Skolkovo Institute of Science and Technology

METHODS FOR FIGHTING HARMFUL MULTILINGUAL TEXTUAL CONTENT

Doctoral Thesis
by

DARYNA DEMENTIEVA

DOCTORAL PROGRAM IN COMPUTATIONAL AND DATA SCIENCE AND ENGINEERING

Supervisor
Assistant Professor, Alexander Panchenko

Moscow, 2022

I hereby declare that the work presented in this thesis was carried out by myself at Skolkovo Institute of Science and Technology, Moscow, except where due acknowledgement is made, and has not been submitted for any other degree.

Candidate (Daryna Dementieva)

Supervisor (Ass. Prof. Alexander Panchenko)

Warning: this work contains texts with rude, obscene words only for example illustrations. We have no intent to offend the reader.

Abstract

Today, a large spectrum of Natural Language Processing (NLP) models has been developed that cover various fields. However, quite a few of the modern NLP technologies were explored in terms of their application for social good. Another important direction covered in this dissertation is multilinguality. Indeed, if an NLP technology for social good is being developed, it's crucial to enable it not only for one language which is usually English but for the wide spectrum of languages used in the world for the maximal positive impact.

In this dissertation, we develop new models that provide applications of how multilingual NLP can be used to tackle the problem of harmful textual content. In Part I, we explore how multilingual NLP technologies can be used to combat fake news. Firstly, we introduce Multiverse – a new feature for fake news detection that is based on cross-lingual evidence extracted from a multilingual search. Then, we explore different approaches to measure the similarity between multilingual and cross-lingual news. In the end, we provide a demonstration of how the proposed fake news detection pipeline with multilingual evidence can be visualized for users and add more explainability to the fake news detection model decision.

Part II is dedicated to the method to fight toxicity with text detoxification. Firstly, we introduce ParaDetox – a new parallel data set of pairs `toxic` $\leftrightarrow$ `nontoxic`. We test the presented approach for such data collection for two languages – Russian and English. After parallel dataset creation, we introduce new methods for detoxification. Firstly, we propose condBERT – a new unsupervised methods for text style transfer. Then, we develop new monolingual supervised text detoxification models EN-Detox and RU-Detox that achieve current state-of-the-art for the text detoxification task. Additionally, we explore the possibility of proposed models to be extended to multilingual and cross-lingual text detoxification setups. We provide several system demonstrations of how proposed models can be already deployed to fight toxic speech. Finally, we provide a discussion of text style transfer task evaluation.

We conclude by discussing the contributions of this dissertation as well as future directions toward the development of NLP systems for social good.

Publications

Main author

1. **Daryna Dementieva**, Igor Markov, and Alexander Panchenko. SkoltechNLP at SemEval-2020 task 11: Exploring unsupervised text augmentation for propaganda detection. In Aurélie Herbelot, Xiaodan Zhu, Alexis Palmer, Nathan Schneider, Jonathan May, and Ekaterina Shutova, editors, *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 1786–1792. International Committee for Computational Linguistics, 2020. URL https://doi.org/10.18653/v1/2020.semeval-1.234 [SCOPUS]

2. **Daryna Dementieva** and Alexander Panchenko. Cross-lingual evidence improves monolingual fake news detection. In Jad Kabbara, Haitao Lin, Amandalynne Paullada, and Jannis Vamvas, editors, *Proceedings of the ACL-IJCNLP 2021 Student Research Workshop, ACL 2021, Online, JUli 5-10, 2021*, pages 310–320. Association for Computational Linguistics, 2021. doi:10.18653/v1/2021.acl-srw.32. URL https://doi.org/10.18653/v1/2021.acl-srw.32 [SCOPUS]

3. **Daryna Dementieva** and Alexander Panchenko. Fake news detection using multilingual evidence. In Geoffrey I. Webb, Zhongfei Zhang, Vincent S. Tseng, Graham Williams, Michalis Vlachos, and Longbing Cao, editors, *7th IEEE International Conference on Data Science and Advanced Analytics, DSAA 2020, Sydney, Australia, October 6-9, 2020*, pages 775–776. IEEE, 2020. URL https://doi.org/10.1109/DSAA49011.2020.00111 [CORE A]

4. **Daryna Dementieva**, Daniil Moskovskiy, Varvara Logacheva, David Dale, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. Methods for detoxification of texts for the russian language. *Multimodal Technol. Interact.*, 5(9): 54, 2021. URL https://doi.org/10.3390/mti5090054 [Q2]

5. **Daryna Dementieva**, Daniil Moskovskiy, Varvara Logacheva, David Dale, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. Methods for detox-

ification of texts for the russian language. In *Computational Linguistics and Intellectual Technologies*, 2021. URL https://www.dialog-21.ru/media/5503/dementievadplusetal046.pdf [SCOPUS]

6. Varvara Logacheva*, **Daryna Dementieva***, Sergey Ustyantsev, Daniil Moskovskiy, David Dale, Irina Krotova, Nikita Semenov, and Alexander Panchenko. ParaDetox: Detoxification with parallel data. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6804–6818. Association for Computational Linguistics, 2022a. URL https://aclanthology.org/2022.acl-long.469 [CORE A*]

7. Varvara Logacheva*, **Daryna Dementieva***, Irina Krotova, Alena Fenogenova, Irina Nikishina, Tatiana Shavrina, and Alexander Panchenko. A study on manual and automatic evaluation for text style transfer: The case of detoxification. In *Proceedings of the 2nd Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 90–101, Dublin, Ireland, May 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.humeval-1.8 [SCOPUS]

