Alejandro Mosquera at SemEval-2021 Task 1: Exploring Sentence and Word Features for Lexical Complexity Prediction

Alejandro Mosquera
Symantec Enterprise Division
Broadcom Corporation
alejandro.mosquera@broadcom.com

Abstract

This paper revisits feature engineering approaches for predicting the complexity level of English words in a particular context using regression techniques. Our best submission to the Lexical Complexity Prediction (LCP) shared task was ranked 3rd out of 48 systems for sub-task 1 and achieved Pearson correlation coefficients of 0.779 and 0.809 for single words and multi-word expressions respectively. The conclusion is that a combination of lexical, contextual and semantic features can still produce strong baselines when compared against human judgement.

1 Introduction

Lexical complexity is a factor usually linked to poor reading comprehension (Dubay, 2004) and the development of language barriers for target reader groups such as second language learners (Saquete et al., 2013) or native speakers with low literacy levels, effectively making texts less accessible (Rello et al., 2013). For this reason, complex word identification (CWI) is often an important sub-task in several human language technologies such as text simplification (Siddharthan, 2004) or readability assessment (Collins-Thompson, 2014).

The Lexical Complexity Prediction (LCP) shared task of Semeval-2021 (Shardlow et al., 2021) proposes the evaluation of CWI systems by predicting the complexity value of English words in context. LCP is divided into two sub-tasks: Sub-task 1, predicting the complexity score for single words; Sub-task 2: predicting the complexity score for multi-word expressions. In our participation in both sub-tasks, we treat the identification of complex words as a regression problem, where each word is given a score between 1 and 5, given the sentence in which it occurs. In order to do this, we have evaluated sub-sets of word and sentence features against different machine learning models. Our best submissions achieved Pearson correlation coefficients of 0.779 and 0.809 for single words and multi-word expressions respectively.

In Section 2 we review related work for this task. Section 3 and 4 introduce the data and feature engineering approaches respectively. In Section 5 the performance of different machine learning models is analysed. In Section 6 we present the obtained results. Finally, in Section 7 we draw our conclusions and outline future work.

2 Related Work

Previous CWI studies applied to the English language have relied mostly on word frequencies, psycholinguistic information (Devlin and Tait, 1998), lexicons and other word-based features such as number of characters or syllable counts (Shardlow, 2013), which considered in most cases the target word in isolation. In order to address the limitations of word-level approaches more recent work made use of contextual and sentence information such as measuring the complexity of word n-grams (Ligozat et al., 2012), applying language models (Maddela and Xu, 2018) or treating the problem as a sequence labelling task (Gooding and Kochmar, 2019).

In this paper, we not only evaluate many of the traditional word-based features found in the literature but we also pay attention to the context surrounding the target by generating additional bigram and sentence features. In the end, we demonstrate that a careful selection of simple features is still competitive against more novel approaches for this task.

3 Datasets

CompLex (Shardlow et al., 2020), which was the official dataset provided by the organizers, contains complexity annotations using a 5-point Likert scale

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 554–559
Bangkok, Thailand (online), August 5–6, 2021.

for 7,662 words and 1,517 multi-word expressions (MWE) from three domains: the Bible, Europarl, and biomedical texts.

External datasets and models are historically allowed and used in SemEval as a way of complementing the original training set. Likewise, based on previous experiences, external resources can also correlate better with the evaluation labels than the official task resources in certain scenarios (Mosquera, 2020). For this reason, related datasets from previous CWI shared tasks such as CWI 2016 (Paetzold and Specia, 2016) and CWI 2018 (Štajner et al., 2018) were considered and evaluated as both extra training data and for deriving additional features. However, the performance of our models during the validation step not only didn't improve but worsened when attempting to use them.

4 Feature Engineering

The 51 features used in order to detect the complexity of single words and each component of MWEs are as follows:

Word length (word_len): The length in characters of the target word.

Syllable count (syl_count): Target word syllable count.

