Vicomtech at ALexS 2020: Unsupervised Complex Word Identification Based on Domain Frequency

Elena Zotova, Montse Cuadros, Naiara Perez and Aitor García-Pablos

SNLT group at Vicomtech Foundation, Basque Research and Technology Alliance (BRTA), Mikeletegi Pasealekua 57, Donostia/San-Sebastián, 20009, Spain

Abstract

This paper introduces Vicomtech's systems for unsupervised complex word identification submitted to the ALexS "Análisis Léxico en la SEPLN 2020" task. The systems are based on clustering algorithms with domain specific features, such as word frequency and probability in several Wikipedia corpora, word length, and number of synsets in WordNet. Our systems are designed to identify complex words, taking into account occurrence of the word in domain-specific texts in order to be able to adapt to the domain. Our systems reported good results, performing in second position.

Keywords

Complex Word Identification, Lexical Simplification, Unsupervised Learning

1. Introduction

Complex word identification (CWI) is one of the steps in the process of lexical simplification, which is useful for learners and children in text comprehension [1]. CWI and further substitution of the complex words may significantly improve readability and understandability of a given text.

CWI is a relatively recent area of interest with at least two shared tasks focused on it in the past years—the CWI Shared Tasks at SemEval 2016 [2] and NAACL-HTL 2018 [3]. Both challenges set out the problem of detecting words difficult to understand for non-native speakers, and introduce annotated corpora in English, Spanish, German and French in order to develop supervised machine learning systems to that end. The majority of the CWI systems presented in those shared tasks explore a large number of features—morphological, lexical, semantic, collocational, syntactical, psycho-linguistic, etc. For instance, the winners of in 2016 [4] and 2018 [5] leverage 69 and 27 features, respectively.

ALexS "Análisis Léxico en la SEPLN 2020" [6] is the first shared task on lexical analysis of university educational texts in Spanish. The organizers of the task propose to implement automatic systems to identify difficult words in the texts, with the following key challenges:

Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020)

EMAIL: ezotova@vicomtech.org (E. Zotova); mcuadros@vicomtech.org (M. Cuadros); nperez@vicomtech.org (N. Perez); agarciap@vicomtech.org (A. García-Pablos)

ORCID: 0000-0002-8350-1331 (E. Zotova); 0000-0002-3620-1053 (M. Cuadros); 0000-0001-8648-0428 (N. Perez); 0000-0001-9882-7521 (A. García-Pablos)

CEUR Workshop Proceedings (CEUR-WS.org)

Table 1
Corpus characteristics

	Number
Transcriptions (documents)	55
Subjects	25
Words	68,414
Complex words	1,084
Words/document, avg	1,244
Words/document, max	2,646
Words/document, min	465

- The difficult terms have to be within the scope of an academic content, which implies that many technical terms should be excluded because they are commonly used in the domain; that is, the systems should be able to adapt to the domains/university subjects in the dataset.
- The task has been evaluated according to the manual annotations of the organizers over the given corpus. The corpus was released without the annotations and no annotation scheme has been published; thus, the participants have had to elaborate their own definition of 'complex word'.
- As no training data has been released, all the automatic systems are expected to use unsupervised learning algorithms.

The VYTEDU corpus [7] provided as dataset for this task consists of 55 transcriptions of videotaped lectures in the University of Guayaquil (Ecuador). The lectures are delivered in Spanish and belong to different subjects, such as botany, psychology, economy, programming, architecture, etc. The purpose of the corpus is to analyze lexical complexity of written and oral text, and develop an automatic text simplification system. Table 1 provides a quantitative description of the dataset.

This paper is organized as follows. Section 2 describes proposed systems for automatic complex word identification and consists of parts about Feature engineering (Subsection 2.1), and clustering methods (2.2). Further, we report the result of the task and performance of proposed systems in Section 3. Finally, we speak about conclusions in Section 4.

2. Systems Description

This section provides a description of the automatic CWI systems with which the reported results have been obtained. First, we explain the features used for training the systems, and we explain the process of building a system with unsupervised learning methods used to solve the task—in this case, clustering.

Previous studies [8, 9, 10] proved that a word's difficulty is closely related to its frequency in a corpus, so the frequency can be used to detect complex words automatically. Words that

rarely occur may be new to a reader or listener and thus not easy to understand. However, the frequency of a word may vary among domains, and the word's understandability may vary with the reader's or listener's familiarity with the topic. For instance, the word 'inflation' may be easy to understand for those who have thorough knowledge of finances, but difficult for a student that has just started economical studies. Our CWI system takes into account word frequency not only in a given text or in the corpus, but also in the domain of the target text.

