LaSTUS/TALN at Complex Word Identification (CWI) 2018 Shared Task

Ahmed AbuRa'ed
Universitat Pompeu Fabra
Large Scale Text Understanding Systems Lab
TALN / DTIC
Barcelona, Spain
ahmed.aburaed@upf.edu

Horacio Saggion
Universitat Pompeu Fabra
Large Scale Text Understanding Systems Lab
TALN / DTIC
Barcelona, Spain
horacio.saggion@upf.edu

Abstract

This paper presents the participation of the LaSTUS/TALN team in the Complex Word Identification (CWI) Shared Task 2018 in the English monolingual track . The purpose of the task was to determine if a word in a given sentence can be judged as complex or not by a certain target audience. For the English track, task organizers provided a training and a development datasets of 27,299 and 3,328 words respectively together with the sentence in which each word occurs. The words were judged as complex or not by 20 human evaluators; ten of whom are natives. We submitted two systems: one system modeled each word to evaluate as a numeric vector populated with a set of lexical, semantic and contextual features while the other system relies on a word embedding representation and a distance metric. We trained two separate classifiers to automatically decide if each word is complex or not. We submitted six runs, two for each of the three subsets of the English monolingual CWI track.

1 Introduction

Automatic identification of complex words is a core component in several language-related areas of research, including *Text Simplification* (Saggion, 2017), *Lexical Simplification* (Bott et al., 2012), and *Readability Assessment* (Collins-Thompson, 2014).

The Complex Word Identification (CWI) Shared Task 2018 proposes a shared platform for evaluating complex word identification systems under four different tracks: English, Spanish and German monolingual CWI in addition to a multilingual French CWI track with only a test set; the three previously mentioned languages can be used as training for this specific track. The task has two subtasks: binary classification task; to determine if a word is complex or not, and a probabilistic classification task; the probability of how complex a word is.

In this paper we describe our work for the binary classification task under the English monolingual CWI track in which task participants were provided with a set of sentences to assess. For each sentence, one or more words have been rated as complex or not by 20 human evaluators (ten of which were native speakers).

An example sentence from this dataset is:

A lieutenant who had defected was also killed in the clashes.

In this sentence, the words 'lieutenant' and 'defected' were classified as complex by at least one out of the 20 evaluators, unlike e.g. 'killed', which did not received this label by any of them.

In our participation we cast the identification of complex words as a binary classification problem in which each word is evaluated as complex or not, given the sentence in which it occurs. We designed two systems, the first system modeled each word by a set of lexical, semantic and contextual features and evaluated distinct binary classification algorithms. This system participated from the (CWI) Shared Task 2016 at SemEval (Ronzano et al., 2016) achieving very good performance. The second system modeled each word with its context through a word embedding representation. Our approaches obtained reasonable performance in general but not in comparison with the other participating systems. For evaluation details, the reader is referred to (Yimam et al., 2018).

In Section 2 we provide an overview of relevant research related to Complex Word Identification. Section 3 and 4 respectively introduce the CWI Shared Task 2018 dataset and present the text analysis tools and resources we exploited to characterize complex words. In Section 5 we describe the features we used to build our complex word clas-

Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 159–165
New Orleans, Louisiana, June 5, 2018.

sifiers (they have been also reported in (Ronzano et al., 2016)). In Section 6 we present and discuss the performance of our Task 11 system. Finally, in Section 7 we formulate our conclusions and outline future venues of research.

2 Related Work

The identification of complex words constitutes a key aspect of *Text Simplification* (Saggion, 2017) and more specifically of *Lexical Simplification* (Bott et al., 2012). It can be defined as the problem of changing complex words by their simpler synonyms taking into account the specific context in which each word is used. Several techniques have been applied so far to identify complex words. In the context of the PSET Project (Devlin and Tait, 1998), people with aphasia were the target of the first lexical simplification system for English. The system relies on a word difficulty assessment based on psycholinguistic evidence (Quinlan, 1992) in order to decide whether to simplify a word. Recent work compared a corpora of original documents (e.g. English Wikipedia) and their 'simplified' versions (e.g. Simple English Wikipedia pages) to prompt measures which can be used to compare and rank 'quasi-synonymic' word pairs (Yatskar et al., 2010).

