General Lexicon-Based Complex Word Identification Extended with Stem N-grams and Morphological Engines

Antonio Rico-Sulayes[a]

[a]*Universidad de las Américas Puebla, San Andrés Cholula, Puebla, 72810, Mexico*

Abstract

This article introduces a CWI system developed to target the VYTEDU corpus, which consists of transcribed college classes in Spanish. With no in-house training data, the system presented relies on lexical complexity, based on the frequency of a general lexicon. The lexicon has been extended with stem n-grams, derived from its own dictionary entries, and a verbal morphological parser. In order to make the system sensitive to both the familiarity of technical terms within their domain and the depth of discussion in different course levels, the system uses document-based normalized frequency to filter out familiar technical terms from the CW candidate list. With three runs competing in the Lexicon Analysis Task, ALexS 2020, at IberLEF, the system developed achieved the highest F1, accuracy and precision scores of all nine methods submitted by five teams.

Keywords

lexical complexity, document-based normalization, lexical frequency thresholding

1. Introduction

Complex word identification (CWI) is the task of detecting document words judged as difficult or complex by the members of a target population [1]. In the NLP community, this task has attracted increased attention and has recently resulted in the organization of competitions at different venues, such as SemEval [2] and NAACL-HTL[3]. This article introduces a system that has participated in the CWI competition ALexS 2020, hosted at IberLEF 2020.

ALexS 2020 differs from previous CWI competitions in a number of aspects. Most importantly, this competition, unlike the competitions at SemEval and NAACL-HTL, has provided participants only with a non-annotated corpus and a target number of 723 tags. The other two competitions presented a training data set that allowed for the implementation of supervised approaches, through which a system can be trained on a substantial amount data with known solutions. Another aspect that makes the two former competitions different from ALexS 2020 is the target population that identified complex words in these previous events. In these events, non-native speakers produced the tags for CWs. Finally, an important difference among all these competitions is that SemEval used English data [2], NAACL-HTL used a multilingual data

Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020)
EMAIL: antonio.rico@udlap.mx (A. Rico-Sulayes)
URL: https://sites.google.com/site/ricosulayes/ (A. Rico-Sulayes)
ORCID: 0000-0003-0932-4733 (A. Rico-Sulayes)

set that included four target languages (English, German, Spanish, and French) [3], and ALexS 2020 focused on Spanish data.

As for the system developed, this article introduces a CWI system based on the use of frequent lexical items in a general lexicon. In the context of CWI, this kind of approach is also called lexical complexity based on frequency [1]. Lexicon-based classification/detection has also been pervasive in various areas of computational linguistics [4]. Among the areas that have benefited from this approach for a number of years are sentiment analysis and opinion mining [5, 6]. The general lexicon-based CWI system developed here has been extended through a number of modules. Extending lexicons for classification has also been a widely used technique to improve automatic classification and detection [4]. A first, simple extension added proper names and internet-related vocabulary to the general lexicon. In a more elaborate extension, a second module was populated by stem n-grams derived from the general lexicon's own frequent dictionary entries. The most common verbs with all their conjugated forms, as produced by an online morphological engine, are also added to a third filtering module, which has a morphological parser of its own to deal with compound verbal forms containing enclitic pronouns (such as mándamelo, 'pay it to me'). These three elements (the general lexicon frequencies complemented by proper names and technology-related vocabulary, the stem n-grams, and the verbal morphological parser) have the goal of filtering out implausible CW candidates from the system. Finally, a review of the distribution of CW candidates in their source documents, based on normalized frequency, allows us to determine whether a given candidate should be excluded from the final CW list despite being a technical term. This can be considered a document-based variation of what [1] calls lexical frequency thresholding. The interaction of all these modules allows the system to produce a CW candidate list without ever using any previously annotated data.