2.1. Features

Several word-level features have been selected to be used in the clustering process. These features are based on length, frequencies and probabilities of the words and their lemmas in large corpora, domain-related corpora and in the VYTEDU documents. Word or lemma frequency is the count of all the occurrences in a given corpus. Word or lemma probability is the proportion of the frequency of a word/lemma to the total number of words/lemmas in the corpus. Frequency and probability in a large corpus is expected to separate commonly used words in all domains, while domain-specific metrics refer to well-known terminology of the domain.

We have used the Spanish Wikipedia data dump and category list from March 20, 2020 [11] and WikiExtractor [12] to get a large corpus of text in Spanish. We have also extracted domain-specific corpora related to the each subject in the VYTEDU dataset. We first mapped manually Wikipedia categories to the 25 subjects encountered in the dataset (e.g., *Contabilidad*, accounting, *Investigación*, research, etc.). Then, we selected all the Wikipedia articles in each category and immediate sub-categories in the category hierarchy. The resulting domain-specific corpora contain 71 to 1,432 articles, depending on how broadly the topic of the subject is represented in Wikipedia. The whole Wikipedia corpus have been tokenized, and domain-specific corpora and VYTEDU documents have been tokenized and also lemmatized. All the pre-processing has been done with spaCy's [13] statistical model `es_core_news_sm`, pretrained on the AnCora and Spanish WikiNER.

The features are explained below.

- **Lemma length**: Each word's lemma's length is calculated. Word length is a common measure of text complexity; for instance, average word length is used in Flesch-Kincaid readability tests [14], and the Automated Readability Index [15] takes into account proportion of words and characters.
- **Lemma frequency in the subject documents:** We calculate the frequency of all lemmas in the documents of given subjects in the VYTEDU corpus.
- **Number of synsets in WordNet:** We take the number of synsets in the Spanish WordNet [16] for each lemma using NLTK Toolkit[17]. According to the studies of [18], it has been known that older—therefore better known—words are more polysemous than recent words, and that frequently used words are more polysemous than infrequent ones. Hence, we assume that the more polysemous a word is, the less complex it is likely to be.
- **Lemma frequency in domain corpora:** We calculate the frequency of all lemmas in the domain-specific corpora.

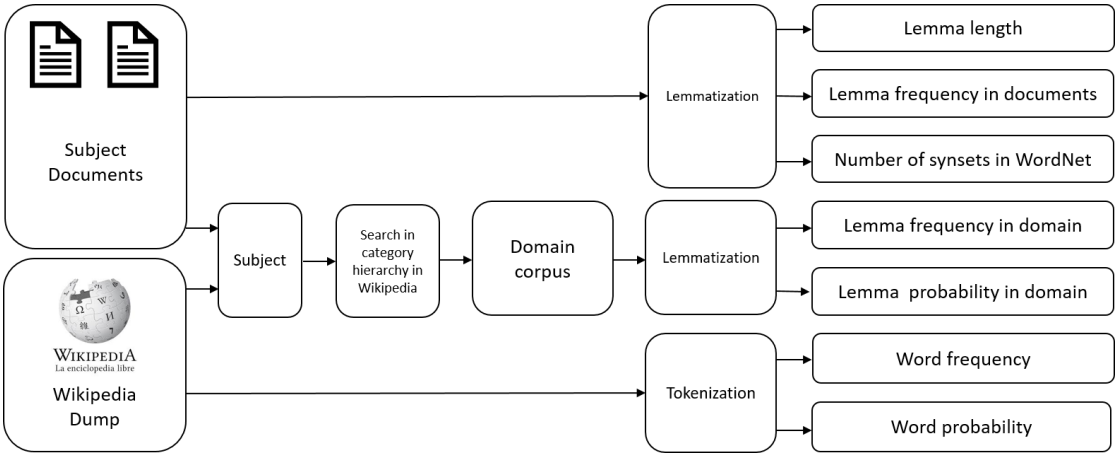

Figure 1: The process of feature engineering using large corpora from Wikipedia

- **Lemma probability in domain corpora:** We calculate the probability of all lemmas in the domain-specific corpora.
- **Word frequency in Wikipedia:** We calculate the frequency of all words in the Wikipedia.
- **Word probability in Wikipedia:** We calculate the probability of all words in the Wikipedia.

All features were normalized from 0 to 1. The workflow of the process is shown in Figure 1

2.2. Clustering

We use clustering as the unsupervised learning method to classify the words in the documents in VYTEDU corpus. The clustering process has been carried out with each domain-related dataset separately, grouping the documents by subject, in order to obtain domain-specific complex words.

As a pre-processing step, we have automatically extracted words candidate to be complex from the VYTEDU dataset. First, we have computed a bag of words by tokenizing and lemmatizing the dataset. Next, we have filtered out stopwords and words that do not belong to meaningful parts of speech, namely, substantives, verbs, adjectives and adverbs. We argue that the removed words are never complex, regardless of the context they occur in. Next, we have lemmatized all the target words.