Besides lexical simplification, the identification of complex words constitutes a core component of *readability assessment* (Collins-Thompson, 2014), the problem of quantifying the readability of a given text. The more complex words a text has, the harder it becomes to read it. Lists of easy words (Dale and Chall, 1948), word characteristics (Kincaid et al., 1975; Gunning, 1952; Mc Laughlin, 1969), or word use in context (e.g. language models) (Si and Callan, 2001) are all techniques or resources which have been used to support the assessment of text readability: these approaches could also be used to evaluate word complexity.

The CWI Shared Task 2018 is a follow up of the CWI shared task at SemEval 2016 - Task 11 [1] reported by (Paetzold and Specia, 2016a) with the complementary evaluation paper by (Zampieri et al., 2017). 21 teams participated in the task submitting the total of 42 systems. The results concluded that word frequencies are the most reliable predictor of word complexity, also highlighted the effectiveness of Decision Trees and Ensemble methods for the task as well.

The best system by (Paetzold and Specia, 2016b) used a voting approach with threshold and machine learning-based classifiers trained on morphological, lexical, and semantic features. TALN (Ronzano et al., 2016) used a Random Forest algorithm over a set of lexical, morphological, semantic and syntactic features.

3 Dataset

The organizers of CWI Shared Task 2018 released a training set and a development set of 27,299 and 3,328 words respectively, together with the sentence in which each word occurs. For each word, the binary complexity judgments of 20 human evaluators were provided (complex word or not complex word); ten of whom where native speakers. Similarly, CWI 2018 task testing dataset consisted of 4,252 words together with the sentence in which each word occurs.

The datasets used in the shared task are described in (Yimam et al., 2017b) and (Yimam et al., 2017a) including the ones for the other tracks in this task.

4 Resources and Tools

In order to identify complex words, we characterize each word by means of a set of lexical, semantic and contextual features, in addition to Word2Vec representations. To this purpose, we analyze both the word and the sentence in which it occurs by means of the language resources and text analysis tools described in what follows.

4.1 Language Resources

To put the word embedding system in use we utilized a pre-trained word2vec model with 300 dimensions representing each vector in the vector space [2]. For the system using engineered features, information about word frequency is important. Therefore, in our complex word identification approach we exploit the word frequency data of two large corpora: (i) a 2014 English Wikipedia Dump and (ii) the British National Corpus (Leech and Rayson, 2014). We also use WordNet (Miller, 1995) to model semantic word features by relying on word senses and synset relations (e.g. hypernymy). Moreover, we use the Dale & Chall list of

[1] http://alt.qcri.org/semeval2016/task11/

[2] https://code.google.com/archive/p/word2vec/

3,000 simple words (Dale and Chall, 1948) in order to incorporate the text readability dimension, as this list contains words which 4th grade students considered understandable.

4.2 Text Analysis Tools

We analyze the sentences in which a word to evaluate occurs by means of the Mate dependency parser (Bohnet, 2010). As a result, we obtain a lemmatized and Part-Of-Speech (POS) tagged version of the sentence, along with its syntactic dependencies. Both POS tags and dependency information are used to compute several features as described in the following Section.

We also processed each sentence by the UKB graph-based Word Sense Disambiguation algorithm (Agirre and Soroa, 2009). Specifically, we benefited from the UKB implementation integrated in the Freeling workbench (Padró and Stanilovsky, 2012). In this way, we may disambiguate single or multiword expressions against WordNet 3.0.

5 Method

In order to evaluate the complexity of a word, we designed two systems, each system had different word and sentence representations.

5.1 Word Embedding (WE) System

We utilized word embeddings and modeled each sentence as a Word2Vec representation from a pre-trained model of Google News with 300 dimensions the binary classifier was trained on a set of features.