Morpheme length (morpheme_len): Number of morphemes for the target word.

Google frequency (google_freq): The frequency of the target word based on a subset Google ngram corpus [1].

Wikipedia word frequency (wiki_freq1): The frequency of the target word based on Wikipedia [2].

Wikipedia document frequency (wiki_freq2): The number of documents in Wikpedia where the target word appears.

Complexity score (comp_lex): Complexity score for the target word from a complexity lexicon (Maddela and Xu, 2018).

Number of morphemes (morpheme_len): Number of morphemes in the target word.

Zipf frequency (zip_freq): The frequency of the target word in Zipf-scale as provided by the wordfreq (Speer et al., 2018) Python library.

Kucera-Francis word (kucera_francis): Kucera-Francis (Kucera et al., 1967) frequency of the target word.

Kucera-Francis lemma (st_kucera_francis): Kucera-Francis (Kucera et al., 1967) frequency of the target word lemma.

Is stopword (stop): True if the target word is an stopword.

Is acronym (acro): Heuristic that is set to True if the target word is a potential acronym based on simple casing rules.

Average age of acquisition (age): At what age the target word is most likely to enter someone's vocabulary (Kuperman et al., 2012).

Average concreteness (concrete): Concreteness rating for the target word (Brysbaert et al., 2014).

Lemma length (lemma_len): Lemma length of the target word.

Word frequency (COCA) (word_freq): Frequency of the target word based on the COCA corpus (Davies, 2008).

Lemma frequency (COCA) (lemma_freq): Frequency of the lemmatized target word based on the COCA corpus (Davies, 2008).

(consonant_freq): Frequency of consonants in the target word.

Number word senses (wn_senses): Number of senses of the target word extracted from WordNet (Fellbaum, 2010).

Number of synonyms (synonyms): Number of synonyms of the target word from WordNet.

Number of hypernyms (hypernyms): Number of hypernyms of the target word from WordNet.

Number of hyponyms (hyponyms): Number of hyponyms of the target word from WordNet.

WordNet min-depth (wn_mindepth): Minimum distance to the root hypernym in WordNet for the target word.

WordNet max-depth (wn_maxdepth): Maximum distance to the root hypernym in WordNet for the target word.

Number of Greek or Latin affixes (greek_or_latin_affix): True if the target word contains Greek or Latin affixes [3].

Bing frequency (bing_counts): The frequency of the target word based on the Bing n-gram corpus (Wang et al., 2010).

Bi-gram frequency (ph_mc2): Bi-gram frequency for the target and its preceding word in Google Books Ngram Dataset obtained via the phrasefinder API [4].

[1] https://github.com/hackerb9/gwordlist

[2] https://github.com/alex-pro-dev/english-words-by-frequency/blob/master/wikipedia_words.zip

[3] https://github.com/sheffieldnlp/cwi

[4] https://phrasefinder.io/api

Volume count (ph_vc2): The number of books where the target and its preceding word appeared in the Google Books Ngram Dataset obtained via the phrasefinder API.

Year of appearance (ph_fy2): The first year where the target and its preceding word appeared in the Google Books Ngram Dataset obtained via the phrasefinder API.

SUBTLEX-US frequency (FREQcount): Target word frequency based on SUBTLEX-US corpus (Brysbaert et al., 2012).

SUBTLEX-US number of films (CDcount): Number of films in which the target word appears in the SUBTLEX-US corpus.

SUBTLEX-US frequency lowercase (FREQlow): Number of times the target word appears in the SUBTLEX-US corpus starting with a lowercase letter

SUBTLEX-US number of films lowercase (Cdlow): Number of films in which the target word appears starting with a lower-case letter in the SUBTLEX-US corpus.

SUBTLEX-US frequency per million (SUBTLWF): Target word frequency per million words in the SUBTLEX-US corpus.

SUBTLEX-US log frequency (Lg10WF): The base-10 logarithm of the absolute frequency of the target word plus one in the SUBTLEX-US corpus.