8. **Daryna Dementieva**, Varvara Logacheva, Irina Nikishina, Alena Fenogenova, David Dale, Irina Krotova, Nikita Semenov, Tatiana Shavrina, and Alexander Panchenko. RUSSE-2022: Findings of the first Russian detoxification task based on parallel corpora. In *Computational Linguistics and Intellectual Technologies*, 2022 [SCOPUS]

Co-author

9. David Dale, Anton Voronov, **Daryna Dementieva**, Varvara Logacheva, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. Text detoxification using large pre-trained neural models. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021,*

Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021, pages 7979–7996. Association for Computational Linguistics, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.629 [CORE A]

10. Daniil Moskovskiy, **Daryna Dementieva**, and Alexander Panchenko. Exploring cross-lingual text detoxification with large multilingual language models. In Samuel Louvan, Andrea Madotto, and Brielen Madureira, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 346–354. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.acl-srw.26 [SCOPUS]

11. Mikhail Kuimov, **Daryna Dementieva**, and Alexander Panchenko. SkoltechNLP at semeval-2022 task 8: Multilingual news article similarity via exploration of news texts to vector representations. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1136–1144, 2022 [SCOPUS]

Acknowledgments

The path to my Ph.D. was full of adventures and incredible life turns (pleasant and not very, even life-threatening in places), but fate gave me the people who made this dissertation possible. Because anyway, in the end, the most important thing that remains is human relationships.

Foremost, I would like to salute gratitude to my mother Olena Dementieva and grandmother Yefroyniia Mochalova. There are no words to express the full magnitude of their work and my corresponding feeling of infinite gratitude. Both of them were my inspiration for women's path in the academic field. Specifically, my grandmother insisted that I go to graduate school, which turned out to be the best decision for my age. They remain my family and the main support group for my whole life.

Secondly, words of gratitude are expressed to my "academic father", to my supervisor Alexander Panchenko. It was an honor to be one of his first Ph.D. students of him and to work with him for these three years. This dissertation is the result of our connection, his constant belief in me, and his guidance in the right direction. I am grateful to him for the fact that it was he who taught me how to be a researcher and shared his experience not only academically but also in life. At the most dangerous moment in my life, Alexander provided the help that my family needed so much.

Along with that, I want to thank all members of the SkoltechNLP group with whom I was lucky to meet and work – Irina Nikishina, Nikolay Babakov, David Dale, and Sergey Ustyantsev. Especially, I would like to stress Varvara Logacheva – we spent amazing two years working together on the project about detoxification. I will always remember the lightness and cool atmosphere in which we worked, discussed ideas, and wrote texts. When I brought parts of the text written by me, Varvara patiently put them in order and guided them with advice. We did an incredible amount of work together and without her, some part of this dissertation would not be possible. I would like to additionally thank Artem Shelmanov. Although we did not cross paths much during his time in the group, it was he who provided me with support and mentorship during my first offline ACL conference. Additionally,

I incredibly appreciate his effort to proof-read this text. Again, to all SkoltechNLP group – guys, you were my family at Skoltech and there will always be memories in my mind of how we always went to dinner together, drank coffee, and celebrated holidays.

In concluding my gratitude to professional colleagues, I would also like to thank the joint SkoltechNLP-MTS AI group and representatives of MTS AI – Olga Kozlova and Nikita Semenov who made this joint research possible. In addition, I would like to thank Irina Krotova. Irina took a huge part in our weekly discussions and brought a lot of nice ideas. She supported the deployment of detoxification models into MTS products proving that our research was worth it. I will be always amazed by the patience with which she solved all coming issues.

During my Ph.D., I was honored to have talented Master's students – Mikhail Kuimov and Daniil Moskovskiy. I want to thank them for their regular work and conscientiousness. Our collaboration brought many results that were issued as publications in respectful venues. It was only a pleasure to supervise both of them.

My Ph.D. life would be boring if there were no friends around. But I was rewarded to have wonderful friends. Yulia Sherbakova, Denis Zaplatnikov, Konstantin Lopatin, and Dmitry Kivattsev are the members of my Moscow "family". Of course, they can not take away the humor about life in an academic environment, but who does not benefit from a "critical" look from the outside on their work. These guys have been the filling of my life. Some friends were not always physically there, but this did not prevent our connection from always being strong – all this about my dearest friend Karyna Volenko. During the first year, she helped me a lot with proofreading my first papers. The value of our connection is just priceless.

Unfortunately, my life was overshadowed by a completely unexpected and out-of-the-ordinary event. In a time of uncertainty about what to do next with life of some incredible whims of fate, people from the German city of Mannheim helped me. I would like to express my gratitude to the Mannheim NLP group led by professor Simone Paolo Ponzetto. Prof. Ponzetto invited me as a guest to the city and his group to survive the rushing events. I will refer to him in my head as "academic uncle" and to the members of his group as "academic cousins" who they really are

behind the scenes. Again fate brought me to another family. Forever, our WG from Turly-Platz 9 took a special place in my heart. The names of the members of this family are Dani Siebert, Birgit Thomas, Julia Strauch, Nina Walker-Emig, and Torsten Schlusche. Exceptional love and gratitude I express to Britta Schlichting for inspiring me to return back to life, feeling loved, and being myself. Our WG made me feel uncomfortably comfortable even far away from my motherland. I only hope that our connection will be lifelong.

From my Munich adventures, I would like to thank my friends Maryna Nemirovskaya and Cristian Soare who also helped me to make the continuation of my life possible. Finally, I would like to thank Edoardo Mosca, Tobias Eder, Prof. Georg Groh, and his wife Hannah Danner for their effort to make my life stabilize.

As you can see, it indeed requires the number of people of a small village to raise a Ph.D. candidate. But I believe, specific people and their love can inspire a person to do science for humanity.