The set of features is described in the remainder of this Section:

5.1.1 Word and Context representation

Each sentence were handled by calculating the centroid of dimensions of the context before the target word, the target word and the context after the target word, generating a total of 900 features in which each 300 dimensions represent one of the three parts of the sentence. The context surrounding the target word were handled by removing any stop words and only calculating the average of all the tokens that exists in the Google News pre-trained model. Finally, in cases in which there is no context before or after the word a 300 dimensions of zeros were assigned.

5.1.2 Word and Context distance

We generated two extra features to represent the distance between the target word and the context before and after it respectively. The cosine similarity was used to calculate the distance between each pair of vectors in the vector space.

5.2 Lexical, Semantic and Contextual (LSC) features System

We modeled each word as a numeric features vector populated with a set of lexical, semantic and contextual features. In the remainder of this Section we describe the set of word features we used, and motivate their relevance with respect to the characterization of complex words. The approach taken is the same as followed in (Ronzano et al., 2016) which we explain here for the sake of completness. When presenting word features, we group subsets of related features in the same subsection (Shallow features, Dependency Tree features, etc.). It is important to note that some of the word features presented are computed by considering, besides the target word, also context words in a $[-3, 3]$ window, where position 0 refers to the target word. If the context word at a specific position cannot be determined, the value of the related feature is set to *undefined*.

5.2.1 Shallow Features

We exploited the following set of shallow word features:

- **Word length**: the length of the target word (number of characters).
- **Position of the word**: the position of the target word in the sentence. The value of this feature is normalized in the interval $[0, 1]$ by dividing the the position of the target word in the sentence by the length of the same sentence (number of words). The position of the first word of a sentence is 0.
- **Words in sentence**: the number of tokens in the sentence.

5.2.2 Dependency Tree Features

The following set of features is derived by processing the dependency tree of the sentences that include the word to evaluate:

- **Word depth in the dependency tree**: we considered the depth in the dependency tree

of the target word (*position* equal to 0), the three previous words and the three following words.

- **Parent word length**: the length (number of characters) of the parent of the current (target) word in the dependency tree.

5.2.3 Corpus-based Features

Word frequency data derived from the British National Corpus and the 2014 English Wikipedia was used to compute the following set of features:

- **British National Corpus frequency**: we considered the BNC frequency[3] of the target word lemma (*position* equal to 0), the three previous word lemmas and the three following word lemmas.

- **English Wikipedia frequency**: we considered the 2014 English Wikipedia frequency of the target word (*position* equal to 0), the three previous words and the three following words. Word frequencies are computed by tokenizing and lower-casing English Wikipedia contents.

- **Simple word list**: a binary feature to point out the presence of the target word in the Dale & Chall list.

5.2.4 WordNet features

We used WordNet 3.0 to compute the following features. Given a target word, we refer as *target-word-synsets* the set of synsets that have the same POS of the target word and include the target word among their lexicalizations (all the senses of the target word). Note that this set of features is computed without relying on Word Sense Disambiguation.

- **Number of Synsets**: the number of synsets in *target-word-synsets* (i.e. number of senses of the target word).

- **Number of Senses**: the sum of the number of word senses (i.e. the number of lexicalizations) of each *target-word-synset*.

- **Depth in the hypernym tree**: the average depth in the WordNet hypernym hierarchy among all the *target-word-synsets*.

- **Number of Lemmas**: the average number of synset lexicalizations among all the *target-word-synsets*.

- **Gloss length** (WNGloss): the average length of synset Glosses among all the *target-word-synsets*, in terms of number of tokens.

- **Number of relations** (WNRelation): the average number of semantic relations among all the *target-word-synsets*.

- **Number of Distinct POSs** (WNDistinctPOS): the number of distinct POS represented by at least one *target-word-synset*.