Then, the clustering has been carried out as follows. We have assigned a feature vector to each target word from pre-calculated values explained in Section 2.1. We have used the lemma of the word for some of the features: lemma length, lemma frequency in the subject-grouped documents in VYTEDU corpus, number of synsets in WordNet, and frequency and probability in the domain-related corpus. In addition, we have used the word itself to assign word frequency and probability in Wikipedia. The obtained vector has size $7 \times 28,798$, i.e., the number of features times the total number of words candidate to be complex.

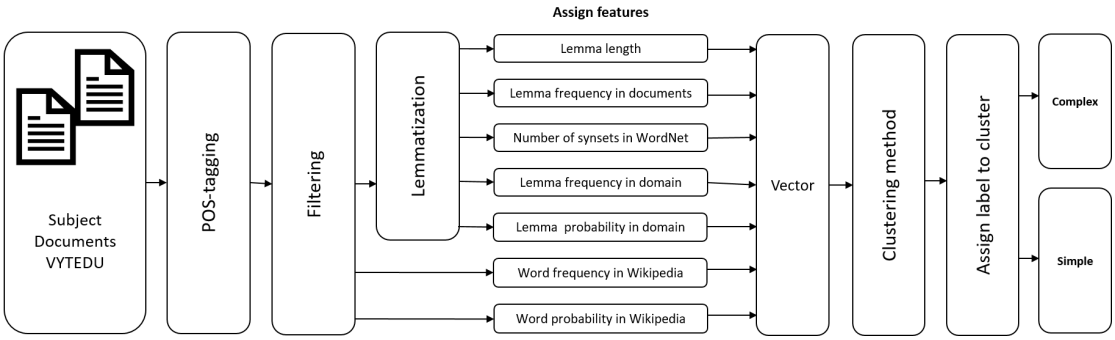

Figure 2: The workflow of the clustering process

In order to select the clustering methods and the adequate parameters, previously we conducted various experiments with annotated corpus from SemEval 2018 task in Spanish [2]. As a result, we have applied two best performing clustering methods—Gaussian Mixture Models (GMM) [19] and K-means [20]—to form two clusters, with the intention that one of the clusters comprises complex words and the other simple words. The GMM model uses spherical covariance. The K-means model uses Elkan's algorithm [21]. Both models have been implemented in Python with the scikit-learn framework [22].

Once the clusters have been established, the system must be able to assign a label (i.e., simple or complex) to each of them. On the grounds that less frequent, probable and polysemous words are more likely to be difficult, we automatically assign the label 'complex' to the words in the cluster with the lowest average value of features.

The entire workflow of the clustering process is depicted in Figure 2.

3. Results

We have presented three runs of the system:

- Run 1: A GMM clustering model trained on all the features (described in Section 2.1).
- Run 2: A K-means clustering model trained on all the features.
- Run 3: A K-means clustering model trained on all the features except the number of synsets in WordNet.

The systems have been evaluated in terms of Accuracy, Precision, Recall, and F1-score, F1-score being the key metric to compare classification systems. In addition, we provide one more metric—G-score—, which consists in the harmonic mean of Accuracy and Recall. The G-score was used in the previous shared tasks [2] because it is important in the task of CWI to minimise the number of false negatives (i.e., complex words being identified as simple) and false positives (i.e., simple words being identified as complex), and also maximise the number of true positives (i.e., complex words identified as complex). One method to measure whether a system achieves these goals, is to give more emphasis to Accuracy to account for the former two and Recall for

Table 2
Results of the proposed systems

System	Accuracy	Precision	Recall	F1 score	G score	# CW
Run 1	90.55	9.68	59.69	16.66	71.95	6,682
Run 2	88.68	9.18	**68.82**	16.20	**77.50**	8,123
Run 3	91.29	10.40	59.32	17.70	71.91	6,158
Organizers	92.17	12.32	65.50	20.74	76.58	5,794
Best	**98.25**	**34.16**	22.67	**27.25**	36.84	726

the latter. This measure considers the minority of the complex class in any text and therefore gives less weight to the false positives. It is possible to tune the system towards higher recall by regulating the proportion between complex words and all words of the corpus.

The results of the proposed systems are shown in Table 2. The challenging nature of the task is reflected in the performance of the systems of all participants. High accuracy with low precision is a result of the imbalanced dataset, where the positive class is underrepresented: there were only 1,084 complex words from 68,414, that is 1.6% of all the words of the corpus.

Our best performing system in terms of F1 score (Run 3) is built without the WordNet feature; thus, we can conclude that this feature is not as relevant for this task. If we take G-score into consideration, the most proficient system is Run 2, which uses all the available features and the K-means clustering method.