Contents

Glossary

AI Artificial Intelligence. 23, 24, 29

AI4SG Artificial Intelligence for Social Good. 30

GLUE General Language Understanding Evaluation. 40

LLM Large Language Model. 45, 116, 117, 165, 181, 184

ML Machine Learning. 19, 20, 35, 87, 92, 94

MLM Masked Language Modeling. 40, 41

NE Named Entity. 72, 81

NER Named Entity Recognition. 16, 20, 88, 89, 93, 94, 100

NLG Natural Language Generation. 40, 116

NLI Natural Language Inference. 16, 19, 69–71, 79, 81, 87, 88, 92, 93, 99

NLP Natural Language Processing. 3, 24, 25, 29, 40, 106, 181

NLP4SG Natural Language Processing for Social Good. 32, 181

NMT Neural Machine Translation. 36, 116

seq2seq Sequence-to-sequence. 116, 118, 119, 146, 152

SOTA State-of-the-Art. 29, 40, 44, 49, 60, 63, 102

SQuAD The Stanford Question Answering Dataset. 40

TST Text Style Transfer. 103, 116, 118, 119, 144, 165

List of Figures

List of Tables

1

Introduction

"Technology is one of the factors in the life of mankind and is as old as the latter. However, the question of the significance of this factor is new and has only been clarified very little."

– Petr Engelmeyer, *Technology as art* (1900)

"Technology is a useful servant but a dangerous master."

– Christian Lange, *The Nobel Peace Prize* (1921)

"Боритеся – поборете. (Fight – you will win.)"

– Taras Shevchenko, *poem Caucasus* (1845)

1.1 Overview

The impact of technology on society was a question for philosophers and histories already in previous centuries. Today since the 2010's we can observe a huge rise of Artificial Intelligence (AI) based technologies. However, there are still questions about the application possibilities of such technologies, their trustworthiness, and their general impact on humanity.

One of the places where AI-based technologies are intensively used is the Internet. It has already played a significant role in the 4^{th} Industrial Revolution [Groumpos, 2021], changing the way we consume information. In addition, with tremendous

improvements in AI and, especially, Natural Language Processing (NLP) technologies, the information handling process has opened a lot of new possibilities. The tremendous improvement in machine translation [Costa-jussà et al., 2022] makes communication around the world much easier. Even during email writing, NLP models can help you to find errors easily and speed up the writing process with autocomplete [Chen et al., 2019].

On a darker note, with the rise of technology usage, the risk of negative impact or harmful consequences is also increasing. Such consequences can be quite unexpected and impossible to predict immediately when technology is developed. The Internet contains a big amount of textual content. Thus, the majority of information on the Web is being transferred through text – both positive and malicious. Therefore, the development of technologies for fighting harmful textual information is a task of high importance.

For instance, at the same time as Twitter can be used for a notification of important personal or public urgent events, it has become a popular platform for bots to propagate fake news [Singh et al., 2020]. NLP models can be used for the generation of not only summaries or poems, but also for the generation of fake stories written so that they are indistinguishable from human-produced. There has already been a case when a student was publishing stories[1] for two weeks, passing them off as his own while all texts were generated by the newly released GPT-3.[2]

A widespread of toxic and hate speech has become another unexpected problem on the Internet. Social networks were created to share "positive" information – make educational information more accessible, share personal news and photos with important people if they are distant, and provide a platform for community discussions to find a solution or a compromise for bothering issues. However, online social networks have become platforms also for the spread of hate speech full of various toxic comments and statements in discussions [Stroińska, 2020]. Industry resorts to the use of NLP technologies to fight harmful information carefully. For instance, toxicity classification models due today struggle from little interpretability [Carton

[1] https://adolos.substack.com/p/feeling-unproductive-maybe-you-should
[2] https://beta.openai.com/docs/models/gpt-3

et al., 2020] or can be biased [Garg et al., 2022].

Finally, it is worth mentioning that all cited risk of the spread of harmful textual information concerns not only one language community but the variety of languages. For this reason, we find it important to develop multilingual solutions to tackle the spread of harmful information. With the common efforts of the international NLP community, we can develop such methods that will help us win the fight against harmful information spread.

In this dissertation, we focus on two types of harmful information – fake news and toxic speech. It was discovered that for both types there is a lack of work aimed at multilinguality and developing human-oriented technology. The majority of works dedicated to fake news cover only one language. For toxic speech, a lot of work is done to create toxic speech classifiers, but only a few to detoxify texts and none for multilingual cases. As a result, this dissertation addresses the following research questions:

Q1: *How can fake news detection benefit from multilingual evidence?*

Q2: *What NLP technologies (both monolingual and multilingual) can be used to detoxify texts?*

1.2 Contributions and Outline

This work has the following structure.

In **Chapter 2**, we introduce the topic of Artificial Intelligence and Natural Language Processing for Social Good. Then, we provide the theoretical background about Transformer-based models. The Chapter ends with an overview of models for multilingual Natural Language Processing.

Part I is dedicated to answering research question **Q1**. While the majority of previous work covers only one language to build fake news classification systems, we want to address this gap and explore if external multilingual information from the Web search can help to improve fake news detection.

We start from task introduction in **Chapter 3**. We provide an overview of existing fake news detection datasets, systems, and an analysis of how fake news

detection can be motivated by user behavior on the Internet. Here, we provide a formal task definition that we want to address in our work.

