Tackling Data Drift with Adversarial Validation: An Application for German Text Complexity Estimation

Alejandro Mosquera
Broadcom Corporation / 1320 Ridder Park Drive San Jose, 95131 California, USA
alejandro.mosquera@broadcom.com

Abstract

This paper describes the winning approach in the first automated German text complexity assessment shared task as part of KONVENS 2022. To solve this difficult problem, the evaluated system relies on an ensemble of regression models that successfully combines both traditional feature engineering and pre-trained resources. Moreover, the use of adversarial validation is proposed as a method for countering the data drift identified during the development phase, thus helping to select relevant models and features and avoid leaderboard overfitting. The best submission reached 0.43 mapped RMSE on the test set during the final phase of the competition.

1 Introduction

Automatically assessing how easy to read a text is has many applications, ranging from text simplification for language learners and people with disabilities to customizing content for a particular audience. For this reason, the Natural Language Processing (NLP) research community have been organizing shared tasks and compiled linguistic resources aiming to solve this problem, not only in English but also for other languages.

The Text Complexity DE Challenge 2022 (Mohtaj et al., 2022) proposes the evaluation of systems able to predict the complexity of German texts by rating each sentence using the Mean Opinion Score (MOS), derived from annotations from a 7 point Likert-scale. In order to solve this problem and addressing the unexpected data drift between training and testing sets, meta-modeling and adversarial validation techniques were applied by combining predictions from multiple estimators via stacked generalization (Wolpert, 1992) and leveraging both traditional feature engineering and pre-trained resources. The best submission generated by the described approach and selected through adversarial validation won the competition by achieving the lowest mapped Root Mean Squared Error (RMSE) score [1].

This paper is organized as follows: First, related work is reviewed in Section 2. Next, Section 3 contains an analysis of the individual models and feature engineering approaches used for this task. In Section 4, model selection and adversarial validation strategies are discussed. Further on, the performance of the system and its components are detailed in Section 5. Finally, in Section 6 the author draws the main conclusions and outlines future work.

2 Related Work

The application of NLP techniques for automatic textual complexity assessment has received attention in several languages other than English (Quispesaravia et al., 2016; Finnimore et al., 2019; Forti et al., 2019), although in an smaller scale. Despite the differences between languages, the use of lexical, morphological and word list-derived features are also common in research works focused on German (Weiss et al., 2019). Likewise, related NLP applications such as readability assessment (Hancke et al., 2012) or evaluation of text simplification pipelines (Suter et al., 2016) demonstrated that similar approaches used to estimate the complexity of English texts could be suitable for German as well, although with some known shortcomings.

3 Methodology

The TextComplexityDE (Naderi et al., 2019) dataset that consists of 1000 sentences in German language taken from 23 Wikipedia articles was the only resource provided by the organizers. In order to solve the challenge, this dataset was used

[1] https://qulab.github.io/text_complexity_challlenge/

as training data following two main approaches: feature engineering based on morphological and lexical information (Mosquera, 2021) and transfer learning via pre-trained transformers. The regression models trained using these two different strategies and the methodology applied to combine their predictions are described in detail below.

3.1 Feature Engineering Models

Several lexical features were calculated from word stats extracted from dlexDB (Heister et al., 2011), SUBTLEX-DE (Brysbaert et al., 2011) and averaged for each text. Likewise, sentence-level metrics from Textstat [2] and Readability [3] Python libraries were also used. A description of all the word and sentence features is as follows (entries ending with an asterisk denote a feature group):

