012).

Automatically mining edit histories was chosen as it provides many instances quickly and at a low cost. The other method of creating a similar corpus would have been to ask several professionally trained annotators to produce hundreds of sets of sentences, and to mark up the CWs in these. The use of professionals would be expensive and annotators may not agree on the way in which words should be simplified, leading to further problems when combining annotations.

2.2 Method

In this section, we explain the procedure to create the corpus. There are many processing stages as represented graphically in Figure 1. The stages in the diagram are further described in the sections below. For simplicity, we view Simple Wikipedia as a set of pages P, each with an associated set of revisions R. Every revision of every page is processed iteratively until P is exhausted.

Content Articles

The Simple Wikipedia edit histories were obtained.[3] The entire database was very large, so only main content articles were considered. All user, talk and meta articles were discarded. Non-content articles are not intended to be read by typical users and so may not reflect the same level of simplicity as the rest of the site.

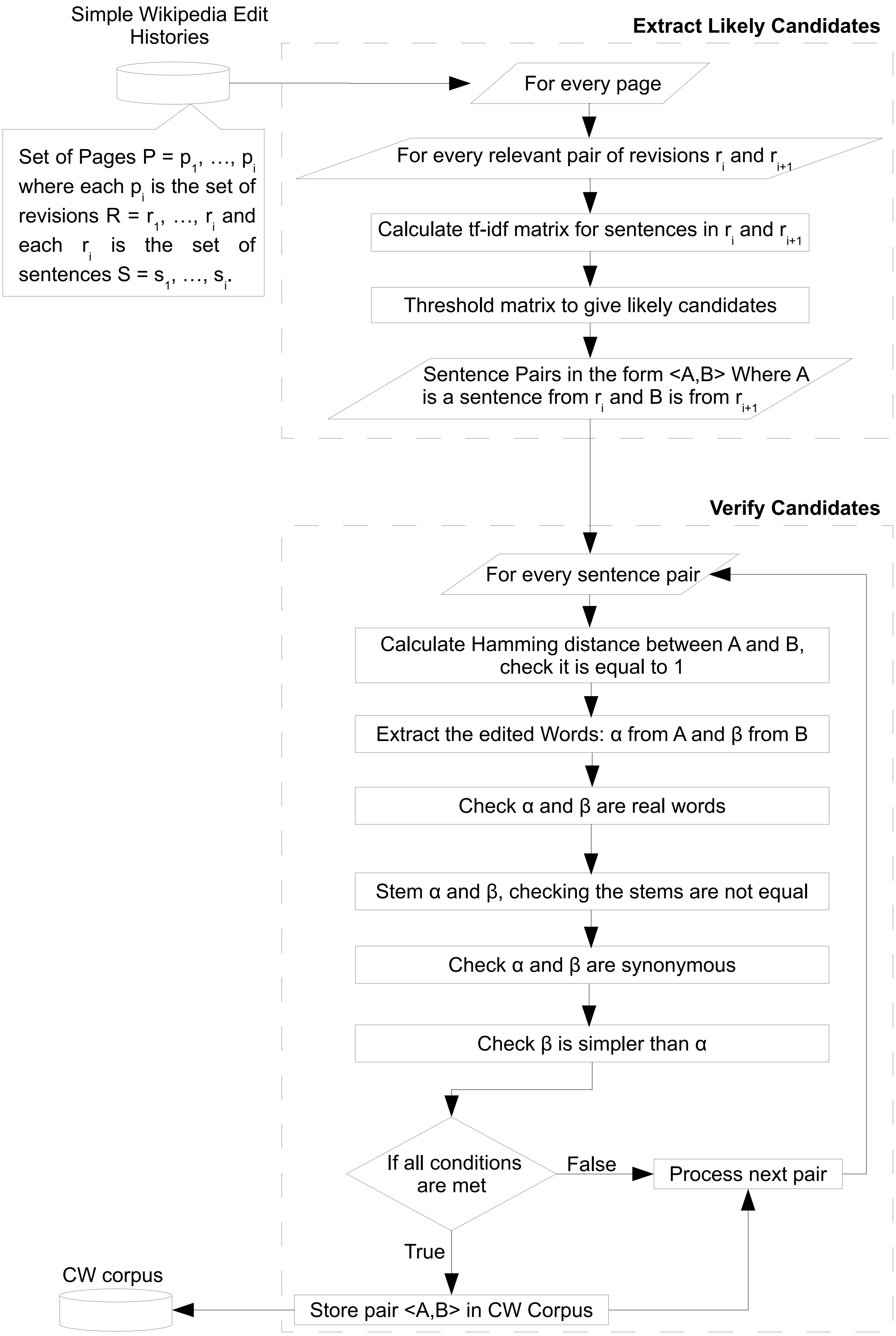
Revisions which Simplify

When editing a Simple Wikipedia article, the author has the option to attach a comment to their revision. Following the work of Yatskar et al. (2010), we only consider those revisions which have a comment containing some morphological equivalent of the lemma 'simple', e.g. simplify, simplifies, simplification, simpler, etc. This allows us to search for comments where the author states that they are simplifying the article.

Tf-idf Matrix

Each revision is a set of sentences. As changes from revision to revision are often small, there will be many sentences which are the same in adjacent revisions. Sentences which are likely to contain a simplification will only have one word difference and sentences which are unrelated will have many different words. Tf-idf (Salton and Yang, 1973) vectors are calculated

[3] Database dump dated 4th February 2012.



Simple Wikipedia Edit Histories
Extract Likely Candidates
For every page
Set of Pages P = $p_1, \dots, p_i$ where each p_i is the set of revisions R = $r_1, \dots, r_i$ and each r_i is the set of sentences S = $s_1, \dots, s_i$.
For every relevant pair of revisions r_i and r_{i+1}
Calculate tf-idf matrix for sentences in r_i and r_{i+1}
Threshold matrix to give likely candidates
Sentence Pairs in the form <A,B> Where A is a sentence from r_i and B is from r_{i+1}
Verify Candidates
For every sentence pair
Calculate Hamming distance between A and B, check it is equal to 1
Extract the edited Words: α from A and β from B
Check α and β are real words