In **Chapter 4**, we introduce **Multiverse** – Multilingual Evidence for Fake News Detection. While substantial work has been done in the direction of developing fake news detection models, one of the limitations of the current approaches is that these models focus only on one language and do not use multilingual information. In our work, we propose a new technique based on cross-lingual evidence (CE) that can be used for the detection of fake news and improve existing approaches. The approach is based on the main *hypothesis*: If the news is *true*, then it will be widespread in different languages and also across media with different biases, and the facts mentioned should coincide; on the contrary, if the news is *fake*, it will receive a lesser response in the foreign press than true news or the facts mentioned contradict. First, we confirmed the proposed hypothesis by a manual experiment based on a set of known true and fake news. Then, we compared our fake news classification system based on the proposed feature with several strong baselines on two multi-domain datasets of general-topic news and one new fake COVID-19 news dataset showing that combining cross-lingual evidence with strong baselines yields significant improvements in fake news detection. The content of this chapter is based on the idea presented in [**Daryna Dementieva** and Panchenko, 2021] extended with a deeper analysis of the results and research on explainability.

Continuing the work with multilingual news texts in **Chapter 5**, we explore new metrics for the measurement of multilingual and cross-lingual news similarity based on dataset presented in "Multilingual News Article Similarity" competition at SemEval-2022 ([Chen et al., 2022]). We experiment with a diverse amount of approaches: different text embeddings, addressing the task as a Natural Language Inference task, and extracting additional signals as Named Entities. After the new multilingual news similarity system selection, we integrate it into the fake news detection system. In the end, we provide a demonstration of the proposed fake news classification approach based on multilingual evidence as a web service and how it can be useful to the final users. The result presented in this chapter are based on work [Kuimov et al., 2022] extended with more diverse models used for analysis and

implementation of the proposed approach into a system demonstration.

In **Part II**, we address the problem of toxicity in social texts answering the research question **Q2**. One of the ways to fight toxicity online is to provide a non-toxic variant of the user's message – the user can rethink what he or she wants indeed to express, downgrade the discussion and choose a less emotional variant of the text. This problem can be named *detoxification*.

In **Chapter 6** we provide the formal formulation of the task along with the industrial motivation of the task. Moreover, we provide a very exact formulation with which types of toxic speech we deal in this work. We provide an overview of the existing text style transfer and detoxification approaches. While all previous detoxification-related works focused on the development of unsupervised approaches (the models trained on classical datasets with non-parallel parts of toxic and non-toxic texts), we dedicate the next chapters to the confirmation of the following *hypothesis*: the development of detoxification methods trained on a *parallel* dataset significantly improves the task performance.

Therefore, in **Chapter 7** we introduce **ParaDetox** – a new Parallel Dataset for Detoxification. We describe a new pipeline for the dataset collection, which theoretically can be reused for any other text style transfer task. We provide the details of the collection process and analysis of the most popular edits of toxic texts' parts. In the end, we introduce two versions of the dataset dedicated to two languages – English and Russian. The content of this chapter is based on [Logacheva* et al., 2022a] and [**Daryna Dementieva** et al., 2022] where Dementieva and Logacheva are equal first co-authors extended with a more detailed description of the dataset collection pipeline and comprehensive examples.

After datasets collection, in **Chapter 8**, we investigate new models for detoxification for English and Russian languages separately. Firstly, we introduce **cond-BERT** – a new unsupervised method for text style transfer. This method addresses the problem of detoxification as a replacement of the exact toxic part of the input text with a non-toxic substitution. Then, we provide a detailed description of the evaluation setup: metrics and description of baseline models. Finally, we introduce **EN-Detox** and **RU-Detox** – new state-of-the-art models for English and

Russian detoxification respectively. The detoxification research for the Russian language introduced in our work is the first of its kind exploration of text style transfer methods for Russian texts. In the end, we explore whether multilingual and cross-lingual detoxification based on the proposed datasets and approaches is possible. The results of this chapter is based on [Logacheva* et al., 2022a] where Dementieva and Logacheva are equal first co-authors, [**Daryna Dementieva** et al., 2021], and [Moskovskiy et al., 2022].

As in the previous chapter, the difference between automatic and manual evaluation is confirmed, in **Chapter 9**, we provide a deep exploration of the correlation between modern techniques for automatic and manual evaluations for the detoxification task. We introduce a new pipeline for automated manual evaluation that allows to get manual assessments quickly. In the result, we provide the correlation analysis of automatic and manual evaluation for 15 detoxification systems. Unfortunately, the correlation is still low, showing that there is a future path for the development of more stable systems for automatic text style transfer evaluation. The content of this chapter is based on [Logacheva* et al., 2022b] and [**Daryna Dementieva** et al., 2022] where Dementieva and Logacheva are equal first co-authors.

Finally, in **Chapter 10**, we provide a general discussion of the results and identify the remaining challenges and future directions.

2

Background

This chapter first provides an overview of how Natural Language Processing (NLP) techniques can be used for social good and which challenges can occur (Section 2.1). Secondly, the theoretical background of the Transformer architecture is presented with a description of the known up-to-date model that achieves SOTA results of various NLP tasks (Section 2.2). Finally, we provide an overview of the state of multilinguality in NLP (Section 2.3).

2.1 Natural Language Processing for Social Good

This dissertation addresses some of the aspects of NLP for Social Good. Here, we start with an overview of the broader topic such as Artificial Intelligence for Social Good, then describe what challenges occur in terms of NLP, and specify which exactly problems are covered in this work.

2.1.1 Artificial Intelligence for Social Good

Artificial Intelligence (AI) technologies and, specifically, NLP technologies are integrated into our daily activities. We use search engines that work with quite precise auto-correct and recommendation systems, machine translation has become incredibly precise over the recent years, and NLP-based agents become usual to reduce the load of call centers.

