Text Complexity DE Challenge 2022 Submission Description: Pairwise Regression for Complexity Prediction

Leander Girrbach
University of Tübingen
leander.girrbach@student.uni-tuebingen.de

Abstract

This paper describes our submission to the Text Complexity DE Challenge 2022 (Mohtaj et al., 2022). We evaluate a pairwise regression model that predicts the relative difference in complexity of two sentences, instead of predicting a complexity score from a single sentence. In consequence, the model returns samples of scores (as many as there are training sentences) instead of a point estimate. Due to an error in the submission, test set results are unavailable. However, we show by cross-validation that pairwise regression does not improve performance over standard regression models using sentence embeddings taken from pretrained language models as input. Furthermore, we do not find the distribution standard deviations to reflect differences in "uncertainty" of the model predictions in an useful way.

1 Introduction

This paper describes our submission to the Text Complexity DE Challenge 2022 (Mohtaj et al., 2022). The task is to predict the linguistic complexity of a given sentence. The task is defined as a regression task, where labels are $\in [1, 7]$. Labels are averaged human ratings, who rated the sentences for complexity, understandability, and lexical defficulty (see (Naderi et al., 2019a) for details). Only complexity labels are taken into account in this shared task. The train set consists of 1000 labelled sentences, the development set consists of 100 sentences, and the test set contains 210 sentences. Only the labels of training sentences where ever revealed to participants.

In this paper, we evaluate pairwise regression for complexity score prediction. Instead of predicting a single complexity score from a single sentence, we predict the relative difference in complexity of two sentences. In practise, this results in a distribution over complexity scores instead of a point estimate, because we predict the relative difference for each training sentence. However, further analysis reveals that pairwise regression neither performs better than standard regression nor does the standard deviation of score distributions contain useful information about model performance.

Furthermore, due to an erroneous submission we do not have test set score for this shared task. Therefore, all our analyses and observations are based on 10-fold cross-validation on the training data.

2 Related Work

Readability scoring of texts is has been researched for over a century. Research started by developing readability formulas based on surface features such as token counts or type-token ratios. Modern approaches use statistical methods, especially supervised learning, to learn readability models. Here, readability scoring can be defined both as a regression task (Naderi et al., 2019a; vor der Brück et al., 2008) and a classification task (Hancke et al., 2012; Weiss et al., 2021). Features usually rely on broad linguistic modelling (Weiß and Meurers, 2018; Naderi et al., 2019b).

Recently, deep neural networks have also been proposed for predicting readability labels (Martinc et al., 2021). Furthermore, the utility of linguistic features compared to deep representations was put into question by Deutsch et al. (2020). However, the main disadvantage of deep neural networks is their black-box nature. This is especially problematic, because practical applications of readability models generally require an especially high level of transparency, for example in an educational context (for giving feedback) or for essay scoring (where grades should be explainable and fair).

3 Method

In this section, we describe our approach at predicting the linguistic complexity of given sentences.

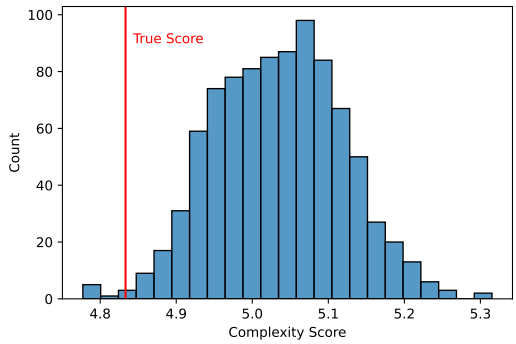

Figure 1: Distribution of scores predicted by a pairwise regression model for sentence "Infolge des gravitationsbedingten Auslaufens (Drainage) der zwischen den Seifenfilmoberflächen befindlichen Flüssigkeit dünnt eine Seifenblase in ihrem oberen Teil zunehmend aus."

We train deep learning models (described in Section 3.1) and also compare them to a traditional machine learning model based on linguistic features (see Section 3.2).

3.1 Pairwise Regression

Our main model is a deep learning model trained in a supervised fashion. Instead of directly predicting the complexity score from a single sentence, we use sentence pairs as inputs. Given a pair of sentences, we predict the difference in complexity of the sentences. At test time, after the model was trained, we predict the label of an unseen sentence by predicting the relative differences in difficulty to all sentences in the training set (in case of large training sets, taking a subset would also be possible). Because we know the true labels of train sentences, we use them to calculate an estimate of the complexity of the unseen sentence for every sentence in the training set. This gives us a sample of estimated complexity scores. We can arrive at a final estimate by taking the mean, or estimating the mode of the resulting distribution in a different way. An example of a predicted distribution produced by one of our models is in Figure 1.