- **Part of Speech** (WN_*POS* - 4 features): for each WordNet POS (*POS* equal to Noun, Verb, Adjective and Adverb) we counted the number of synsets with that POS among the *target-word-synsets*, thus generating four features.

5.2.5 WordNet and corpus frequency features

The following set of features was computed by combining WordNet data, the word frequencies of the British National Corpus (BNC) and the results of the UKB WordNet-based Word Sense Disambiguation algorithm applied to the sentences where complex words appear. Thanks to the UKB algorithm, we identify the WordNet 3.0 synset that characterizes the sense of each target word (*WSD-synset*). Besides the target word, each *WSD-synset* usually has other lexicalizations, i.e. other synonyms. We retrieve the BNC frequency of all the lexicalizations of the *target-word-WSD-synset* and compute the following features:

- **Percentage of lexicalizations with higher / lower frequency than target word**: the percentage of the lexicalizations of the *WSD-synset* with a BNC frequency higher / lower than the target word BNC frequency.

- **Ratio of total lexicalizations' frequencies related to lexicalizations with higher / lower frequency than target word**: the ratio between the sum of BNC frequencies of the lexicalizations of the *WSD-synset* with a frequency higher / lower than the target word frequency and the sum of BNC frequencies of all the lexicalizations of the *WSD-synset*.

[3] http://ucrel.lancs.ac.uk/bncfreq/lists/1_1_all_fullalpha.txt.Z

We also computed the previous set of 4 features without relying on the results of the UKB Word Sense Disambiguation algorithm: we considered for each target word all the lexicalizations of all the synsets that represent possible senses and have the same POS of the same target word. Similarly to the UKB based features.

With the total of 902 features for the word embedding system and 60 features before applying any filtering to the lexical, semantic and contextual features System, we enabled the training and evaluation of distinct binary classification algorithms tailored to determine whether a word is complex or not. To this end, we relied on the Weka machine learning framework (Witten and Frank, 2000).

6 Results

We evaluated the performance of five classification algorithms: Support Vector Machine (with linear and radial basis function kernels), Naïve Bayes, Logistic Regression, Random Tree and Random Forest. We applied 10 fold-cross validation over the training data, based on the obtained reults we decided to build the classifiers using Random Forest for both systems since they performed best over the whole dataset. The results of the Random Forest system in 10-fold cross validation experiments over the training data can be seen in Table 1.

Table 1: 10-fold cross validation over the training datasets

System	Dataset	P	R	F
WE	News	0.810	0.811	**0.810**
	WikiNews	0.741	0.742	0.736
	Wikipedia	0.708	0.703	0.694
	all	0.803	0.803	**0.803**
LSC	News	0.796	0.793	0.787
	WikiNews	0.747	0.745	**0.738**
	Wikipedia	0.769	0.768	**0.766**
	all	0.785	0.783	0.778

Tables 2, 3 and 4 presents the top 3 systems participating in the evaluation together with our results. We have obtained mixed results: in the English News our Word Embedding (WE) system outperformed the system based on human engineered features (LSC) – eleventh position in the ranking. While the LSC system performed better on WikiNews and Wikipedia, placing the team in the tenth position in the ranking.

Table 2: Comparison with the top three teams for the English News submissions

Team	Accuracy
camb	0.8792
dirkdh	0.8721
TMU	0.8706
WE	0.8172
LSC	0.7785

Table 3: Comparison with the top three teams for the English WikiNews submissions

Team	Accuracy
camb	0.8430
ajason08	0.8368
nathansh	0.8329
LSC	0.7615
WE	0.7374

Table 4: Comparison with the top three teams for the English Wikipedia submissions

Team	Accuracy
camb	0.8115
nathansh	0.7966
andrei.butnaru	0.7920
LSC	0.7414
WE	0.6966

7 Conclusion

In conclusion, we tried to approach the problem of identifying complex words at the CWI shared task 2018 by designing two systems based on binary classifiers, one represents the context as word embedding vectors and the other use a set of lexical, semantic and contextual features. The WE system performed better in the English News part and the LSC system excelled for Wikinews and Wikipedia. For future work we are planning on better analyzing our set of features by applying some feature selection methods e.g. info gain. Afterwards, we will attempt deep-learning neural networks to create our classifiers.