2. Data

The system has been applied to a corpus of 55 files, with 68,618 tokens and 8,084 types. These files represent transcriptions of videotaped classes at the University of Guayaquil, in Ecuador. This corpus, which is a developers' version, does not have any annotations. It has been derived from the VYTEDU-CW corpus, which is annotated for the presence of CWs [7]. As announced to competing teams, the annotated corpus has 723 tags.

Since there is not a training, annotated corpus available for the development of the CWI system, it was not possible to develop a supervised learning system based on an in-house data set. Besides this special challenge of the ALexS 2020 competition, there are two additional challenges unique to this event, as compared to other CWI competitions. First, CWs have to be considered within their genre. Namely, technical terms should be excluded from the identification if they are commonly used in their domain. Second, this genre-based exclusion of technical terms should also be dependent upon the depth of discussion in each transcribed video, as different classes on the same topic may represent different course levels.

3. Methods

In order to tackle the task of CWI without any previously annotated data, the system presented in this section uses a general lexicon-based approach expanded with a dictionary of stem n-grams and a dictionary of verbal forms, which uses the conjugation produced by an online morphological engine. These three dictionaries rely on a number of parameters that either expand or reduce the number of lexical items employed in the task. Finally, to be sensitive to the genre-based exclusion of technical terms common within their domain, the system uses a calculation of document-normalized frequency, as a document-based lexical frequency thresholding. All of these modules, the three dictionaries and the normalized frequency-based exclusion of technical terms, are explained in the rest of this section.

3.1. CWI system components

The first module of the proposed system uses a dictionary of general Spanish. The dictionary is derived from the total list of types in Corpus de Referencia del Español Actual (CREA) [8]. This general lexicon corpus has 152,558,294 tokens and 737,799 types, appearing in documents from 22 Spanish-speaking countries. This list was pre-processed to eliminate various lexical types, such as numerical data (which included dates and quantities). After pre-processing this list, several experiments were conducted to filter out CW candidates from the VYTEDU corpus based on the most frequent dictionary entries of this lexicon. After several experiments, it was determined that the ideal size of the dictionary to filter out CW candidates was between 50,000 and 75,000 lexical entries. Experiments with a 25,000-lexical entry dictionary revealed that too many common words became CW candidates, and a 100,000-lexical entry dictionary excluded too many CWs that appeared to be acceptable candidates. Since the documents in CREA date from 1975 to 2004, this first general lexicon was extended with an internet vocabulary [9] and a list of common first names [10, 11, 12], which were noted in the first few experiments. A list of last names was not included as they sometimes are part of academic concepts, such as in the Doppler Effect or Chomsky Hierarchy.

The second module of the system developed includes the extension of the first original dictionary without the extensions just mentioned. This module extracts the most common stem n-grams from the original dictionary lexical entries. Testing the system suggested that best conditions for this module were reached when using 5-grams with a minimum frequency between six and 12 instances. A third module also expands the first general lexicon using the output from the online morphological conjugator in Diccionario de la Lengua Española (DLE) [13], although it also works with the output of other morphological engines. This last module uses only the conjugation of the most common verbs, as derived from the list of types in the first module. The module also includes a number of morphological rules to check if a CW candidate is a compound verb form with enclitic pronouns and filters it out if this is the case.

Finally, the fourth module attempted to respond to two additional challenges announced by the competition organizers: adjusting CWI systems to reject technical terms that are commonly used in their domain and considering the depth of discussion that can make technical terms more or less complex at different course levels. In order to respond to these challenges, the system uses a document-based lexical frequency thresholding, implemented in the form of a

Table 1
Effects of lexical complexity values on the number of CW candidates

Lexical complexity	Number of CW candidates
50,000 most frequent types	807
62,500 most frequent types	647
75,000 most frequent types	531

document-based normalized frequency. This normalized frequency, normFreq, is calculated with formula (1) as shown below. This formula assumes the following conditions: given a corpus C with m number of documents, $C = d_1, d_2, \dots, d_m$, any given document in the corpus has n number of words, $d \in C = w_1, w_2, \dots, w_n$, and j number of types $d \in C = t_1, t_2, \dots, t_j$, and any given type in the document has k number of instances or tokens in the document $t \in d = i_1, i_2, \dots, i_k$.

$$normFreq(t \in d) = \left(\frac{k}{n}\right)\left(\frac{\sum n \sum d \in C}{m}\right) \quad (1)$$

The normalized frequency for any CW candidate is obtained by dividing its number of instances by the number of words in the document. Then, the result is multiplied by the average number of words per document in the corpus.