SUBTLEX-US percentage of films (SUBTLCD): The percentage of the films where the target word appears in the SUBTLEX-US corpus.

SUBTLEX-US log number of films (Lg10CD): The base-10 logarithm of the number of films in which the target word appears in the SUBTLEX-US corpus.

SUBTLEX-UK frequency (LogFreqZipf): The base-10 logarithm of the target word frequency in Zipf-scale for the SUBTLEX-UK corpus (Van Heuven et al., 2014).

SUBTLEX-UK Cbeebies frequency (LogFreqCbeebiesZipf): The base-10 logarithm of the target word frequency in Zipf-scale for the Cbeebies subset of SUBTLEX-UK corpus

SUBTLEX-UK CBBC frequency (LogFreqCBBCZipf): The base-10 logarithm of the target word frequency in Zipf-scale for the CBBC subset of SUBTLEX-UK corpus

SUBTLEX-UK BNC frequency (LogFreqBNCZipf): The base-10 logarithm of the target word frequency in Zipf-scale for the BNC subset of SUBTLEX-UK corpus

ANC word frequency (anc): Frequency of the target word based on the American National Corpus (ANC) (Ide and Macleod, 2001).

Kincaid grade level (sentence_Kincaid): Kincaid grade level of the whole sentence.

ARI score (sentence_ARI): Automated readability index (Senter and Smith, 1967) of the whole sentence.

Coleman-Liau score (sentence_Coleman-Liau): Coleman-Liau readability score (Coleman and Liau, 1975) of the whole sentence.

Flesch score (sentence_FleschReadingEase): Flesh reading ease score (Flesch, 1948) of the whole sentence.

Gunning-Fog (sentence_GunningFogIndex): Gunning-Fog readability index (Gunning et al., 1952) of the whole sentence.

LIX score (sentence_LIX): LIX readability score (Anderson, 1983) of the whole sentence.

SMOG index (sentence_SMOGIndex): SMOG readability index (Mc Laughlin, 1969) of the whole sentence.

RIX score (sentence_RIX): RIX readability score (Anderson, 1983) of the whole sentence.

Dale-Chall index (sentence_DaleChallIndex): Dale-Chall readability index (Chall and Dale, 1995) of the whole sentence.

All the readability features were calculated using the readability Python library [5].

5 Machine Learning Approach

Since the labels in the training dataset were continuous we have modelled both sub-tasks as regression problems. For sub-task 1, we made use of LightGBM (LGB) (Ke et al., 2017) implementation of gradient tree boosting. Minimal hyper-parameter optimization was performed against our development set, using a 0.01 learning rate and limiting the number of leaves of each tree to 30 over 500 boosting iterations.

For sub-task 2, the complexity score of each MWE component was obtained by using a linear regression (LR) model and averaged with equal weights.

By examining the feature importance for both the LGB model in Figure 2 and the LR model in Figure 3 we can observe several sentence readability features being identified as top contributors. While some degree of correlation between the complexity of the sentence and the target word was expect

[5] https://github.com/andreasvc/readability

a priori, a machine learning model can also use sentence-level complexity as a predictor of formality and genre (Mosquera and Moreda, 2011), thus being able to differentiate between the different sub-corpora present in the training data as seen in Figure 1 .

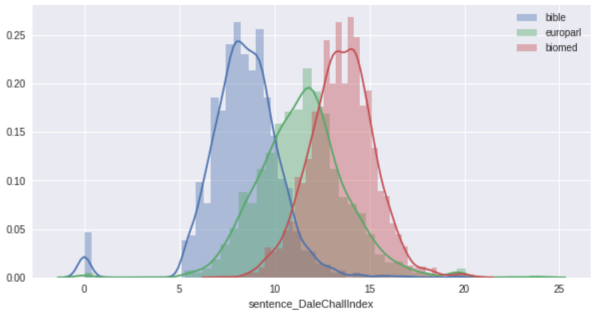

Figure 1: Example of differences in readability across the CompLex sub-corpora as measured by the Dale-Chall index.