Stem α and β, checking the stems are not equal
Check α and β are synonymous
Check β is simpler than α
If all conditions are met
False
Process next pair
True
CW corpus
Store pair <A,B> in CW Corpus

Figure 1: A flow chart showing the process undertaken to extract lexical simplifications. Each part of this process is further explained in Section 2.2. Every pair of revisions from every relevant page is processed, although the appropriate recursion is omitted from the flow chart for simplicity.

for each sentence and the matrix containing the dot product of every pair of sentence vectors from the first and second revision is calculated. This allows us to easily see those vectors which are exactly the same — as these will have a score of one.[4] It also allows us to easily see which vectors are so different that they could not contain a one word edit. We empirically set a threshold at $0.9 <= X < 1$ to capture those sentences which were highly related, but not exactly the same.

Candidate Pairs

The above process resulted in pairs of sentences which were very similar according to the tf-idf metric. These pairs were then subjected to a series of checks as detailed below. These were designed to ensure that as few false positives as possible would make it to the corpus. This may have meant discarding some true positives too, however the cautious approach was adopted to ensure a higher corpus accuracy.

Hamming Distance

We are only interested in those sentences with a difference of one word, because sentences with more than one word difference may contain several simplifications or may be a rewording. It is more difficult to distinguish whether these are true simplifications. We calculate the Hamming distance between sentences (using wordforms as base units) to ensure that only one word differs. Any sentence pairs which do not have a Hamming distance of 1 are discarded.

Reality Check

The first check is to ensure that both the words are a part of our lexicon, ensuring that there is SUBTLEX frequency data for these words and also that they are valid words. This stage may involve removing some valid words, which are not found in the lexicon, however this is preferable to allowing words that are the result of spam or vandalism.

Inequality Check

It is possible that although a different word is present, it is a morphological variant of the original word rather than a simplification. E.g., due to a change in tense, or a correction. To identify this, we stem both words and compare them to make sure they are not the same. If the word stems are equal then they are unlikely to be a simplification, so this pair is discarded. Some valid simplifications may also be removed at this point, however these are difficult to distinguish from the non-simplifications.

