UNBNLP at SemEval-2021 Task 1: Predicting lexical complexity with masked language models and character-level encoders

Milton King and **Ali Hakimi Parizi** and **Samin Fakharian** and **Paul Cook**
Faculty of Computer Science, University of New Brunswick
Fredericton, NB E3B 5A3, Canada
{milton.king,ahakimi,samin.fakharian,paul.cook}@unb.ca

Abstract

In this paper, we present three supervised systems for English lexical complexity prediction of single and multiword expressions for SemEval-2021 Task 1. We explore the use of statistical baseline features, masked language models, and character-level encoders to predict the complexity of a target token in context. Our best system combines information from these three sources. The results indicate that information from masked language models and character-level encoders can be combined to improve lexical complexity prediction.

1 Introduction

SemEval 2021 Task 1 (Shardlow et al., 2021) focuses on predicting the complexity of a target token in an English sentence on a scale from 0 (not very complex) to 1 (very complex). For example, the token *land* in the example below is judged to have a low complexity of 0.19.

1. Our land will yield its increase.

On the other hand, the token *doxycycline* in the following example is judged to have a relatively higher complexity of 0.75.[1]

2. The reason these two lines were unresponsive to doxycycline is unknown.

The dataset for this task was originally proposed in Shardlow et al. (2020), and includes sentences from three sources: a translated bible, European Parliament proceedings, and a biomedical corpus. This shared task contains two sub-tasks with the first focusing on lexical complexity for single words, which will be referred to as *SINGLE*, and the second focusing on complexity of multiword expressions, which will be referred to as *MULTI*.

In this paper, we explore the use of statistical baseline features, masked language models, and character-level encoders to predict the complexity of a target token in context. We first consider these approaches individually and then consider supervised methods for combining them. We evaluate our models with Pearson correlation (R), Spearman correlation (Rho), mean absolute error (MAE), mean square error (MSE), and R-squared (R2). We apply all our models to both sub-tasks and find that we achieve our best results with a model that combines all three sources of information — baseline features, masked language modeling, and character-level encoding. Specifically, we achieve our best results with respect to Pearson correlation — the evaluation measure used for the official shared task system ranking — using a model that combines complexity predictions from two approaches, one based on a character-level encoder, and the other based on a masked language model, which further incorporates the baseline features, using support vector regression. Although we achieve our best results using this approach on both *SINGLE* and *MULTI* with respect to Pearson correlation, we note substantial variation in performance with respect to the other evaluation metrics on *SINGLE*. This suggests that future work could further explore the variation in performance of this model on single and multiword expressions

2 Model components

In this section, we describe three approaches to lexical complexity prediction. In Section 3 we then describe how these approaches are combined into systems that we submitted as official runs to the shared task.

[1] These example sentences and complexity scores are taken from the dataset provided for the shared task.

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 650–654
Bangkok, Thailand (online), August 5–6, 2021.

2.1 Baseline features

We extract a set of seven statistical baseline features. These features include the length of the target token, the frequency of the token in SubtlexUS (Brysbaert and New, 2009), the frequency of the token in SubtlexUK (van Heuven et al., 2014), a binary feature indicating whether the token is an MWE, and a binary feature for each text type the instances were taken from (i.e., biomedical text, European Parliament proceedings, and the bible) indicating whether the instance is from that corpus. For MWEs, the frequency features are calculated as the average of the frequencies of the component words in the MWE. In the case that frequency information is not available for a word in SubtlexUS or SubtlexUK, the frequency of that word is set to the average frequency for words in the dataset for which frequency information is available. These features are combined with the approach described in Section 2.3, and the systems presented in Section 3.

2.2 Character-level encoder

This approach is based on a character-level encoder, which has three parts. The first part is a pre-trained character-level language model which gets a sentence containing a target expression as its input and its last hidden state is used as an embedding representing the input sentence. This language model uses a bi-directional GRU with a hidden layer size of 256. It is trained on the sentences in the trial and training data provided for the shared task.