dlexDB

- **typ_syls_cnt**: number of syllables.
- **typ_freq_***: absolute / normalized / log absolute / log normalized / rank / rank123 corpus frequency.
- **typ_fam_***: absolute / normalized / log absolute / log normalized / rank / rank123 familiarity (Kennedy et al., 2002) (cumulative frequency of all words of the same length sharing the same initial trigram).
- **typ_inf_***: absolute / normalized / log absolute / log normalized / rank / rank123 regularity (Kennedy et al., 2002) (the number of words of the same length sharing the same initial trigram).
- **typ_div_con_***: absolute / normalized / log absolute / log normalized / rank / rank123 document frequency.
- **typ_div_sen_***: absolute / normalized / log absolute / log normalized / rank / rank123 sentence count.
- **typ_uniq_orth_strict_pos**: length of the shortest prefix uniquely identifying the word.
- **typ_uniq_orth_strict_neg**: negative offset for the last character of the shortest prefix uniquely identifying the word.
- **typ_uniq_lemma_strict_pos**: length of the shortest prefix uniquely identifying the lemmatized word.
- **typ_uniq_lemma_strict_neg**: negative offset for the last character of the shortest prefix uniquely identifying the lemmatized word.
- **typ_pia_avgcondprob_big**: average conditional probability of a word, based on an evaluation of all bigrams having this word as their second component.
- **typ_pia_avginfcont_big**: average information content of a word, based on an evaluation of all bigrams having this word as second component (Piantadosi et al., 2011).
- **typ_pia_avgcondprob_trig**: average conditional probability of a word, based on an evaluation of all triigrams having this word as their third component.
- **typ_pia_avginfcont_trigr**: average information content of a word, based on an evaluation of all trigrams having this word as third component (Piantadosi et al., 2011).
- **typ_cts_cumfreq_token_***: absolute / normalized / log absolute / log normalized / rank / rank123 cumulative corpus frequency of all character trigrams contained in the word.
- **typ_cts_cumfreq_type_***: absolute / normalized / log absolute / log normalized / rank / rank123 cumulative lexicon frequency of all character trigrams contained in the word.
- **typ_init_trigr_***: absolute / normalized / log absolute / log normalized / rank / rank123 cumulative frequency of all words sharing the same initial character trigram (Lima and Inhoff, 1985).
- **typ_nei_col_all_cnt_abs**: absolute number of orthographic neighbors (Coltheart, 1977).
- **typ_syls_cumfreq_token_***: absolute / normalized / log absolute / log normalized / rank / rank123 cumulative corpus frequency of all syllables contained in the word.
- **typ_syls_cumfreq_type_***: absolute / normalized / log absolute / log normalized / rank / rank123 cumulative lexicon frequency of all syllables contained in the word.

[2] https://pypi.org/project/textstat/
[3] https://pypi.org/project/readability/

SUBTLEX-DE

- **WFfreqcount**: target word frequency in the German subtitle corpus.
- **spell-check OK (1/0)**: 1 if the word had no spelling errors, 0 otherwise.
- **CUMfreqcount**: case-independent word frequency in the German subtitle corpus.
- **SUBTLEX**: frequency per million based on CUMfreqcount.
- **lgSUBTLEX**: log10(CUMfreqcount+1).
- **Google00**: word frequency based on Google 2000-2009 Books corpus.
- **Google00cum**: case-independent word frequency based on Google 2000-2009 Books corpus.
- **Google00pm**: Google frequency per million words.
- **lgGoogle00**: log10(Google00cum+1).

Sentence Readability

- **Kincaid**: Kincaid grade level.
- **ARI**: Automated readability index (Senter and Smith, 1967).
- **Coleman-Liau**: Coleman-Liau readability score (Coleman and Liau, 1975).
- **Flesch reading ease**: Flesh reading ease score (Flesch, 1948).
- **Gunning-Fog index**: Gunning-Fog readability index (Gunning et al., 1952).
- **LIX**: LIX readability score (Anderson, 1983).
- **SMOG index**: SMOG readability index (Mc Laughlin, 1969).
- **RIX**: RIX readability score (Anderson, 1983).
- **Dale-Chall index**: Dale-Chall readability index (Chall and Dale, 1995) of the whole sentence.
- **Wiener Sachtextformel**: grade level for German texts (Schulz et al., 1985)

Linear regression and gradient boosting models were trained with all the above features with default hyper-parameters. The 2 resulting estimators are referred across the paper as LR and LGB (Ke et al., 2017) respectively.

A list of the top 20 features in terms of minimum redundancy and maximum relevance (mRMR) (Ding and Peng, 2003) can be found in Table 1.

Table 1: Top 20 features (minimal-optimal set).

Feature
RIX
ARI
Kincaid
GunningFogIndex
LIX
SMOGIndex
typ_init_trigr_abs
wiener_sachtextformel
FleschReadingEase
typ_init_trigr_nor
Google00
Coleman-Liau
typ_uniq_orth_strict_pos
Google00pm
DaleChallIndex
typ_syls_cumfreq_type_rank123
Google00cum
typ_uniq_lemma_strict_pos
typ_cts_cumfreq_type_abslog
typ_fam_abs

3.2 Transformer Models

Regression models using neural network architectures based on the Transformer were trained via fine-tuning on the dataset provided by the task organizers. A selection of the estimators that were used in order to generate some of the best scoring submissions is as follows:

- **NN**: BERT (Devlin et al., 2019) fine-tuned for 1 epoch [4].
- **NNr**: BERT fine-tuned for 1 epoch (reverse word order) [5].
- **NN3**: RoBERTa (Liu et al., 2019) fine-tuned for 3 epochs [6].