The following section explains how these four modules can be adjusted to produce CW lists of different lengths, depending on how their selection parameters are manipulated. Through this manipulation, the system attempts to extract the targeted 723 CWs using different values for the modules presented. For ALexS 2020, three runs have been submitted for the task evaluation.

3.2. Parameter Combinations for Tested Approaches

For each of the four modules formerly described, three different options were selected to produce a CW candidate with a number of elements as close as possible to the target figure of 723. In the lexicon module, three dictionary sizes were used, with 50,000, 62,500, and 75,000 lexical elements. In order to appreciate the effects of changing the number of words in the first module, Table 1 shows the number of CW candidates extracted when changing the size of the general lexicon dictionary, while keeping the stem n-gram minimum frequency at 5 and the normalized frequency greater than 3. Table 1 shows that, as one would expect, using more frequent types in the dictionary from module one filters out more CW candidates, decreasing the size of the extracted list.

Since the size of the dictionary for module one was tested and decided on first, modules two (and eventually module four) became rather dependent on the lexical complexity values implemented in the first module. As to the effects of using more or less frequent stem n-grams, Table 2 shows the effect of using a minimum stem n-gram frequency of 5, 8, and 12, while keeping the lexicon dictionary size at 75,000 entries and the normalized frequency greater than 3. As shown in Table 2, the effect of using stem n-grams that are more frequent eliminates n-grams from the dictionary in module two and produces lists with more CW candidates, i.e., when there are fewer patterns to filter out CWs, the list of these grows.

Table 2
Effects of stem n-gram minimum frequency on the number of CW candidates

N-gram frequency	Number of n-grams	Number of CW candidates
Min freq >= 5	3,232	531
Min freq >= 8	1,855	596
Min freq >= 12	1,141	647

Table 3
Effects of normalized frequency on the number of CW candidates

Normalized frequency values	Number of CW candidates
Normalized freq < 2	549
Normalized freq < 3	647
Normalized freq < 4	695

Table 4
Configuration of three runs competing at ALexS 2020

Run	Approach parameters	Number of CWc
1	50,000 entry lexicon, stem minfreq = 6 (1,737), normalFreq < 2.1	720
2	62,500 entry lexicon, stem minfreq = 8 (1,549), normalFreq < 3.6	726
3	75,000 entry lexicon, stem minfreq = 12 (1,141), normalFreq < 6	724

As for module 3, no parameters were changed in this case. In general, the ultimate goal was to have as many frequent verbs as possible. The final dictionary included all conjugated forms for slightly over one hundred frequent verbs.

Module four at the end also became dependent on module one. The effects of changing the document-based normalized frequency can be observed in Table 3. This table shows the effects of using a general lexicon dictionary with 62,500 types and a stem n-gram minimum frequency of 5, while alternatively using a normalized frequency greater than 2, 3, and 4. Table 3 shows that the greater the minimum value of normalized frequency, the more CW candidates are extracted. This is because when the system uses a larger frequency value, only CWs of very frequent types are eliminated from the list, and this produces a larger list of candidates.

As mentioned before, the size of the general lexicon dictionary was first tested and chosen. Then, module two and four were manipulated until three dictionaries of 50,000, 62,500, and 75,000 entries produced a list with a number of CWs as close as possible to the 723 target figure. The three resulting configurations of the system are shown in Table 4. The first run listed in Table 4 uses a dictionary of 50,000 entries, a stem n-gram minimum frequency of 6, which produces 1,737 stems, and a normalized frequency greater than 2.1. These settings produced a CW candidate list of 720 items. The equivalent values for the two other runs are also shown in this table.