While AI technologies are becoming more and more sophisticated every year, sometimes after scientific breakdowns there raises a question of how newly developed models can be applied to real-world problems. Moreover, the so-called Artificial Intelligence for Social Good (AI4SG) is becoming a more emerging theme. The aim of research in the field of AI4SG is to develop AI methods and tools to address not only industry needs, but also social problems and improve the well-being of society [Shi et al., 2020]. We can use such a definition of AI4SG [Floridi et al., 2020]:

Figure 2-1: Popular domains for AI4SG applications.

Definition 1 ***AI4SG*** *is the design, development, and deployment of AI systems in ways that:*

- *prevent, mitigate, or resolve problems adversely affecting human life and/or the wellbeing of the natural world;*
- *enable socially preferable and/or environmentally sustainable developments;*
- *not introduce new forms of harm and/or amplify existing disparities and inequities.*

There are already several works dedicated to AI4SG technology. The most popular domains in which AI4SG has found its application are education, healthcare, environmental sustainability, agriculture, combating information manipulation, reduced inequalities, transportation, and several more (see Figure 2-1) [Cowls et al., 2021]. For instance, for agriculture, AI technologies can be used to predict crop disease [Quinn et al., 2011]. The in-time prediction of crop diseases can be quite important for developing countries to prevent a deficit. For the environment and climate monitoring, there can be developed systems modeling energy usage [Li and Zha, 2015] or complex ecosystem [Martinez et al., 2012]. However, all these works are only scientific discoveries and there are only a few companies that are working on real-life implementation of the proposed technologies. Most of the AI4SG research has not (yet) achieved observable social impact.

Summarizing the ideas of previous work [Shi et al., 2020, Tomašev et al., 2020, Floridi et al., 2020], we can formulate the main principles of qualitative AI4SG technology:

P1 Humanity: the goal to increase human well-being should be satisfied.

P2 Fairness: the technology should be equal in terms of development and working processes and results for all represented groups.

P3 Transparency: the goals and development steps of the technology should be clear and the level of abstraction should be appropriate for the system and the receivers.

P4 Explainability: the results should be understandable and free to semanticize for the receivers.

P5 Sustainability: the possibility to manipulate the data, technology development, and the analysis of the results should be eliminated.

P6 Dialogue: the conversation between AI developers, social representatives, and receivers should exist.

P7 Security: personal data protection requirements should be satisfied.

2.1.2 Natural Language Processing for Social Good

Natural Language Processing field has as well its Natural Language Processing for Social Good (NLP4SG) initiative [Jin et al., 2021]. The development of NLP technologies already has shown their usefulness in several socially important applications. Thus, social networks, for example, Facebook, are using the NLP model to detect fake information and prevent its widespread spread [Meta, 2018]. During COVID-19 pandemic, the NLP community was united to analyze medical texts to find useful information for treatment development [Bhatia et al., 2020]. The AlphaFold model [Jumper et al., 2021] brought humanity closer to the search for the structure of new proteins that can help discover new medicines.

Figure 2-2: The use case demonstration how of NLP-based chat-bots can be used for mental health treatment help.

Health Care Use-case As AI technology, NLP models also have found applications in different social fields – healthcare, education, equality, agriculture, energy, etc. For instance, NLP-based chatbots can help to treat patients with anxiety or depression [Pham et al., 2022]. Unfortunately, there are cases where it takes quite a long to find a place for therapy – several months or even a year. During this time,

the disease can only progress. Moreover, patients can be ashamed to discuss their problems with friends or family. However, if someone can talk impersonally with someone, that can be helpful. This 'someone', for example, can be a chatbot that shows empathy and the ability to listen to any problem that causes a person to be upset (Figure 2-2).

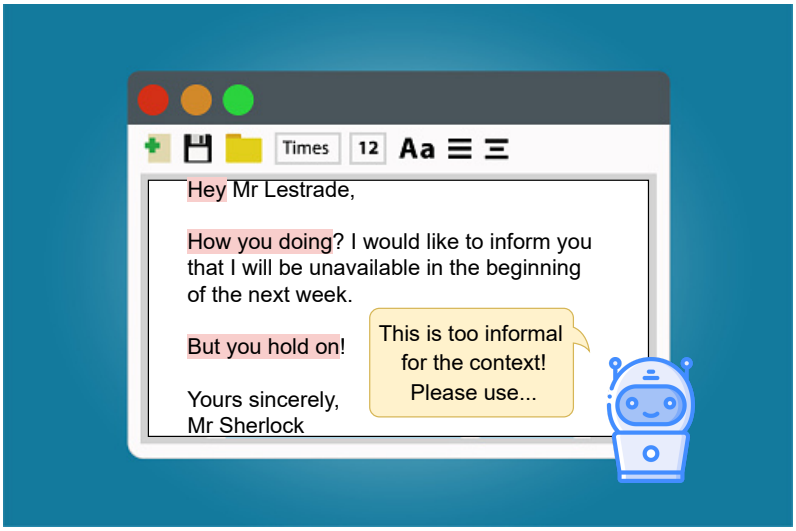


Figure 2-3: The use case demonstration of how NLP-based helpers can be useful to maintain suitable style in document according to situation.

Education Use-Case For educational purposes, one of the applications of NLP can be different helpers for studying foreign languages. During writing, the system can detect if a student makes any mistakes and can also generate explanations for why a mistake should be fixed in the proposed way. There are already works dedicated to the development of such systems and scientific research in this field [Nagata et al., 2022]. Another example can be the checker of appropriate text style usage. If a person uses an inappropriate context style (for instance, mixing informal writing into a formal letter), then an NLP-based helper can detect this misuse and help a user prevent misunderstanding in communication (Figure 2-3). Moreover, it can be an important case for intercultural communications (for example, German academics can be offended if in the official letter you refer to them with just "Mr/Ms" title but not with their obtained academic title as "Dr").