Our main motivations for pairwise regression instead of single-sentence regression are: Given the data for this task is relatively small (1000 sentences in the train set), using sentence pairs is an easy way to increase the data set size. Furthermore, pairwise regression makes more use of the given data by treating sentences not only as isolated datapoints, but seeing them in relation to all other sentences in the dataset. Also, previous work (Lee and Vajjala, 2022; Weiss and Meurers, 2022) showed promising performance of pairwise readability ranking models. Therefore, we wanted to evaluate whether this also is true for a regression setting. In detail, or model is designed as follows:

Sentence Embedding First, we encode a sentence by 3 different openly available pretrained language models models:

- GottBERT (Scheible et al., 2020).[1] The sentence embeddings is calculated by averaging embeddings of all non-special tokens.
- dbmz's German BERT (cased) model.[2] The sentence embedding is simply the embedding of the "[CLS]" token.
- A multilingual sentence transformer model (Reimers and Gurevych, 2020).[3] We found German pretrained sentence transformers to not perform as well.

We concatenate all 3 embeddings to arrive at the final encoding of a sentence. Note that we do not fine-tune the pretrained models, but simply use them as feature extractors.

Prediction First, we transform each sentence separately (using the same model) by a MLP with 2 hidden layers and GELU activation. Then, we concatenate the transformed sentence embeddings and use a MLP with 1 hidden layer and GELU activation to predict the complexity difference. Optionally, we also predict the absolute complexity score of the input sentences. A visualisation of the model is shown in Figure 2.

Training Setup All models are implemented in PyTorch (Paszke et al., 2019). We train models for 6 epochs using batch size 32, dropout probability 0.3 (applied before every linear layer) and hidden sizes 300 and 600 (the first hidden layer of each MLP is twice the standard hidden size). In each of the 6 epochs, the model is trained on all combinations of sentences. Given the size of the present dataset, this is feasible, but in case of larger datasets sampling combinations is an option.

[1] https://huggingface.co/uklfr/gottbert-base

[2] https://huggingface.co/dbmdz/bert-base-german-cased

[3] https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1

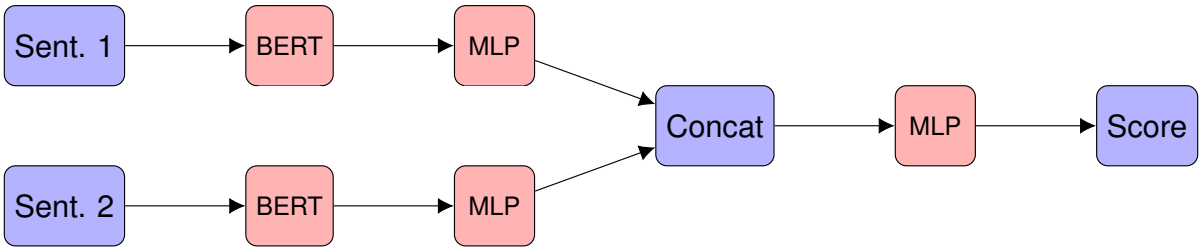

Figure 2: Flowchart showing how the pairwise regression model predicts relative complexity difference scores. Blue blocks are data and red blocks are neural networks.

The number of 6 epochs was found to work best by manual hyperparameter exploration. The optimizer is SGD with weight decay coefficient 1e-4. We set the learning rate according to a One-Cycle-Scheduler (Smith and Topin, 2019) with maximum learning rate 0.001. As regression loss, we use the smoothed L1 metric.

3.2 Baselines

In addition to the pairwise regression model described in Section 3.1, we evaluate 2 baselines:

One baseline is a random forest model trained on linguistic features extracted by CTAP (Chen and Meurers, 2016; Weiss et al., 2021).[4] We extract all features available for German. Then, we remove all features that resulted in NaN for at least 1 sentence, and we remove constant features. We train a random forest model using the scikit-learn implementation (Pedregosa et al., 2011) with the following hyperparameters: The number of trees is 450, the maximum percentage of features used for calculating splits is 85%, and both the minimum number of datapoints required for internal and leaf splits is 5.

Secondly, we train a simple (i.e. without pairwise regression) MLP regressor to predict complexity scores from single sentences. To be as comparable as possible to the pairwise regression model, we use the same hyperparameters. However, due to the different datasets, we need to change the number of epochs. We found 500 epochs to work best. Also, we evaluate all 3 pretrained language models as feature extractors and the combination of their sentence embeddings.