Acknowledgments

This work is (partly) supported by the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502) and by the TUNER project (TIN2015-65308-C5-5-R, MINECO/FEDER, UE).

References

Eneko Agirre and Aitor Soroa. 2009. Personalizing pagerank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 33–41. Association for Computational Linguistics.

Bernd Bohnet. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China. Coling 2010 Organizing Committee.

Stefan Bott, Luz Rello, Biljana Drndarevic, and Horacio Saggion. 2012. Can Spanish Be Simpler? LexSiS: Lexical Simplification for Spanish. In *Proceedings of the 24th International Conference on Computational Linguistics (CoLing 2012)*.

Kevyn Collins-Thompson. 2014. Computational assessment of text readability. a survey of current and future research. *ITL - International Journal of Applied Linguistics 165:2*, 165(2):97–135.

Edgar Dale and Jeanne S. Chall. 1948. The concept of readability. *Elementary English*, 23(24).

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.

Robert Gunning. 1952. *The Technique of Clear Writing*. McGraw-Hill.

J. Peter Kincaid, Robert P. Fishburne Jr., Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel. Technical report, Naval Technical Training Command.

Geoffrey Leech and Paul Rayson. 2014. *Word frequencies in written and spoken English: Based on the British National Corpus*. Routledge.

G. Harry Mc Laughlin. 1969. SMOG grading - a new readability formula. *Journal of Reading*, pages 639–646.

George A. Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Lluis Padró and Evgeny Stanilovsky. 2012. Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey. ELRA.

Gustavo Paetzold and Lucia Specia. 2016a. SemEval 2016 Task 11: Complex Word Identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 560–569, San Diego, California. Association for Computational Linguistics.

Gustavo Paetzold and Lucia Specia. 2016b. Sv000gg at semeval-2016 task 11: Heavy gauge complex word identification with system voting. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 969–974, San Diego, California. Association for Computational Linguistics.

Philip T. Quinlan. 1992. *The Oxford Psycholinguistic Database*. Oxford University Press.

Francesco Ronzano, Ahmed Abura'ed, Luis Espinosa Anke, and Horacio Saggion. 2016. Taln at semeval-2016 task 11: Modelling complex words by contextual, lexical and semantic features. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1011–1016, San Diego, California. Association for Computational Linguistics.

Horacio Saggion. 2017. *Automatic Text Simplification*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Luo Si and Jamie Callan. 2001. A statistical model for scientific readability. In *Proceedings of the Tenth International Conference on Information and Knowledge Management*, CIKM '01, pages 574–576, New York, NY, USA. ACM.

Ian H. Witten and Eibe Frank. 2000. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

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 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 365–368.

Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo Paetzold, Lucia Specia, Sanja Štajner, Anaïs Tack, and Marcos Zampieri. 2018. A Report on the Complex Word Identification Shared Task 2018. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications*, New Orleans, United States. Association for Computational Linguistics.

Seid Muhie Yimam, Sanja Štajner, Martin Riedl, and Chris Biemann. 2017a. CWIG3G2 - Complex Word Identification Task across Three Text Genres and Two User Groups. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 401–407, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Seid Muhie Yimam, Sanja Štajner, Martin Riedl, and Chris Biemann. 2017b. Multilingual and Cross-Lingual Complex Word Identification. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP*

2017, pages 813–822, Varna, Bulgaria. INCOMA Ltd.

Marcos Zampieri, Shervin Malmasi, Gustavo Paetzold, and Lucia Specia. 2017. Complex Word Identification: Challenges in Data Annotation and System Performance. In *Proceedings of the 4th Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2017)*, pages 59–63, Taipei, Taiwan. Asian Federation of Natural Language Processing.