Equality Use-Case Another use-case example is that NLP technologies can be used to detect biases in languages and help respect equity. Humanity has still to make a long way to full equality and diversity. Language is one of the ways to realize ingrained patterns of injustice. In our speech, we can be unintendedly biased using the old patterns of descriptions of gender roles and prejudices about races, professions, and sexuality. Thus, talking about education, there can still be

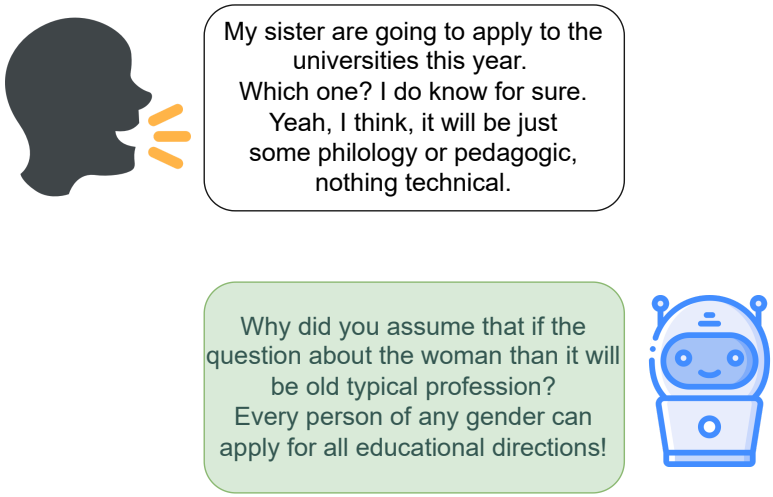


Figure 2-4: The use case demonstration of how NLP-based prompter can help to detect bias in language.

prejudices about the “typical" female or male field of getting an education. However, in modern society, the aim is to provide equal rights for any field of education and then for any profession of any gender. An NLP-based moderator, for instance, in comment sessions or group discussions on social networks, can help chat participants recognize that in their way of thinking, they refer to a biased way of understanding the world (Figure 2-4).

Fight with Harmful Information Use-Cases The development of the Internet and the growing popularity of social networks has led to the dissemination of not only useful information, but also to the dissemination of a large amount of harmful textual information. Unfortunately, it is a common case when the users of a platform can start to insult each other in comments or discussion section or propagate lies or rumors. The types of harmful information on the Internet is quite diverse. The example of several types are presented in Figure 2-5.

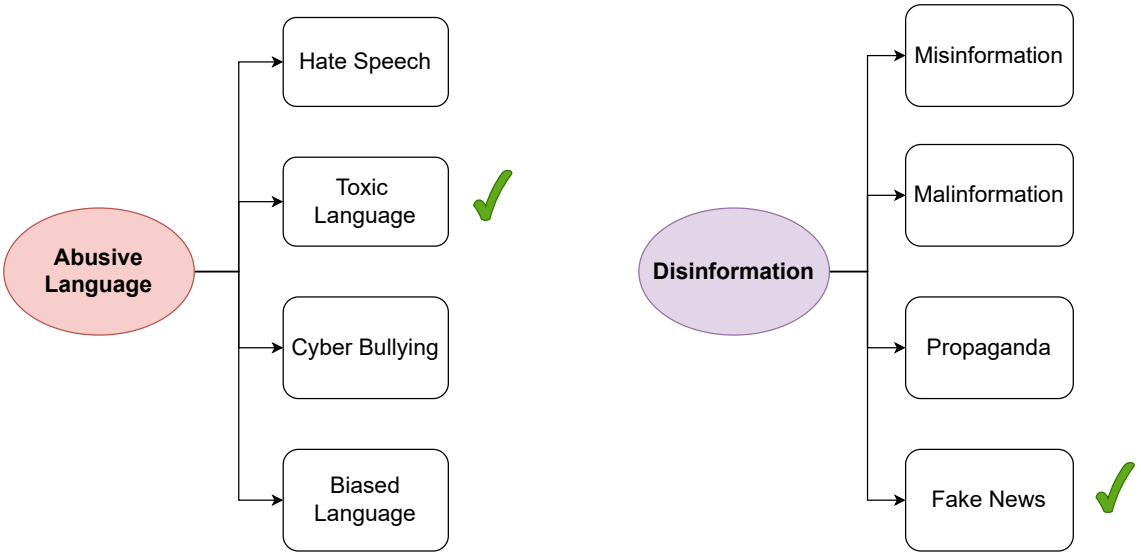


Figure 2-5: The examples of different types of harmful textual information with emphasize of types that are covered in this work.

For sure, there are a lot of NLP research works dedicated to fight these types of harmful information. Thus, there are works dedicated to detect hate speech Mathew et al. [2021] and generate counter speech [Tekiroglu et al., 2020]. Also, there are dataset and work created to detect propaganda [Da San Martino et al., 2020, **Daryna Dementieva** et al., 2020].

Specifically, in this work we propose approaches to fight one of the types of abusive language – toxic speech, and one of the types of misinformation – fake news. Despite all previously developed approaches and datasets, we develop approaches covering not only one language (that is in majority of cases only English), but for multiple languages exploring possibility of multilingual NLP approaches.

2.2 Transformer-based Models

Machine Learning (ML) model is basically a function that takes as input some numerical value and makes a prediction. The text value cannot be taken as an input to such a model as it is. For this reason, one of the main tasks in a text processing pipeline is text vectorization, i.e. projection of a text string into a set of numerical values. Some of the baseline approaches broadly used before are Bag-of-Words (BoW) and TF-IDF, which mostly took into account statistics of occurrences

of words in a document collection. However, these approaches do not take into account word semantics.