[4]https://huggingface.co/dbmdz/bert-base-german-cased
[5]https://huggingface.co/dbmdz/bert-base-german-cased
[6]https://huggingface.co/xlm-roberta-base

- **NN5**: BERT fine-tuned for 2 epochs [7].

3.3 Ensemble

Meta-modeling techniques were applied in order to combine base models into single predictors by using stacking generalization. The second level algorithm used for this task was linear regression which used the following weights for Ensemble1 and Ensemble2 respectively:

$$Ensemble1 = 0.18 \times LR + 0.17 \times LGB + 0.21 \times NN + 0.39 \times NNr$$

$$Ensemble2 = 0.1 \times LR + 0.05 \times LGB + 0.02 \times NN + 0.05 \times NNr + 0.25 \times NN3 + 0.478 \times NN5$$

The out-of-fold cross validation scores of the base and meta models can be found in Table 2 .

Table 2: Train set errors calculated with 5-fold cross validation.

Model	RMSE	MAE
LR	0.726	0.585
LGB	0.707	0.561
NN	0.685	0.542
NNr	0.662	0.527
NN3	0.673	0.531
NN5	0.61	0.477
Ensemble1	0.625	0.5
Ensemble2	**0.588**	**0.464**

4 Model Selection and Adversarial Validation

In the final phase of the competition it became clear that validation and test data had relevant dissimilarities. Some potential reasons were identified by the participants such as the application of non-random splits or different pre-processing [8]. While this is not a totally uncommon phenomenon in NLP (Karpov, 2017; Mosquera, 2020) to the best of the authors' knowledge there have not been many efforts to address this problem in comparison with other domains.

The use of adversarial validation as a solution to identify concept drift has been explored recurrently in the literature (Pan et al., 2020). However, due the relatively small data sizes involved, the usual approach of training a binary classifier between train/dev/test sets and selecting the data points with the closest distribution (Qian et al., 2021) was deemed sub-optimal. Therefore, Principal Component Analysis (PCA) was used instead in order to calculate low-dimensional projections of the evaluation datasets and estimate their drift from the training data by analyzing the reconstruction errors. Taking that into account, the author hypothesized that models using features that would remain stable across different data splits based on the criteria define above would exhibit better correlation between the errors estimated during cross-validation and the final scores.

In Table 3, it can be observed that the Ensemble1 meta-model could be affected by the data drift and its estimated performance during the development phase would likely not translate to the final phase evaluation. These insights were particularly relevant since development phase models were also partially tuned using feedback from the public leaderboard and ignoring this valuable information would have resulted in a non-optimal model and feature selection.

Table 3: PCA reconstruction errors against train (3 components, 0.95 variance).

Model	Train	Development	Test
Ensemble1	0.0275	0.0311	0.029
Ensemble2	**0.0329**	**0.0334**	**0.0328**

5 Results

Aiming to compensate for the possible variance between several subjective ratings, the challenge organizers decided to use a custom evaluation metric by applying a 3rd order linear mapping function per each dataset before calculating the error, which meant that the RMSE score would always differ from the mapped RMSE. Considering that the mapping function was unknown to the participants, RMSE was used instead as the main metric for local validation and optimization purposes.

While in practice, the mapped scores seemed to correlate with the local validation, rankings based on RMSE scores differed substantially from rankings derived from the mapped version. This was particularly obvious during the development phase [9] where only 2 out of 14 participants (5 out of 14 during the final phase) had the same ranking when

[7] https://huggingface.co/amine/bert-base-5lang-cased
[8] https://codalab.lisn.upsaclay.fr/forums/4964/741/
[9] https://codalab.lisn.upsaclay.fr/competitions/4964#results

considering both mapped and original RMSE metrics which highlights the extra difficulty added by the chosen evaluation metric for this competition.

In Table 4 the results for the highest ranked submissions generated by the described approach during different phases of the competition are listed. As expected after the adversarial validation step, the meta-model Ensemble1, which produced a high score solution during the evaluation phase (lowest RMSE, second best mapped RMSE overall), underperformed in the final phase and would have ended up in the 9th position in absence of better submissions after being affected by the aforementioned data drift. On the other hand, the meta-model Ensemble2 had similar reconstruction errors in all the datasets and ended up generating the winning submission.

Table 4: Task results (mapped RMSE and RMSE) for selected submissions during different competition phases.

Model	Development	Test
Ensemble1	**0.326 - 0.361**	0.484 - 0.502
Ensemble2	n/a	**0.43 - 0.446**

Since the challenge organizers decided to not release the labels of the evaluation datasets and disabled post-competition submissions, additional ablation analysis can not be performed in this section.