The second part is a similar pre-trained GRU bi-directional character-level language model, which receives the target word as its input and the first hidden state is initialized with the embedding of the input sentence. The last hidden state is then passed to the systems described in Section 3 for complexity prediction.

In this approach, our hypothesis is that the hidden state of the first language model provides a representation for the input sentence, and the language model can encode the complexity of the target word by having access to a representation of the sentence in which the target word appears. Figure 1 shows a diagram of this model.

2.3 Masked language model

In this approach, we recruit BERT (Devlin et al., 2019) as a masked language model to estimate the probability for the target token in context. Collins-Thompson and Callan (2004) show that language modeling can be used to predict reading difficulty, and therefore we hypothesize that language modeling can also be useful for predicting lexical complexity.

We use the large uncased pretrained BERT model, which consists of 16 heads and 24 layers of 1024 hidden units each. Given a sentence, we replace the target token with the special *[MASK]* token and use the modified sentence as input to BERT to obtain the probability of the target token. BERT's tokenizer can split tokens into multiple pieces. The probability of a target token (single word or multiword expression) is therefore calculated by averaging the probabilities of its parts.

The probability of the target is then used as a feature, alongside the baseline features from Section 2.1, in a support vector regressor (SVR) to predict complexity. The SVR uses an rbf kernel, with a kernel coefficient of 0.1, epsilon of 0.1, and a regularization of $1/100$ The output of the SVR is used as a feature in the systems described in Section 3. Figure 2 shows a diagram of this model.

3 Submitted Systems

In this section, we describe how we combine the approaches discussed in Section 2 to form systems that were submitted as official runs to the shared task. We selected these systems because they achieved the best performance in a ten-fold cross-validation experiment on the combined trial and training data, in terms of Pearson correlation (R), the evaluation metric used to rank systems in the shared task. The performance of each submitted system, a baseline in which we train logistic regression on the baseline features from Section 2.1, and the masked language model approach described in Section 2.3 (our best-performing model that was not submitted to the shared task), are shown in Tables 1 and 2 for *SINGLE* and *MULTI*, respectively.

3.1 System 1

In this system, we concatenate the output of the character-level encoder approach discussed in Section 2.2 with the baseline features from Section 2.1. This representation is then used as input to a feed-forward network to predict the complexity. The feed-forward network has two fully connected hidden layers with sizes 128 and 64, and ReLU activation functions. We train this network using Adam optimizer (Kingma and Ba, 2015) with a

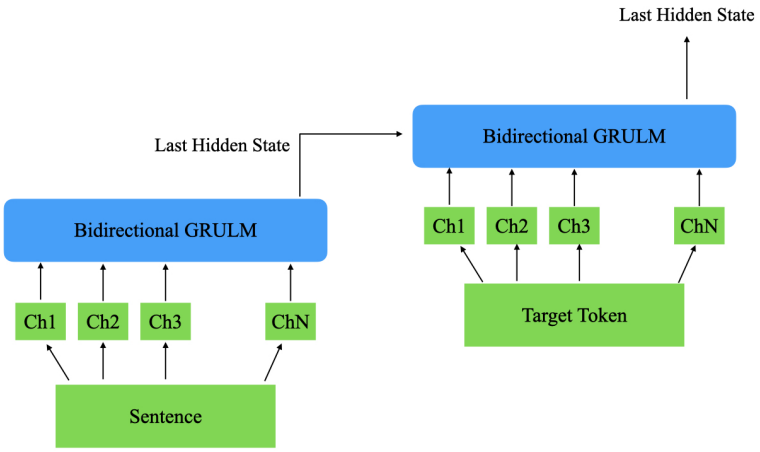

Figure 1: The character-level encoder model.