4 Results

Here, we present performance results of the pairwise regression model (see Section 3.1) and baselines (see Section 3.2). Unfortunately, we cannot present the shared task's test set scores due to an erroneous submission: Instead of submitting results on the real test set, we accidentally submitted results on a custom test set that we had created for internal evaluation. This error remained unnoticed until after the submission deadline. Therefore, we decide to report 10-fold cross-validation results on the training set, because we do also not have development set scores for all baselines and ablations.

Results for the pairwise regression model are in Table 1. Here, we can make 2 observations: Firstly, models perform similarly, but the best performing models only use GottBERT as sentence encoder. This suggests that the GottBERT model is, among the evaluated models, best at representing complexity-relevant features. Secondly, additionally predicting absolute complexity scores does not have a visible effect on the performance. Therefore, replacing absolute complexity score predictions by relative score predictions is possible.

In Figure 3, we show the loss curve for a pairwise regression model only predicting the relative difference and using all sentence embeddings. The curve shows that the loss starts to decrease quickly after about half an epoch. This may be an artifact of the initially very low learning rate due to the One-Cycle-Scheduler. After about 2 epochs, the loss only shows little improvements. This suggests that training for fewer epochs may already be sufficient. However, given that we did not observe better generalisation performance with shorter training, this may also suggest that the model is somewhat robust to longer training and still does not overfit the data.

Results for the baselines are in Table 2. The best performing model uses all 3 sentence embeddings and also yields the best overall results. This puts benefits of pairwise regression into question, since they apparently do not yield improvements in performance. However, neural models generally outperform the non-neural baseline, although the difference is not very large in absolute terms. Also,

[4] http://sifnos.sfs.uni-tuebingen.de/ctap/

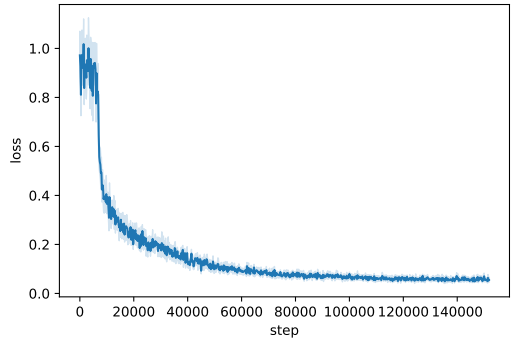

Figure 3: Loss curve for pairwise regression model (GottBERT + BERT + S-BERT + Only Δ). For each step, we display the loss mean and standard deviation (shaded area) calculated from the 10 cross-validation runs.

the non-neural baseline has the advantage of being interpretable to some degree. We would also like to note that the neural models outperform the results reported by Naderi et al. (2019b) and Weiss and Meurers (2022), who use a similar setup. Finally, we note that the model based on sentence transformers did not converge and would need more epochs. For the sake of comparability, we decide to still keep the setup the same for all baseline models.

5 Analysis

In Section 4, we have established that pairwise regression does not achieve better performance than direct prediction of absolute complexity scores. However, we are still interested in whether having a distribution of scores instead of a single score can provide additional insights. For example, it would be of advantage if we could use the score sample standard deviation to detect uncertain predictions, i.e. sentences where the model is not confident about the complexity. To be able to do this, the sample standard deviation has to correlate with the prediction error. This is, however, not the case: Figure 4 shows that while most errors are small, sentence score distributions that result in larger prediction errors do not have larger standard deviation. In fact, Pearson correlation is -0.18, however the negative value could be an artifact of the small number of large errors. Therefore, we conclude that the score distribution predicted by pairwise regression models does not provide further insights into the model predictions.

Finally, we also evaluate whether we can find linguistic features that are informative about which

Only Δ	GottBERT	BERT	S-BERT	RMSE
✓	✓	✓	✓	0.6270
✓	✓	✓		0.6333
✓	✓		✓	0.6130
✓	✓			0.6178
✓		✓	✓	0.6596
✓		✓		0.6830
✓			✓	0.6725
	✓	✓	✓	0.6315
	✓	✓		0.6375
	✓		✓	0.6188
	✓			0.6170
		✓	✓	0.6593
		✓		0.6769
			✓	0.6706

Table 1: Ablation results of various pairwise regression configurations (10-fold cross-validation on training set). "Only Δ" mean whether we only predict the relative score differences or also predict abolute scores. "GottBERT", "BERT", "S-BERT" are the different sentence embedding models described in Section 3.1.

Model	RMSE
GottBERT	0.6123
BERT	0.6639
S-BERT	1.1612
Combined	0.6068
Random Forest	0.6946

Table 2: RMSE results (10-fold cross-validation on training set) for baselines. "Combined" means representing sentences by concatenating sentence embeddings calculated by all 3 pretrained models. Random Forest uses linguistic features extracted by CTAP.

sentences are hard to score by the deep learning models. To evaluate this, we conduct another 10-fold cross-validation experiment using a Lasso model (scikit-learn implementation) to predict the squared error from linguistic features. However, the resulting R^2-score is only 0.02, which is barely better than always predicting the average. Therefore we conclude that linguistic features in this case cannot help detect sentences that are difficult for the deep models to score and we refrain from further analysing the importance of individual features.

