

Artificial Text Detection with Multiple Training Strategies

Bin Li^{1*}, Yixuan Weng^{2*}, Qiya Song^{1*}, Hanjun Deng³

¹ College of Electrical and Information Engineering, Hunan University, Changsha, China

² National Laboratory of Pattern Recognition Institute of Automation,
Chinese Academy Sciences, Beijing, China

³ Experimental High School Affiliated to Beijing Normal University, Beijing, China
{libincn, sqyunb}@hnu.edu.cn, wengsyx@gmail.com, Hanjun_Deng@gmail.com

Abstract

As the deep learning rapidly promote, the artificial texts created by generative models are commonly used in news and social media. However, such models can be abused to generate product reviews, fake news, and even fake political content. The paper proposes a solution for the Russian Artificial Text Detection in the Dialogue shared task 2022 (RuATD 2022) to distinguish which model within the list is used to generate this text. We introduce the DeBERTa pre-trained language model with multiple training strategies for this shared task. Extensive experiments conducted on the RuATD dataset validate the effectiveness of our proposed method. Moreover, our submission ranked **second place** in the evaluation phase for RuATD 2022 (Multi-Class).

Keywords: Artificial Text Detection, Pre-trained Language Model, Multiple Training Strategies.

DOI: 10.28995/2075-7182-2022-20-375-381

Обнаружение искусственного текста с несколькими стратегиями обучения

Бин Ли

Хунаньский университет
Район Юэлу, город Чанша
libincn@hnu.edu.cn

Йи Суан Вен

Китайская академия наук
Район Хайдянь, Пекин
wengsyx@gmail.com

Кийя Сон

Хунаньский университет
Район Юэлу, город Чанша
sqyunb@hnu.edu.cn

Хан Джун Дэн

Средняя школа при
Пекинском педагогическом университете
Район Сичэн, Пекин
Hanjun_Deng@gmail.com

Аннотация

С быстрым распространением глубокого обучения все чаще тексты, созданные с помощью генеративных моделей, начинают использоваться в новостях и социальных сетях. Однако такими моделями можно злоупотреблять для создания отзывов на продукты, фейковых новостей и даже фальшивого политического контента. В этой работе предлагается решение для задачи классификации модели, использованной для генерации текста на русском языке, в рамках соревнования Dialogue RuATD-2022. Мы применили предобученную языковую модель DeBERTa с несколькими стратегиями обучения. Обширные эксперименты с данными RuATD подтверждают эффективность предложенного нами метода. Кроме того, представленные нами результаты заняли **второе место** на этапе оценки RuATD 2022 (Мультиклассовая классификация).

Ключевые слова: Обнаружение сгенерированного текста, предобученная языковая модель, несколько стратегий обучения.

1 Introduction

With the rapid development of AI technologies, a growing number of methods the ability to generate realistic artifacts. For instance, amount of texts generated by recent text generation methods the transformer

*These authors contribute equally to this work.

encoder-decoder framework are very close to the text written by humans, including lots of security issues (de Rosa and Papa, 2021; Topal et al., 2021). Extensive Transformer-based text generation models, such as Bert-style (Devlin et al., 2018a), GPT-style (Radford et al., 2019), have achieved excellent results on a large number of NLP tasks. (Keskar et al., 2019) proposed the conditional transformer language model (CTRL) with 1.63 billion parameters to control the text generation. The model is trained with different codes that control task-specific behavior, entities, specify style and content. (Zellers et al., 2019) introduced a controllable text generation model named Grover, which can overwrite propaganda papers. For example, given a headline "Discovering the link between vaccines and autism," Grover could generate a description article for this title. Humans rated this generated text as more trustworthy than human-written text. However, the success of natural language generation has drawn dual-use concerns. On the one hand, applications such as summary generation and machine translation are positive. On the other hand, related techniques may also enable adversaries to generate neural fake news, targeted propaganda and even fake political content. Therefore, several researchers have made many attempts to develop artificial text detectors (Jawahar et al., 2020). (Solaiman et al., 2019) used the pre-trained language model RoBERTa for the downstream text detection task and achieved the best performance in recognizing web pages generated by the GPT-2 model. (Kushnareva et al., 2021) proposed a novel method based on Topological Data Analysis (TDA). The interpretable topological features that can be derived from the attention map of any transformer-based language model are introduced for the task of artificial text detection. (Shamardina et al., 2022) originated two tracks on the RuATD 2022 Dialogue Shared task to solve the problem of automatic recognition of generated texts. In this paper, we adopt DeBERTa method with multiple training strategies for the Russian artificial text detection in the Dialogue shared task 2022 (Multi-Class). More details about our system are introduced in the following sections.