Synonymy Check

Typically, lexical simplification involves the selection of a word's synonym. WordNet (Fellbaum, 1998) is used as a thesaurus to check if the second word is listed as a synonym of the first. As previously discussed (Section 2.1), we do not take word sense into account at this point. Some valid simplifications may not be identified as synonyms in WordNet, however we choose to take this risk — discarding all non-synonym pairs. Improving thesaurus coverage for complex words is left to future work.

Stemming is favoured over lemmatisation for two reasons. Firstly, because lemmatisation requires a lot of processing power and would have terminally slowed the processing of the large revision histories. Secondly, stemming is a dictionary-independent technique, meaning it can handle any unknown words. Lemmatisation requires a large dictionary, which may not contain the rare CWs which are identified.

Simplicity Check

Finally, we check that the second word is simpler than the first using the SUBTLEX frequencies. All these checks result in a pair of sentences, with one word difference. The differing words are synonyms and the change has been to a word which is simpler than the original. Given these conditions have been met, we store the pair in our CW Corpus as an example of a lexical simplification.

2.3 Examples

This process was used to mine the following two examples:

[4] As tf-idf treats a sentence as a bag of words it is possible for two sentences to give a score of 1 if they contain the same words, but in a different order. This is not a problem as if the sentence order is different, there is a minimum of 2 lexical edits — meaning we still wish to discount this pair.

Complex word: functions.
Simple word: uses.
A dictionary has been designed to have one or more ________ that can help the user in a particular situation.

Complex word: difficult
Simple word: hard
Readability tests give a prediction as to how ________ readers will find a particular text.

3 Corpus Analysis

3.1 Experimental Design

To determine the validity of the CW corpus, a set of six mutually exclusive 50-instance random samples from the corpus were turned into questionnaires. One was given to each of 6 volunteer annotators who were asked to determine, for each sentence, whether it was a true example of a simplification or not. If so, they marked the example as correct. This binary choice was employed to simplify the task for the annotators. A mixture of native and non-native English speakers was used, although no marked difference was observed between these groups. All the annotators are proficient in English and currently engaged in further or higher education. In total, 300 instances of lexical simplification were evaluated, covering over 40% of the CW corpus.

A 20 instance sample was also created as a validation set. The same 20 instances were randomly interspersed among each of the 6 datasets and used to calculate the inter-annotator agreement. The validation data consisted of 10 examples from the CW corpus and 10 examples that were filtered out during the earlier stages of processing. This provided sufficient positive and negative data to show the annotator's understanding of the task. These examples were hand picked to represent positive and negative data and are used as a gold standard.

Agreement with the gold standard is calculated using Cohen's kappa (Cohen, 1968). Inter-annotator agreement is calculated using Fleiss' kappa (Fleiss, 1971), as in the evaluation of a similar task presented in de Belder and Moens (2012). In total, each annotator was presented with 70 examples and asked to label these. A small sample size was used to reduce the effects of annotator fatigue.

Annotation Index	Cohen's Kappa	Sample Accuracy
1	1	98%
2	1	96%
3	0.4	70%
4	1	100%
5	0.6	84%
6	1	96%

Table 2: The results of different annotations. The kappa score is given against the gold standard set of 20 instances. The sample accuracy is the percentage of the 50 instances seen by that annotator which were judged to be true examples of a lexical simplification. Note that kappa is strongly correlated with accuracy (Pearson's correlation: $r = 0.980$)

3.2 Results

Of the six annotations, four show the exact same results on the validation set. These four identify each of the 10 examples from the CW corpus as a valid simplification and each of the 10 examples that were filtered out as an invalid simplification. This is expected as these two sets of data were selected as examples of positive and negative data respectively. The agreement of these four annotators further corroborates the validity of the gold standard. Annotator agreement is shown in Table 2.

The 2 other annotators did not strongly agree on the validation sets. Calculating Cohen's kappa between each of these annotators and the gold standard gives scores of 0.6 and 0.4 respectively, indicating a moderate to low level of agreement. The value for Cohen's kappa between the two non-agreeing annotators is 0.2, indicating that they are in low agreement with each other.

Analysing the errors made by these 2 annotators on the validation set reveals some inconsistencies. E.g., one sentence marked as incorrect changes the fragment 'education and teaching' to 'learning and teaching'. However, every other annotator marked the enclosing sentence as correct. This level of inconsistency and low agreement with the other annotators shows that these annotators had difficulty with the task. They may not have read the instructions carefully or may not have understood the task fully.