Table 5
Confusion matrix for run 2

n = 68,621	Predicted = Yes	Predicted = No
Actual = Yes	TP = 248	FN = 846
Actual = No	FP = 478	TN = 67,049

4. Analysis of results

Five teams submitted a total of nine solutions to the CWI task in ALexS 2020. The results for all the competing methods were presented and compared in terms of four different metrics: accuracy, precision, recall and F1. The main goal of a CWI task is to detect and locate the words previously identified as complex by a target population. Therefore, every time that word is identified as complex by the system, if it was also identified as such by the population, it represents a true positive (TP). If the target population did not identified this word, the system has produced a false positive (FP) instead. In a similar fashion, words identified as non-complex by both the system and the population are true negatives (TN), and words that the system identifies as non-complex, but the population as complex, are false negatives (FN). Table 5 shows the confusion matrix with these values for run 2 in Table 4 – this configuration achieved the highest accuracy, precision, and F1 of all models submitted to the competition, as discussed further below in this section.

Using the information from Table 5, an accuracy of 0.98 is obtained solving for formula (2). Formula (3) renders a precision of 0.34, formula (4) produces a re-call value of 0.23, and with formula (5), an F1 of 0.27 is obtained [14].

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}$$

$$precision = \frac{TP}{TP + FP} \tag{3}$$

$$recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 = \frac{2 * (precision * recall)}{precision + recall} \tag{5}$$

The former values for run 2 can be observed in the first row of Table 6, which summarizes the evaluation of the five teams and their nine runs or methods submitted to the event organizers. The methods in this table have been sorted according to F1, which synthesizes in one figure both, precision and recall, and is included in the rightmost column of the table. As shown in Table 6, taking into account F1 scores, the three runs resulting of applying the system here developed obtained the first, second, and third best results at ALexS 2020. Another important value that is often taken into account to measure the overall performance of a classification/detection system is accuracy. This metric combines all the correct judgements in which a complex word is labeled as such by the system (TP), and all the judgements in which all non-complex words are correctly labeled with this category (TN). In this metric, the three runs obtained also the three

Table 6
Results by 5 participants and 9 methods submitted to ALexS 2020

Participants and methods	Accuracy	Precision	Recall	F1
Rico-Sulayes, run 2	**0.98**	**0.34**	0.23	**0.27**
Rico-Sulayes, run 1	**0.98**	0.33	0.22	0.26
Rico-Sulayes, run 3	**0.98**	0.33	0.22	0.26
AlexS 2020 Organizers	0.92	0.12	0.66	0.2
Zotova, run 1, run 1	0.91	0.1	0.6	0.17
Zotova, run 3, run 3	0.91	0.1	0.59	0.17
Zotova, run 2, run 2	0.89	0.09	0.69	0.16
Alarcón, run 1	0.9	0.09	0.67	0.16
Zaharia, run 3	0.91	0.02	0.08	0.03

best scores. Table 6 also shows that the system configuration that produced run 2 achieved the highest precision score. It should also be mentioned that, regarding the system developed here, there is quite a room for improvement in terms of recall scores, as they are comparatively low.

4.1. Conclusions and future work

Despite the fact that there have been critiques against lexicon-based classification [4], the figures reported in the last section show the potential of this technique, especially when it is used along its various possible extensions. One important observation about the system performance is the fact that it was aimed at retrieving only 723 tokens. When the golden standard was released, it became clear that the target words were actually types and not tokens. Along with this incorrect assumption in the design of the system, the final number of words used in the evaluation was 1,094. This number is the result of adding TP an TN in Table 5 (248+846=1,094). Given this final number of all true assignments, the 723 tokens targeted by the system resulted in a relatively low recall score. However, this also resulted in very competitive precision figures, as it has been shown in the former section. At the end, the system proved to be very robust in this kind of competition where no information was provided about the tagging process and the final golden standard format. This means that the system has a great potential to maintain its performance and deal with "in the wild conditions", to burrow a common term from the facial recognition community [15] that refers to the testing of systems using non-controlled, unconstrained data collection techniques.