To overcome this issue, a more advanced method for word embedding was presented – Word2Vec [Mikolov et al., 2013]. The model is based on a distributional semantics hypothesis, i.e. we can learn the meaning of the word by its "surroundings". Unfortunately, the usage of this model also has limitations. If we feed word vectors into the text sequence processing model one by one, the model can "forget" the information at the beginning. As a remedy to this problem, the attention mechanism was devised.

2.2.1 Attention Mechanism

The Attention mechanism [Graves et al., 2014] allows a model to highlight or "focus attention" relevant sections of the input data, which can be either used with a raw text or any other high-level representation. The core idea is to calculate the weight distribution based on the input sequence and give higher weights to more important parts of the text, while leaving smaller weights for less important parts.

Attention was introduced as a solution to the problem of long sequences of text in Neural Machine Translation (NMT) models. Consider an input sequence $\mathbf{x}$ and an output sequence $\mathbf{y}$:

$$\mathbf{x} = [x_1, x_2, \ldots, x_n] \tag{2.1}$$

$$\mathbf{y} = [y_1, y_2, \ldots, y_m] \tag{2.2}$$

A bidirectional encoder transforms an input sequence $\mathbf{x}$ into a concatenation of hidden forward and backward representations: $\mathbf{h} = [\overrightarrow{h}, \overleftarrow{h}]$. The decoder generates its own hidden state $s_j = \text{decoder}\,(s_{j-1}, y_{j-1}, c_j)$, where s_{j-1} is the previous hidden state of the decoder, y_{j-1} is the last generated element of the output sequence and c_j is the context vector: $c_j = \sum_{i=1}^{n} \alpha_{j,i}\mathbf{h}_i$ is the sum of i^{th} of hidden encoder states $\mathbf{h}$

multiplied by alignment coefficient:

$$\alpha_{j,i} = \text{alignment}(y_j, x_i) = \frac{\exp\left(\text{score}\left(\boldsymbol{s}_{j-1}, \mathbf{h}_i\right)\right)}{\sum_{k=1}^{n} \exp\left(\text{score}\left(\boldsymbol{s}_{j-1}, \boldsymbol{h}_k\right)\right)} \tag{2.3}$$

Alignment is calculated for a i^{th} element of input sequence and j^{th} element of an output sequence. The whole set $\boldsymbol{\alpha}_i = \{\alpha_{j,i}\}_{j=1}^{n}$ is a set of weights that indicates how each element of an input sequence $\boldsymbol{x}$ affects the j^{th} element of an output sequence y_j. The score function can be chosen.

Surely, the method described above is not the only version of the attention mechanism. Currently, there are different variations in the attention mechanism [Graves et al., 2014, Cheng et al., 2016]. We cover the scaled dot product attention introduced by Vaswani et al. [2017] – the most well-known and effective one:

$$\text{score}(\boldsymbol{s}_j, \boldsymbol{h}_i) = \frac{\boldsymbol{s}_j^T \boldsymbol{h_i}}{\sqrt{n}} \tag{2.4}$$

In this case, alignment scores are calculated as a dot product of the vectors $\boldsymbol{s}_j$ and $\boldsymbol{h}_i$. A division by $\sqrt{n}$ (where n is the dimension of the encoder) was added to ensure the stability of the training.

Another variation of attention mechanism is *self-attention* [Cheng et al., 2016]. Self-attention calculates attention weights for an element of the sequence with respect to other elements of this sequence, thus, the impact of the sequence elements on each other is calculated. This approach was proven to be useful in many NLP tasks.

2.2.2 Transformer Architecture

Transformer architecture was introduced in [Vaswani et al., 2017]. Once it appeared, this model proved to be quite useful to solve many NLP tasks such as neural machine translation, sequence-to-sequence modeling, and many other tasks. Having scaled dot-product self-attention mechanism inside, Transformer block and its variations are now a core part of any modern language model [Devlin et al., 2019, Lewis et al., 2020, Raffel et al., 2020, Radford et al., 2019].

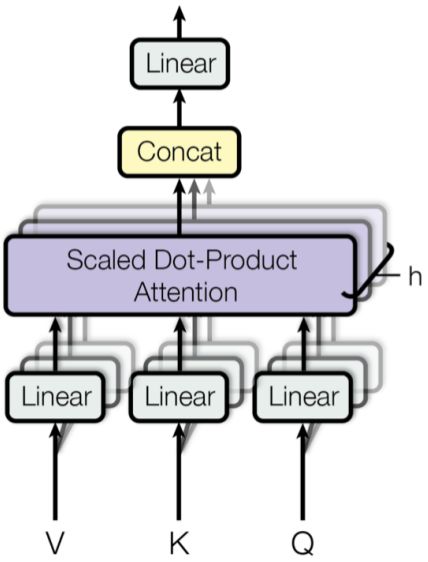


Figure 2-6: Multi-head self-attention - a core part of a Transformer model [Vaswani et al., 2017].

Scaled dot-product self-attention: Query, Key, Value Multihead self-attention lies within the Transformer model. In this case, attention is viewed as a mapping of the output of $\boldsymbol{Q}$ and $\boldsymbol{K}, \boldsymbol{V}$, where $\boldsymbol{Q}$ stands for *query*, $\boldsymbol{K}$ stands for *key* and $\boldsymbol{V}$ represent the value accordingly. Here $\boldsymbol{Q}$ is the previous output of the decoder. Both $\boldsymbol{K}, \boldsymbol{V}$ are encoded representations of the input sequence.