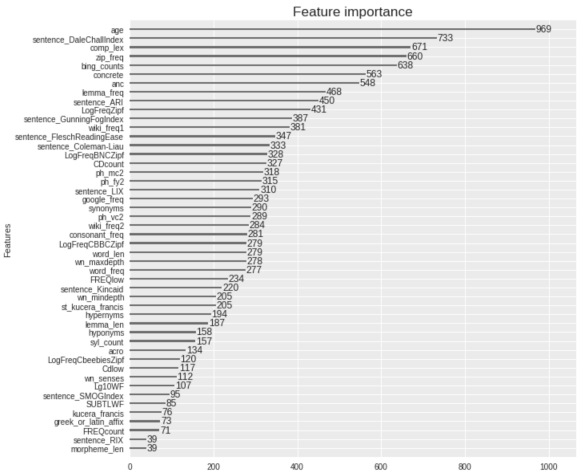

Figure 2: LGB feature importance as the number of times the feature is used in the model.

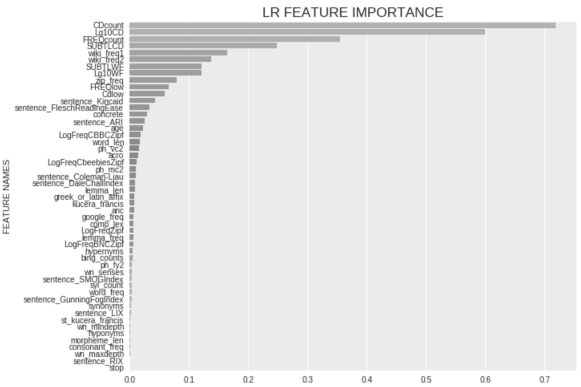

Figure 3: Linear regression weights.

Model	Dev	Trial	Test
Sub-task 1 LR	0.789	0.798	0.760
Sub-task 1 RF	0.792	0.824	0.766
Sub-task 1 LGB	0.801	0.841	**0.779**
Sub-task 2 LR	0.771	0.780	**0.809**
Sub-task 1 winner			**0.788**
Sub-task 2 winner			**0.861**

Table 1: Performance comparison of different models for each sub-task and evaluation set using Pearson's r.

6 Results

For sub-task 1, we have evaluated the performance of both linear and tree ensembles using the provided trial set and a randomly selected holdout with 30% of the training data as development set. The best performing model was gradient boosting. See Table 1.

7 Conclusion and Future Work

In this paper, we present the system developed for the Lexical Complexity Prediction task of SemEval 2021. Even though most of the features we made use of are relatively common in previous works, we demonstrate that a careful selection of lexical, contextual and semantic features at both target word and sentence level can still produce competitive results for this task. In a future work we would like to explore different neural network architectures and automated machine learning (AutoML) approaches.

References

Jonathan Anderson. 1983. Lix and rix: Variations on a little-known readability index. *Journal of Reading*, 26(6):490–496.

Marc Brysbaert, Boris New, and Emmanuel Keuleers. 2012. Adding part-of-speech information to the subtlex-us word frequencies. *Behavior research methods*, 44(4):991–997.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911.

Jeanne Sternlicht Chall and Edgar Dale. 1995. *Readability revisited: The new Dale-Chall readability formula*. Brookline Books.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

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

Mark Davies. 2008. The corpus of contemporary american english: 450 million words, 1990-present.

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.

William H. Dubay. 2004. The principles of readability. *Costa Mesa, CA: Impact Information*.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.

Sian Gooding and Ekaterina Kochmar. 2019. Complex word identification as a sequence labelling task. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1148–1153, Florence, Italy. Association for Computational Linguistics.

Robert Gunning et al. 1952. Technique of clear writing.

Nancy Ide and Catherine Macleod. 2001. The american national corpus: A standardized resource of american english. In *Proceedings of corpus linguistics*, volume 3, pages 1–7. Citeseer.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3146–3154. Curran Associates, Inc.