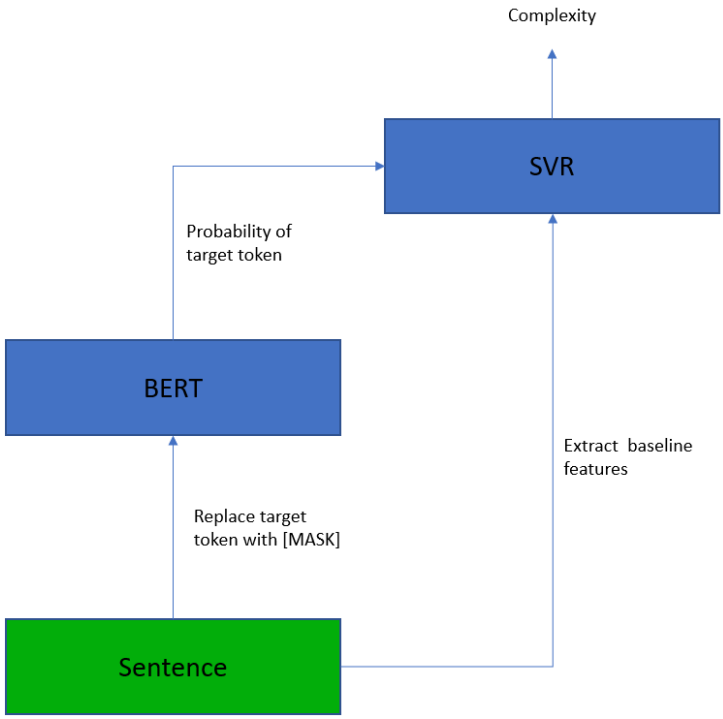

Figure 2: A diagram of the masked language model.

Model	R	Rho	MAE	MSE	R2
Baseline	0.618	0.633	0.080	0.011	0.378
MLM	0.622	0.635	0.080	0.011	0.383
System$_1$	0.698	0.673	0.074	0.009	0.476
System$_2$	0.712	0.680	0.072	0.009	0.499
System$_3$	0.705	0.678	0.073	0.009	0.482

Table 1: Results for each model for each evaluation metric on single word expressions in the validation set. *MLM* is the masked language model described in Section 2.3. (For R, Rho, and R2 bigger numbers indicate better performance, whereas for MSE and MAE, smaller numbers are better.)

Model	R	Rho	MAE	MSE	R2
Baseline	0.665	0.683	0.094	0.015	0.412
MLM	0.665	0.684	0.094	0.015	0.411
System$_1$	0.675	0.686	0.092	0.014	0.440
System$_2$	0.684	0.693	0.091	0.014	0.442
System$_3$	0.673	0.684	0.091	0.014	0.432

Table 2: Results for each model for each evaluation metric on multiword expressions in the validation set.

learning rate of 0.001 for 100 epochs. This system is referred to as *System*$_1$ from heron.

3.2 System 2

In this system, we concatenate the feature from the masked language model approach described in Section 2.3 with the output from the system described in Section 3.1 to create a 2-dimensional vector. This vector is then used as input to an SVR to predict complexity.

We perform a grid search to tune the hyperparameters of the SVR via evaluating its performance on the validation set. We achieve our best results with an SVR using a polynomial kernel with a degree of 3, a kernel coefficient of 1, a regularization parameter of 100, a stopping tolerance of 1, and a gamma value of $1/$*number of training instances*. We use these parameters for all further experiments with this system, which we refer to *System*$_2$.

3.3 System 3

In this system, similar to *System*$_1$, we concatenate the output of several approaches from Section 2, and then pass this representation to a fully connected network to predict the complexity. Here we concatenate the baseline features with the output of both the character-level encoder approach (Section 2.2) and the masked language model approach (Section 2.3). We use the same fully-connected network structure, and training settings, as for *System*$_1$. We refer to this approach as *System*$_3$.