Corpus accuracy is defined as the percentage of instances that were marked as being true instances of simplification (not counting those in the validation set). This is out of 50 for each annotator and can be combined linearly across all six annotators.

Taking all six annotators into account, the corpus accuracy is 90.67%. Removing the worst performing annotator (kappa = 0.4) increases the corpus accuracy to 94.80%. If we also remove the next worst performing annotator (kappa = 0.6), leaving us with only the four annotators who were in agreement on the validation set, then the accuracy increases again to 97.5%.

There is a very strong Pearson's correlation ($r = 0.980$) between an annotator's agreement with the gold standard and the accuracy which they give to the corpus. Given that the lower accuracy reported by the non-agreeing annotators is in direct proportion to their deviation from the gold standard, this implies that the reduction is a result of the lower quality of those annotations. Following this, the two non-agreeing annotators should be discounted when evaluating the corpus accuracy — giving a final value of 97.5%.

4 Discussion

The necessity of this corpus developed from a lack of similar resources. CW identification is a hard task, made even more difficult if blind to its evaluation. With this new resource, CW identification becomes much easier to evaluate. The specific target application for this is lexical simplification systems as previously mentioned. By establishing and improving upon the state of the art in CW identification, lexical simplification systems will directly benefit by knowing which wordforms are problematic to a user.

Methodologically, the corpus is simple to use and can be applied to evaluate many current systems (see Section 6). Techniques using distributional semantics (Bott et al., 2012) may require more context than is given by just the sentence. This is a shortcoming of the corpus in its present form, although not many techniques currently require this level of context. If necessary, context vectors may be extracted by processing Simple Wikipedia edit histories (as presented in Section 2.2) and extracting the required information at the appropriate point.

There are 731 lexical edits in the corpus. Each one of these may be used as an example of a complex and a simple word, giving us 1,462 points of data for evaluation. This is larger than a comparable data set for a similar task (de Belder and Moens, 2012). Ways to further increase the number of instances are discussed in Section 6.

It would appear from the analysis of the validation sets (presented above in Section 3.2) that two of the annotators struggled with the task of annotation, attaining a low agreement against the gold standard. This is most likely due to the annotators misunderstanding the task. The annotations were done at the individual's own workstation and the main guidance was in the form of instructions on the questionnaire. These instructions should be updated and clarified in further rounds of annotation. It may be useful to allow annotators direct contact with the person administering the questionnaire. This would allow clarification of the instructions where necessary, as well as helping annotators to stay focussed on the task.

The corpus accuracy of 97.5% implies that there is a small error rate in the corpus. This occurs due to some non-simplifications slipping through the checks. The error rate means that if a system were to identify CWs perfectly, it would only attain 97.5% accuracy on the CW corpus. CW identification is a difficult task and systems are unlikely to have such a high accuracy that this will be an issue. If systems do begin to attain this level of accuracy then a more rigorous corpus will be warranted in future.

There is significant interest in lexical simplification for languages which are not English (Bott et al., 2012; Aluísio and Gasperin, 2010; Dell'Orletta et al., 2011; Keskisärkkä, 2012). The technique for discovering lexical simplifications presented here relies heavily on the existence of Simple English Wikipedia. As no other simplified language Wikipedia exists, it would be very difficult to create a CW corpus for any language other than English. However, the corpus can be used to evaluate CW identification techniques which will be transferrable to other languages, given the existence of sufficient resources.

5 Related Work

As previously noted, there is a systemic lack of evaluation in the literature. Notable exceptions come from the medical domain and include the work of Zeng et al. (2005), Zeng-Treitler et al. (2008) and Elhadad (2006). Zeng et al. (2005) first look at word familiarity scoring correlated against user questionnaires and predictions made by a support vector machine. They show that they are able to predict the complexity of medical terminology with a relative degree of accuracy. This work is continued in Zeng-Treitler et al. (2008), where a word's context is used to predict its familiarity. This is similarly correlated against a user survey and used to show the importance of context in predicting word familiarity. The work of Elhadad (2006) uses frequency and psycholinguistic features to predict term familiarity. They find that the size of their corpus greatly affects their accuracy. Whilst these techniques focus on the medical domain, the research presented in this paper is concerned with the more general task of CW identification in natural language.