Henry Kucera, Henry Kučera, and Winthrop Nelson Francis. 1967. *Computational analysis of present-day American English*. University Press of New England.

Victor Kuperman, Hans Stadthagen-Gonzalez, and Marc Brysbaert. 2012. Age-of-acquisition ratings for 30,000 english words. *Behavior research methods*, 44(4):978–990.

Anne-Laure Ligozat, Cyril Grouin, Anne Garcia-Fernandez, and Delphine Bernhard. 2012. ANNLOR: A naïve notation-system for lexical outputs ranking. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 487–492, Montréal, Canada. Association for Computational Linguistics.

Mounica Maddela and Wei Xu. 2018. A word-complexity lexicon and a neural readability ranking model for lexical simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3749–3760, Brussels, Belgium. Association for Computational Linguistics.

G. Harry Mc Laughlin. 1969. Smog grading-a new readability formula. *Journal of reading*, 12(8):639–646.

Alejandro Mosquera. 2020. Amsqr at SemEval-2020 task 12: Offensive language detection using neural networks and anti-adversarial features. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1898–1905, Barcelona (online). International Committee for Computational Linguistics.

Alejandro Mosquera and Paloma Moreda. 2011. The use of metrics for measuring informality levels in web 2.0 texts. In *Proceedings of the 8th Brazilian Symposium in Information and Human Language Technology*.

Gustavo Paetzold and Lucia Specia. 2016. Semeval 2016 task 11: Complex word identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*.

Luz Rello, Ricardo Baeza-Yates, and Horacio Saggion. 2013. The impact of lexical simplification by verbal paraphrases for people with and without dyslexia. In *Proceedings of the 14th International Conference on Computational Linguistics and Intelligent Text Processing - Volume 2*, CICLing'13, page 501–512, Berlin, Heidelberg. Springer-Verlag.

Estela Saquete, Sonia Vazquez, Elena Lloret, Fernando Llopis, Jose M. Gomez-Soriano, and Alejandro Mosquera. 2013. Improving reading comprehension of educational texts by simplification of language barriers. In *EDULEARN13 Proceedings*, 5th International Conference on Education and New Learning Technologies, pages 3784–3792. IATED.

R.J. Senter and Edgar A. Smith. 1967. Automated readability index. Technical report, CINCINNATI UNIV OH.

Matthew Shardlow. 2013. 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*, pages 103–109, Sofia, Bulgaria. Association for Computational Linguistics.

Matthew Shardlow, Michael Cooper, and Marcos Zampieri. 2020. Complex: A new corpus for lexical complexity predicition from likert scale data. In *Proceedings of the 1st Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI)*.

Matthew Shardlow, Richard Evans, Gustavo Paetzold, and Marcos Zampieri. 2021. Semeval-2021 task 1: Lexical complexity prediction. In *Proceedings of the 14th International Workshop on Semantic Evaluation (SemEval-2021)*.

Advaith Siddharthan. 2004. Syntactic simplification and text cohesion. *Research on Language & Computation*, 4.

Robyn Speer, Joshua Chin, Andrew Lin, Sara Jewett, and Lance Nathan. 2018. Luminosoinsight/wordfreq: v2.2.

Walter J.B. Van Heuven, Pawel Mandera, Emmanuel Keuleers, and Marc Brysbaert. 2014. Subtlex-uk: A new and improved word frequency database for british english. *Quarterly journal of experimental psychology*, 67(6):1176–1190.

Sanja Štajner, Chris Biemann, Shervin Malmasi, Gustavo Paetzold, Lucia Specia, Anaïs Tack, Seid Muhie Yimam, 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*.

Kuansan Wang, Chris Thrasher, Evelyne Viegas, Xiaolong(Shiao-Long) Li, and Bo-June (Paul) Hsu. 2010. An overview of microsoft web n-gram corpus and applications.