2 Main Method

This section will elaborate on the main method for the Russian artificial text detection dialogue shared task, where we adopt the pre-trained model with multiple training strategies, such as adversarial training, child tuning, and intrust loss function.

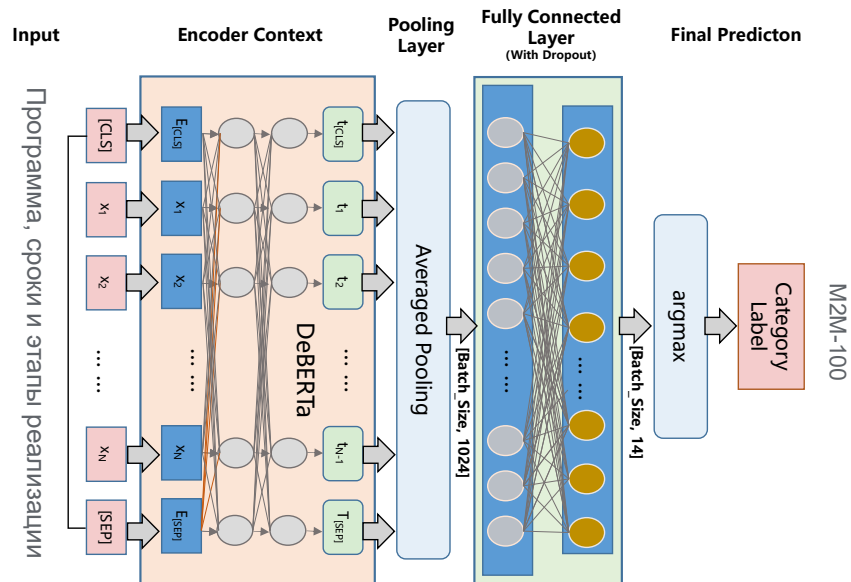


Figure 1: The model architecture of the submitted system.

2.1 Overview of Pre-trained Model

It is noted that the pre-trained model has a solid ability to differentiate the results generated by different models (Qiu et al., 2020; Naseem et al., 2021), we resort to the state-of-the-art (SOTA) pre-trained

language model for better prediction. As shown in Figure 1, we present the main model architecture for the shared task. Specifically, we adopt the DeBERTa family, i.e., mDeBERTa (He et al., 2020) and DeBERTa (He et al., 2021a), for the category classification. The pooling and fully connected layers are at the top of the pre-trained language model for leveraging global semantics. Finally, the argmax is performed after the 14 categories classification to obtain the final results.

2.2 Multiple Training Strategies

2.2.1 Adversarial Training

The common method in adversarial training is the Fast Gradient Method (Nesterov, 2013; Dong et al., 2018). The idea of FGM is straightforward. Increasing the loss is to increase the gradient so that we can take

$$\Delta x = \epsilon \nabla_x L(x, y; \theta) \quad (1)$$

where x represents the input, y represents the label, θ is the model parameter, $L(x, y; \theta)$ is the loss of a single sample, Δx is the anti-disturbance.

Of course, to prevent Δx from being too large, it is usually necessary to standardize $\nabla_x L(x, y; \theta)$. The more common way is

$$\Delta x = \epsilon \frac{\nabla_x L(x, y; \theta)}{\|\nabla_x L(x, y; \theta)\|} \quad (2)$$

2.2.2 Child-tuning Training

The efficient Child-tuning (Xu et al., 2021) method is used to fine-tuning the backbone model in our method, where the parameters of the Child network are updated with the gradients mask. For this shared task, the task-independent algorithm is used for child-tuning. When fine-tuning, the gradient masks are obtained by Bernoulli Distribution (Chen and Liu, 1997) sampling from in each step of iterative update, which is equivalent to randomly dividing a part of the network parameters when updating. The equation of the above steps is shown as follows

$$w_{t+1} = w_t - \eta \frac{\partial \mathcal{L}(w_t)}{\partial w_t} \odot B_t \quad (3)$$

$$B_t \sim \text{Bernoulli}(p_F)$$

where the notation \odot represents the dot production, p_F is the partial network parameter.

2.2.3 Ensemble Method

Once obtaining the pre-trained model, we need to maximize the advantages of each model. So, we use ensemble each model with the Bagging Algorithm (Skurichina and Duin, 2002) via voting on the predicted results of the trained models. The Bagging algorithm is used during the prediction, where this method can effectively reduce the variance of the final prediction by bridging the prediction bias of different models, enhancing the overall generalization ability of the system.