6 Conclusions and Future Work

This paper introduces a meta-model for German text complexity estimation using both manual feature engineering and neural networks. The use of adversarial validation by comparing feature distribution changes between different datasets is proposed as a mechanism to detect data drift via PCA reconstruction errors. The described system has achieved the first ranking in mapped RMSE in the Text Complexity DE Challenge of KONVENS 2022. In a future work, this approach can be extended through AutoNLP techniques in order to build multi-lingual text complexity estimation solutions that could be integrated in other NLP pipelines.

References

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

Marc Brysbaert, Matthias Buchmeier, Markus Conrad, Arthur Jacobs, Jens Bölte, and Andrea Böhl. 2011. The word frequency effect: A review of recent developments and implications for the choice of frequency estimates in german. *Experimental psychology*, 58:412–24.

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.

Max Coltheart. 1977. Access to the internal lexicon.

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.

Chris Ding and Hanchuan Peng. 2003. Minimum redundancy feature selection from microarray gene expression data. volume 3, pages 523– 528.

Pierre Finnimore, Elisabeth Fritzsch, Daniel King, Alison Sneyd, Aneeq Ur Rehman, Fernando Alva-Manchego, and Andreas Vlachos. 2019. Strong baselines for complex word identification across multiple languages. 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 970–977, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Luciana Forti, Alfredo Milani, Luisa Piersanti, Filippo Santarelli, Valentino Santucci, and Stefania Spina. 2019. Measuring text complexity for Italian as a second language learning purposes. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 360–368, Florence, Italy. Association for Computational Linguistics.

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

Julia Hancke, Sowmya V., and Detmar Meurers. 2012. Readability classification for german using lexical, syntactic, and morphological features. pages 1063–1080.

Julian Heister, Kay-Michael Würzner, Johannes Bubenzer, Edmund Pohl, Thomas Hanneforth, Alexander Geyken, and Reinhold Kliegl. 2011. dlexdb—eine lexikalische datenbank für die psychologische und linguistische forschung. *Psychologische Rundschau*, 62:10–20.

Nikolay Karpov. 2017. NRU-HSE at SemEval-2017 task 4: Tweet quantification using deep learning architecture. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 683–688, Vancouver, Canada. Association for Computational Linguistics.

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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3149–3157, Red Hook, NY, USA. Curran Associates Inc.

Alan Kennedy, Joël Pynte, and Stéphanie Ducrot. 2002. Parafoveal-on-foveal interactions in word recognition. *The Quarterly journal of experimental psychology. A, Human experimental psychology*, 55:1307–37.

Susan Lima and Albrecht Inhoff. 1985. Lexical access during eye fixations in reading. effects of word-initial letter sequence. *Journal of experimental psychology. Human perception and performance*, 11:272–85.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

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

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.

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. 2021. Alejandro mosquera at SemEval-2021 task 1: Exploring sentence and word features for lexical complexity prediction. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 554–559, Online. Association for Computational Linguistics.

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

Jing Pan, Vincent Pham, Mohan Dorairaj, Huigang Chen, and Jeong-Yoon Lee. 2020. Adversarial validation approach to concept drift problem in user targeting automation systems at uber.

Steven Piantadosi, Harry Tily, and Edward Gibson. 2011. Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences of the United States of America*, 108:3526–9.

Hongyi Qian, Baohui Wang, Ping Ma, Lei Peng, Songfeng Gao, and You Song. 2021. Managing dataset shift by adversarial validation for credit scoring.

Andre Quispesaravia, Walter Perez, Marco Sobrevilla Cabezudo, and Fernando Alva-Manchego. 2016. Coh-Metrix-Esp: A complexity analysis tool for documents written in Spanish. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4694–4698, Portorož, Slovenia. European Language Resources Association (ELRA).

Renate A. Schulz, Richard Bamberger, and Erich Vanecek. 1985. Lesen-verstehen-lernen-schreiben: Die schwierigkeitsstufen von texten in deutscher sprache. *Die Unterrichtspraxisteaching German*, 18:366.

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

Julia Suter, Sarah Ebling, and Martin Volk. 2016. Rule-based automatic text simplification for german. In *13th Conference on Natural Language Processing (KONVENS 2016)*. s.n.

Zarah Weiss, Anja Riemenschneider, Pauline Schröter, and Detmar Meurers. 2019. Computationally modeling the impact of task-appropriate language complexity and accuracy on human grading of German essays. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 30–45, Florence, Italy. Association for Computational Linguistics.

David Wolpert. 1992. Stacked generalization. *Neural Networks*, 5:241–259.