There are two standard ways of identifying CWs in lexical simplification systems. Firstly, systems attempt to simplify every word (Devlin and Tait, 1998; Thomas and Anderson, 2012; Bott et al., 2012), assuming that CWs will be modified, but for simple words, no simpler alternative will exist. The danger is that too many simple words may be modified unnecessarily, resulting in a change of meaning. Secondly, systems use a threshold over some word familiarity score (Biran et al., 2011; Elhadad, 2006; Zeng et al., 2005). Word frequency is typically used as the familiarity score, although it may also be combined with word length (Biran et al., 2011). The advent of the CW corpus will allow these techniques to be evaluated alongside each other on a common data set.

The CW corpus is similar in conception to the aforementioned lexical simplification dataset (de Belder and Moens, 2012) which was produced for the SemEval 2012 Task 1 on lexical simplification. This dataset allows synonym ranking systems to be evaluated on the same platform and was highly useful during this research (see Table 1).

6 Future Work

The CW corpus is still relatively small at 731 instances. It may be grown by carrying out the same process with revision histories from the main English Wikipedia. Whilst the English Wikipedia revision histories will have fewer valid simplifications per revision, they are much more extensive and contain a lot more data. As well as growing the CW corpus in size, it would be worthwhile to look at ways to improve its accuracy. One way would be to ask a team of annotators to evaluate every single instance in the corpus and to discard or keep each according to their recommendation.

Experiments using the corpus are presented in Shardlow (2013), further details on the use of the corpus can be found by following this reference. Three common techniques for identifying CWs are implemented and statistically evaluated. The CW Corpus is available from META-SHARE[5] under a CC-BY-SA Licence.

Acknowledgments

This research is supported by EPSRC grant EP/I028099/1. Thanks go to the annotators and reviewers, who graciously volunteered their time.

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.

Or Biran, Samuel Brody, and Noémie Elhadad. 2011. Putting it simply: a context-aware approach to lexical simplification. In *Proceed-*

[5] `http://tinyurl.com/cwcorpus`

ings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2, HLT '11, pages 496–501, Stroudsburg, PA, USA. Association for Computational Linguistics.

Stefan Bott, Luz Rello, Biljana Drndarevic, and Horacio Saggion. 2012. Can Spanish be simpler? LexSiS: Lexical simplification for Spanish. In *Coling 2012: The 24th International Conference on Computational Linguistics.*, pages 357–374.

Marc Brysbaert and Boris New. 2009. Moving beyond Kučera and Francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English. *Behavior Research Methods*, 41(4):977–990.

Jacob Cohen. 1968. Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70(4):213–220.

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.

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 Second Workshop on Speech and Language Processing for Assistive Technologies*, SLPAT '11, pages 73–83, Stroudsburg, PA, USA. Association for Computational Linguistics.

Siobhan Devlin and John Tait. 1998. *The use of a psycholinguistic database in the simplification of text for aphasic readers*, volume 77. CSLI Lecture Notes, Stanford, CA: Center for the Study of Language and Information.

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

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database.* MIT Press, Cambridge, MA.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76:378–382, November.

Robin Keskisärkkä. 2012. Automatic text simplification via synonym replacement. Master's thesis, Linköping University.

Henry Kučera and W. Nelson Francis. 1967. *Computational analysis of present-day American English.* Brown University Press.

Gerard Salton and Chung-Shu Yang. 1973. On the specification of term values in automatic indexing. *Journal of Documentation*, 29(4):351–372.

Matthew Shardlow. 2013. A comparison of techniques to automatically identify complex words. In *Proceedings of the Student Research Workshop at the 51st Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics.

S. Rebecca Thomas and Sven Anderson. 2012. WordNet-based lexical simplification of a document. In *Proceedings of KONVENS 2012*, pages 80–88. ÖGAI, September.

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 *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 365–368, Stroudsburg, PA, USA. Association for Computational Linguistics.

Qing Zeng, Eunjung Kim, Jon Crowell, and Tony Tse. 2005. A text corpora-based estimation of the familiarity of health terminology. In *Biological and Medical Data Analysis*, volume 3745 of *Lecture Notes in Computer Science*, pages 184–192. Springer, Berlin Heidelberg.

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*, 15:349–356.