3 Experiment

We will introduce the RuATD dataset, evaluation indicators, implementation details and method description.

3.1 RuATD

However, some people may use these models with malicious intent to generate false news, automatic product reviews, and false political content. RuATD 2022 proposes a new task, which requires judging whether a sentence is generated by the model (binary classification) or even which model it is generated by (multi-class classification). More task details can be found in the website¹.

¹<https://www.kaggle.com/c/ruatd-2022-multi-task>

Model	FGM	ChildTune	In-trust loss	Backbone
DeBERTa-large	61.27	61.23	61.54	61.42
mDeBERTa-base	61.89	61.68	62.21	62.06

Table 1: The experimental results of the text detection.

Methods	Accuracy
Random sample	19.927
Tf-idf	44.280
BERT fine-tuning	59.813
Ours	64.731

Table 2: Comparison with baselines in official test set.

3.2 Evaluation

In the multi-classification task, accuracy is used as the evaluation index. The task requires the model to judge whether a sample is written by humans or generated by other generation models.

$$Acc = \left(\frac{Right}{All} \right)$$

3.3 Baseline Introduction

Tf-idf: With the help of sklearn (Pedregosa et al., 2018), the organizers connect TF-IDF, SVD, standard-scaler and logistic regression in turn for training.

BERT Fine-tuning: BERT (Devlin et al., 2018b) is a model designed for natural language understanding task. It uses MLM pre-training method and has strong semantic feature understanding ability. The organizer added a 14 category linear layer after the output layer of BERT-base, and Cross-Entropy loss is used to fit.

3.4 Implementation Details

We train the model using the Pytorch² (Paszke et al., 2019) on the NVIDIA RTX3090 GPU and use the hugging-face³ (Wolf et al., 2020) framework. For all uninitialized layers, We set the dimension of all the hidden layers in the model as 1024. The AdamW (Loshchilov and Hutter, 2018) optimizer which is a fixed version of Adam (Kingma and Ba, 2014) with weight decay, and set β_1 to 0.9, β_2 to 0.99 for the optimizer. We set the learning rate to $1e - 5$ with the warm-up (He et al., 2016). The batch size is 32. We set the maximum length of 280, and delete the excess. Linear decay of learning rate and gradient clipping is set to $1e - 4$. Dropout (Srivastava et al., 2014) of 0.1 is applied to prevent over-fitting. All experiments select the best parameters in the valid set. Finally, we report the score of the best model (valid set) in the test set.

We use the mDeBERTa-base (He et al., 2021c; He et al., 2021b) as our pretrained model, and fine-tune the model⁴. The mDeBERTa⁵ model comes with 12 layers and a hidden size of 768. And it was trained with the CC100 (Conneau et al., 2020) multilingual data .

4 Case Study

We counted and analyzed the mispredicted samples, and the distribution of error types is shown in Figures 2. We chose the top 100 samples with the most significant difference from the ground truth as the analysis

²<https://pytorch.org>

³<https://github.com/huggingface/transformers>

⁴You can reproduce the baseline code from here <https://github.com/dialogue-evaluation/RuATD/blob/main/Baseline.ipynb>

⁵[microsoft/mdeberta-v3-base](https://github.com/microsoft/mdeberta-v3-base)

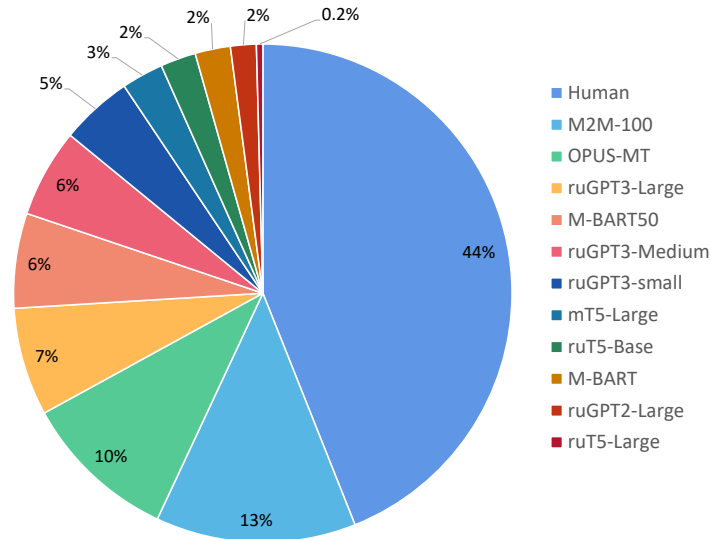


Figure 2: Case study in the shared task.

object. As we can see from Figure 2, the most mispredicted type in the classification task was “Human”, with 44%, followed by “M2M-100” with 13%, etc, the “ruT5-Large” obtains the least error with 0.2%. Further conclusions can be that sorting from high to low actually shows the capability performance of the model. The higher the error rate, the better the performance of the model, and the effects like M2M/GPT3 are better. Then the bigger the model, the harder the target is to distinguish.