6 Discussion

We evaluated pairwise regression in comparison to standard regression (predicting a single complexity score from a single sentence). Our results are

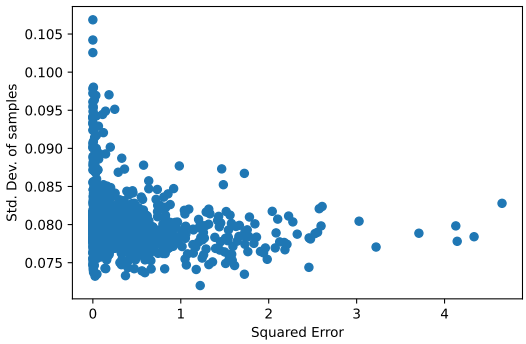

Figure 4: Scatter plot showing the relationship of standard deviation of score distributions predicted by a pairwise regression model (only Δ, all embeddings).

largely negative, showing that pairwise regression does not perform better than standard regression and the resulting score distribution does not seem to have additional use over the point estimates returned by standard regression. Furthermore, there seems to be no trend that can be captured by linguistic features about which sentences are more difficult to score by deep learning based models.

On the positive side, our evaluations show that pairwise regression and standard regression can be exchanged with only very little difference in prediction quality, and that deep learning based models perform somewhat better than models based on linguistic features.

Acknowledgements

We thank the organisers for organising this shared task.

References

Xiaobin Chen and Detmar Meurers. 2016. CTAP: A web-based tool supporting automatic complexity analysis. In *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, pages 113–119, Osaka, Japan. The COLING 2016 Organizing Committee.

Tovly Deutsch, Masoud Jasbi, and Stuart M. Shieber. 2020. Linguistic features for readability assessment. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@ACL 2020, Online, July 10, 2020*, pages 1–17. Association for Computational Linguistics.

Julia Hancke, Sowmya Vajjala, and Detmar Meurers. 2012. Readability classification for German using lexical, syntactic, and morphological features. In *Proceedings of COLING 2012*, pages 1063–1080, Mumbai, India. The COLING 2012 Organizing Committee.

Justin Lee and Sowmya Vajjala. 2022. A neural pairwise ranking model for readability assessment. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3802–3813, Dublin, Ireland. Association for Computational Linguistics.

Matej Martinc, Senja Pollak, and Marko Robnik-Sikonja. 2021. Supervised and unsupervised neural approaches to text readability. *Comput. Linguistics*, 47(1):141–179.

Salar Mohtaj, Babak Naderi, and Sebastian Möller. 2022. Overview of the GermEval 2022 shared task on text complexity assessment of german text. In *Proceedings of the GermEval 2022 Shared Task on Text Complexity Assessment of German Text*, Potsdam, Germany. Association for Computational Linguistics.

Babak Naderi, Salar Mohtaj, Kaspar Ensikat, and Sebastian Möller. 2019a. Subjective assessment of text complexity: A dataset for german language. *CoRR*, abs/1904.07733.

Babak Naderi, Salar Mohtaj, Karan Karan, and Sebastian Möller. 2019b. Automated text readability assessment for german language: A quality of experience approach. In *11th International Conference on Quality of Multimedia Experience QoMEX 2019, Berlin, Germany, June 5-7, 2019*, pages 1–3. IEEE.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

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, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. Gottbert: a pure german language model. *CoRR*, abs/2012.02110.

Leslie N Smith and Nicholay Topin. 2019. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE.

Tim vor der Brück, Sven Hartrumpf, and Hermann Helbig. 2008. A readability checker with supervised learning using deep indicators. *Informatica*, 32(4).

Zarah Weiss, Xiaobin Chen, and Detmar Meurers. 2021. Using broad linguistic complexity modeling for cross-lingual readability assessment. In *Proceedings of the 10th Workshop on NLP for Computer Assisted Language Learning*, pages 38–54, Online. LiU Electronic Press.

Zarah Weiß and Detmar Meurers. 2018. Modeling the readability of German targeting adults and children: An empirically broad analysis and its cross-corpus validation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 303–317, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Zarah Weiss and Detmar Meurers. 2022. Assessing sentence readability for German language learners with broad linguistic modeling or readability formulas: When do linguistic insights make a difference? In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 141–153, Seattle, Washington. Association for Computational Linguistics.