Model	R	Rho	MAE	MSE	R2
Baseline	0.584	0.597	0.080	0.108	0.334
System_1	0.691	**0.656**	0.073	0.0094	0.418
System_2	**0.695**	0.654	0.072	0.0089	0.450
System_3	0.689	0.653	**0.069**	**0.0086**	**0.471**

Table 3: Results on *SINGLE* for each system and each evaluation metric. The best result for each evaluation metric is shown in boldface.

Model	R	Rho	MAE	MSE	R2
Baseline	0.731	0.704	0.092	0.013	0.470
System$_1$	0.741	0.735	0.0843	0.0116	0.519
System$_2$	**0.752**	**0.742**	**0.0802**	**0.0106**	**0.562**
System$_3$	0.736	0.730	0.0851	0.0116	0.521

Table 4: Results on *MULTI* for each system and each evaluation metric. The best result for each evaluation metric is shown in boldface.

4 Results

In this section we present our results with respect to the five evaluation metrics. We evaluate our models on the single word sub-task (*SINGLE*) first and then on the multiword expressions sub-task (*MULTI*). Each system is trained on all training instances from both *SINGLE* and *MULTI*.[2] We include the performance of our baseline to show that all submitted systems continue to outperform this baseline model on the test data.

In Table 3, we show the performance of our systems on *SINGLE*. *System*$_2$ achieves the best results with respect to R, which is the metric used to rank submissions in the shared task. Interestingly, however, this is the only metric for which *System*$_2$ outperforms the other systems. *System*$_3$ — which like *System*$_2$ incorporates information from both the character-level encoder and masked language model approaches — performs worst of these three systems with respect to R, but achieves the best performance amongst these systems for MSE, MAE, and R2.

In Table 4, we show the performance of our models on *MULTI*. In contrast to the results on *SINGLE*, these results show *System*$_2$ consistently performs

[2]The training data for the approach described in Section 2.3 also includes instances from the provided trial data from both *SINGLE* and *MULTI*.

best for all evaluation metrics. The improvement of $System_2$ — which uses information from both the character-level encoder and masked language model approaches — over $System_1$ in particular — which does not incorporate information from the masked language model — suggests that these two sources of information can be combined to improve lexical complexity prediction.

5 Conclusions

We evaluated three systems for lexical complexity prediction of single and multiword expressions for SemEval 2021 Task 1. These systems incorporated information from statistical baseline features, a character-level encoder approach, and a masked language model approach. We found that a system that combined the complexity predictions of the character-level encoder approach and the masked language model approach, which further incorporates the statistical baseline features, using support vector regression performed best amongst our submitted systems with respect to Pearson correlation on both the single word and multiword expressions sub-tasks. This approach further performed best of our submitted systems with respect to all evaluation metrics on the multiword expression sub-task, although this was not the case for the single word sub-task.

In future work, the relationship between the sub-tasks, models, and evaluation metrics warrants further exploration, including studying the effect that the type of the target expression, i.e., single word or multiword expression — has on the performance of the models with respect to the various evaluation metrics.

Acknowledgments

This work is financially supported by the Natural Sciences and Engineering Research Council of Canada and the University of New Brunswick.

References

M. Brysbaert and B. 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*, 41:977–990.

Kevyn Collins-Thompson and James P. Callan. 2004. A language modeling approach to predicting reading difficulty. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 193–200, Boston, Massachusetts, USA. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Walter van Heuven, Paweł Mandera, Emmanuel Keuleers, and Marc Brysbaert. 2014. Subtlex-uk: a new and improved word frequency database for british english. *Quarterly journal of experimental psychology (2006)*, 67.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Matthew Shardlow, Michael Cooper, and Marcos Zampieri. 2020. CompLex — a new corpus for lexical complexity prediction from Likert Scale data. In *Proceedings of the 1st Workshop on Tools and Resources to Empower People with REAding DIfficulties (READI)*, pages 57–62, Marseille, France. European Language Resources Association.

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)*.