5 Result and Discussion

As shown in Table 1, we implement the DeBERTa-large and mDeBERTa-large with multiple training strategies. It can be further concluded that the in trust loss method with the pre-trained model can achieve the best results in artificial text detection. It may be the reason that the model is trained through In-trust training. can be more robust. Moreover, we found that the mDeBERTa outperforms the original version, which indicates that the multi-lingual can provide differentiated knowledge for this text detection. Table 2 also presents the comparison between ours and baselines, where our method outperforms the BERT baseline by 2.397 in accuracy score on the official test set.

6 Conclusion

This paper illustrates our contributions for Russian Artificial Text Detection Dialogue Shared task (Multi Class). We use the DeBERTa pre-trained language model with multiple training strategies to distinguish which model from the list was used to generate this text. In the evaluation phase, our submission achieves **second place**.

References

- Sean X Chen and Jun S Liu. 1997. Statistical applications of the poisson-binomial and conditional bernoulli distributions. *Statistica Sinica*, P 875–892.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. // *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, P 8440–8451, Online, July. Association for Computational Linguistics.
- Gustavo H de Rosa and Joao P Papa. 2021. A survey on text generation using generative adversarial networks. *Pattern Recognition*, 119:108098.

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018a. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018b. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. // *Proceedings of the IEEE conference on computer vision and pattern recognition*, P 9185–9193.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. // *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, P 770–778.
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021b. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021c. Deberta: Decoding-enhanced bert with disentangled attention. // *International Conference on Learning Representations*.
- Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. 2020. Automatic detection of machine generated text: A critical survey. *arXiv preprint arXiv:2011.01314*.
- Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.
- Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv: Learning*.
- Laida Kushnareva, Daniil Cherniavskii, Vladislav Mikhailov, Ekaterina Artemova, Serguei Barannikov, Alexander Bernstein, Irina Piontkovskaya, Dmitri Piontkovski, and Evgeny Burnaev. 2021. Artificial text detection via examining the topology of attention maps. *arXiv preprint arXiv:2109.04825*.
- Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. // *International Conference on Learning Representations*.
- Usman Naseem, Imran Razzak, Shah Khalid Khan, and Mukesh Prasad. 2021. A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. *Transactions on Asian and Low-Resource Language Information Processing*, 20(5):1–35.
- Yu Nesterov. 2013. Gradient methods for minimizing composite functions. *Mathematical programming*, 140(1):125–161.
- 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. // H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Andreas Müller, Joel Nothman, Gilles Louppe, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2018. Scikit-learn: Machine learning in python.
- Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

- Tatiana Shamardina, Vladislav Mikhailov, Daniil Chernianskii, Alena Fenogenova, Marat Saidov, Anastasiya Valeeva, Tatiana Shavrina, Ivan Smurov, Elena Tutubalina, and Ekaterina Artemova. 2022. Findings of the The RuATD Shared Task 2022 on Artificial Text Detection in Russian. // *Computational Linguistics and Intellectual Technologies: Papers from the Annual Conference "Dialogue"*, volume 21.
- Marina Skurichina and Robert PW Duin. 2002. Bagging, boosting and the random subspace method for linear classifiers. *Pattern Analysis & Applications*, 5(2):121–135.
- Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askeff, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.
- Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.
- M Onat Topal, Anil Bas, and Imke van Heerden. 2021. Exploring transformers in natural language generation: Gpt, bert, and xlnet. *arXiv preprint arXiv:2102.08036*.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. // *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, P 38–45, Online, October. Association for Computational Linguistics.
- Runxin Xu, Fuli Luo, Zhiyuan Zhang, Chuanqi Tan, Baobao Chang, Songfang Huang, and Fei Huang. 2021. Raise a child in large language model: Towards effective and generalizable fine-tuning. // *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, P 9514–9528.
- Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *Advances in neural information processing systems*, 32.