As the golden standard has been released, and in accordance with the suggestions of the reviewers, an analysis of the errors and a stratified evaluation of the system modules has been planned. This stratified evaluation should explore the contribution of the different modules: the general lexicon dictionary, its three extensions – proper names, verb conjugations, and specialized Internet-related lexicon –, the frequent n-grams, and the document-based normalized frequency filter. However, this analysis should require a much longer discussion than the one presented in the current article. Due to space constraints, and also because this discussion is beyond the current paper scope, an error analysis, a stratified evaluation, and some further improvement to the system are expected to appear in a future work currently under preparation.

Acknowledgments

The development of the system presented in this article was carried out using equipment purchased with a grant from the Academic Dean's Office at Universidad de las Americas Puebla.

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. URL: https://www.aclweb.org/anthology/P13-3015.

[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. URL: https://www.aclweb.org/anthology/S16-1085. 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. URL: https://www.aclweb.org/anthology/W18-0507. doi:10.18653/v1/W18-0507.

[4] J. Eisenstein, Unsupervised learning for lexicon-based classification, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI'17, AAAI Press, 2017, p. 3188–3194.

[5] A. García, S. Gaines, M. T. Linaza, A lexicon based sentiment analysis retrieval system for tourism domain, e-Review of Tourism Research (eRTR) (2012) 35–38.

[6] A. Jurek-Loughrey, M. Mulvenna, Y. Bi, Improved lexicon-based sentiment analysis for social media analytics, Security Informatics 4 (2015). doi:10.1186/s13388-015-0024-x.

[7] J. Ortiz Zambrano, A. Montejo Ráez, Alexs 2020: Lexicon analysis task at sepln, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF-2020), CEUR-WS, Malaga, Spain, 2020.

[8] CREA homepage, 2020. URL: http://corpus.rae.es/creanet.html.

[9] Glosario de redes sociales, 2020. URL: https://rockcontent.com/es/blog/glosario-de-redes-sociales/.

[10] Wikipedia, categoría: Nombres masculinos, 2020. URL: https://es.wikipedia.org/wiki/Categor%C3%ADa:Nombres_masculinos/.

[11] Wikipedia, categoría: Nombres femeninos, 2020. URL: https://es.wikipedia.org/wiki/Categor%C3%ADa:Nombres_femeninos/.

[12] Mongabay homepage, 2020. URL: https://names.mongabay.com/.

[13] DLE homepage, 2020. URL: https://dle.rae.es/.

[14] I. Guyon, L. Sun-Hosoya, M. Boullé, H. J. Escalante, S. Escalera, Z. Liu, D. Jajetic, B. Ray, M. Saeed, M. Sebag, A. Statnikov, W.-W. Tu, E. Viegas, Analysis of the automl challenge series 2015–2018, in: F. Hutter, L. Kotthoff, J. Vanschoren (Eds.), Automated Machine Learning: Methods, Systems, Challenges, Springer International Pub-

lishing, Cham, 2019, pp. 177–219. URL: https://doi.org/10.1007/978-3-030-05318-5_10. doi:10.1007/978-3-030-05318-5_10.

[15] A. A. Salah, H. Kaya, F. Gürpınar, Chapter 17 - video-based emotion recognition in the wild, in: X. Alameda-Pineda, E. Ricci, N. Sebe (Eds.), Multimodal Behavior Analysis in the Wild, Computer Vision and Pattern Recognition, Academic Press, 2019, pp. 369 – 386. URL: http://www.sciencedirect.com/science/article/pii/B9780128146019000316. doi:https://doi.org/10.1016/B978-0-12-814601-9.00031-6.