4. Conclusions

Automatic complex word detection with unsupervised methods is a highly challenging task for various reasons. First of all, the definition of complex word is very subjective and depends on the annotators, their level of education, whether they are familiar with the domain, and so on. Secondly, no criteria has been provided about what a complex word is in this particular task. That is why our system is based on the intuition that frequency of the word in a given corpus and its polysemy in WordNet may give significant information. At last, the task of CWI always deals with unbalanced data, where the proportion of complex words depends on the level of difficulty of the text. Our systems consider the domain of the document, but not its difficulty.

Our systems leverage a semi-automatic process of corpus extraction from Wikipedia, based on manual formatting of the subjects. One of the possible improvements could be creating an automatic topic modeling system. This method may also be used to create silver labels for further training supervised learning models.

Acknowledgments

This work has been supported by Vicomtech and partially funded by the project DeepReading (RTI2018-096846-B-C21, MCIU/AEI/FEDER,UE)

References

[1] M. Shardlow, A Comparison of Techniques to Automatically Identify Complex Words., in: 51st Annual Meeting of the Association for Computational Linguistics Proceedings of the Student Research Workshop, Association for Computational Linguistics, Sofia, Bulgaria, 2013, pp. 103–109.

[2] G. Paetzold, L. Specia, SemEval 2016 Task 11: Complex Word Identification, in: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), Association for Computational Linguistics, San Diego, California, 2016, pp. 560–569. doi:10.18653/v1/S16-1085.

[3] S. M. Yimam, C. Biemann, S. Malmasi, G. Paetzold, L. Specia, S. Štajner, A. Tack, M. Zampieri, A Report on the Complex Word Identification Shared Task 2018, in: Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 66–78. doi:10.18653/v1/W18-0507.

[4] G. Paetzold, L. Specia, 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), Association for Computational Linguistics, San Diego, California, 2016, pp. 969–974. doi:10.18653/v1/S16-1149.

[5] S. Gooding, E. Kochmar, CAMB at CWI Shared Task 2018: Complex Word Identification with Ensemble-Based Voting, in: Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 184–194. doi:10.18653/v1/W18-0520.

[6] J. Otriz Zambrano, A. Montejo-Ráez, AlexS 2020: Lexical Analysis Task at SEPLN, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020), CEUR-WS, Malaga, Spain, 2020.

[7] J. A. Ortiz Zambrano, A. Montejo-Ráez, VYTEDU: Un Corpus de Vídeos y sus Transcripciones para Investigación en el Ámbito Educativo, Procesamiento del Lenguaje Natural 59 (2017) 167–170.

[8] L. Specia, S. K. Jauhar, R. Mihalcea, SemEval-2012 Task 1: English Lexical Simplification, 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), Association for Computational Linguistics, Montréal, Canada, 2012, pp. 347–355.

[9] G. H. Paetzold, L. Specia, A Survey on Lexical Simplification, J. Artif. Int. Res. 60 (2017) 549––593. doi:10.1613/jair.5526.

[10] T. Kajiwara, M. Komachi, Complex Word Identification Based on Frequency in a Learner Corpus, in: Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 195–199. doi:10.18653/v1/W18-0521.

[11] Wikimedia, Wikimedia Downloads, 2020. URL: https://dumps.wikimedia.org/.

[12] G. Attardi, Wikiextractor, 2015. URL: https://github.com/attardi/wikiextractor.

[13] spaCy.io, spaCy, 2016. URL: https://spacy.io/.

[14] R. Flesch, A New Readability Yardstick, The Journal of Applied Psychology 32 (1948)

221–233. doi:10.1037/h0057532.

[15] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, B. S. Chissom, Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel, Naval Technical Training Command Millington TN Research Branch (1975).

[16] A. Fernández-Montraveta, G. Vázquez, C. Fellbaum, The Spanish Version of WordNet 3.0, Text Resources and Lexical Knowledge. Mouton de Gruyter (2008) 175–182.

[17] Bird, Steven, Edward Loper and Ewan Klein, Natural Language Processing with Python., 2009. URL: https://www.nltk.org.

[18] C. J. Lee, Some Hypotheses Concerning the Evolution of Polysemous Words, Journal of Psycholinguistic Research 19 (1990) 211–219. doi:10.1007/BF01077257.

[19] G. J. McLachlan, K. E. Basford, Mixture Models : Inference and Applications to Clustering, volume 84, Marcel Dekker, 1988. doi:10.2307/2348072.

[20] J. MacQueen, Some Methods for Classification and Analysis of Multivariate Observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, University of California Press, Berkeley, Calif., 1967, pp. 281–297.

[21] C. Elkan, Using the Triangle Inequality to Accelerate K-Means, in: Proceedings of the Twentieth International Conference on International Conference on Machine Learning, AAAI Press, 2003, p. 147–153.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.