The Transformer employs a scaled dot-product attention version, with the output being a weighted sum of the values. Each weight corresponds to a value given by the scalar product of the query with all the keys:

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{n}}\right)\boldsymbol{V} \tag{2.5}$$

Multi-head self-attention Instead of computing attention weights once or subsequently, multi-head attention is introduced (Figure 2-6): attentions are calculated several times in parallel, and the results are simply concatenated into a large vector. This approach was proven to be beneficial allowing model to obtain knowledge from various representation subspaces while ordinary attention was not able to perform this. During the training process, several weight matrices $\boldsymbol{W}_i^{\boldsymbol{Q}}, \boldsymbol{W}_i^{\boldsymbol{K}}, \boldsymbol{W}_i^{\boldsymbol{V}}$ are learned:

$$\begin{cases} \text{MultiHead}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = [\text{head}_1, \ldots, \text{head}_n]\boldsymbol{W^O} \\ \text{head}_i = \text{Attention}(\boldsymbol{QW_i^Q}, \boldsymbol{KW_i^K}, \boldsymbol{VW_i^V}) \end{cases} \tag{2.6}$$

Position-wise Feed-Forward Network This network is applied to each position of an input vector. The layer consists of two linear layers and ReLU activation between them:

$$f(x) = (\max 0, xW_1 + b)\, W_2 + b_2 \tag{2.7}$$

where W_1 and W_2 are the weight matrices of linear layers, b_1 and b_2 are the corresponding bias vectors.

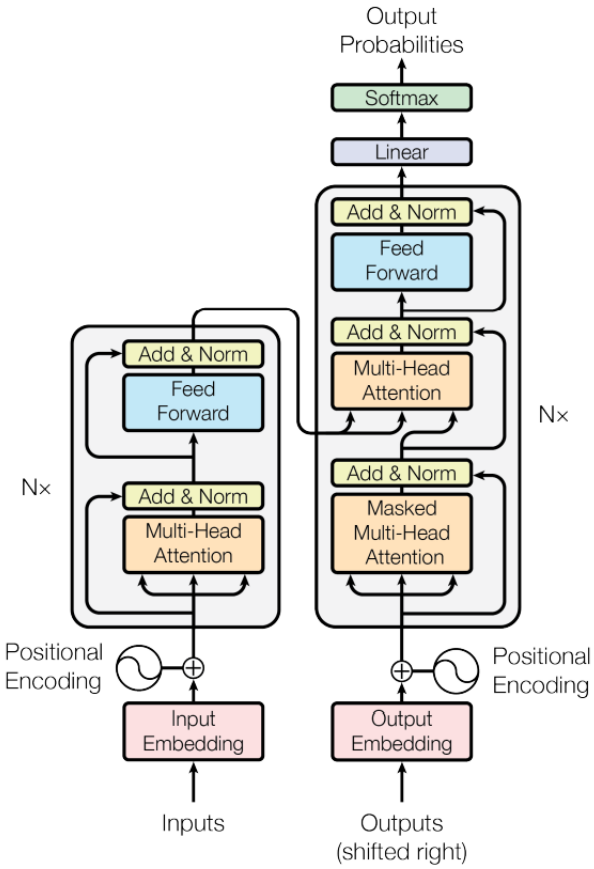


Figure 2-7: Transformer model architecture [Vaswani et al., 2017].

The entire architecture of the Transformer model is presented in Figure 2-7. From high-scale point of view, it can be divided into two parts: Encoder and Decoder.

Encoder In the original paper [Vaswani et al., 2017], the encoder is a stack of 6 similar layers consisting of 2 sublayers each: the first sublayer is multi-head attention and the second one is position-wise fully connected neural network. Inside the encoder, skip connections and LayerNorm are also used.

Decoder The decoder is similar to the encoder except for an additional sublayer which is a multi-head attention over the output of the encoder stack. Additionally, a self-attention in the decoder is modified in order to ensure that predictions at position i can depend only on known positions less than i.

2.2.3 Models Zoo

After the introduction of Transformer architecture, there appeared several new architectures based on Transformer blocks. The diversity of new models is great. In Table 2.1, we introduce the models that are used in this work. We choose these models as they achieve a lot of SOTA result on different NLP tasks.

Model	Building Blocks	Data	Training	Performance
BERT [Devlin et al., 2019]	Encoder	16GB	- Masked Language Modeling (MLM); - Next sentence prediction	SOTA on GLUE and SQuAD
RoBERTa [Liu et al., 2019a]	Encoder	160GB	MLM	Outperformed BERT
GPT-2 [Radford et al., 2019]	Decoder	40GB	Causal language modeling (CLM)	Ability to perform question answering, summarization, translation.
T5 [Raffel et al., 2020]	Encoder+ Decoder	7TB	A multi-task mixture of unsupervised and supervised tasks for which each task is converted into a text-to-text format	SOTA on many NLG tasks
BART [Lewis et al., 2020]	Encoder+ Decoder	160GB	Reconstruct corrupted texts	- Comparable to RoBERTa; - SOTA on some NLG tasks.

Table 2.1: The summarized information about different models based on the Transformer blocks used in this work.

BERT BERT (Bidirectional Encoder Representations from Transformers) model consists only of Encoder Transformer blocks [Devlin et al., 2019]. Moreover, it takes into account both left and right context during training, which makes it to be named bidirectional. The model is pre-trained on two tasks (Figure 2-8):

1. **Masked Language Modeling (MLM)**. Randomly mask 15% of tokens in each sequence. The model only predicts the missing words, but it has no information on which words have been replaced or which words should be predicted.

2. **Next sentence prediction**. Motivated by the fact that many downstream tasks involve the understanding of relationships between sentences, the model was additionally pre-trained on the task to predict if sentence B follows sentence A.

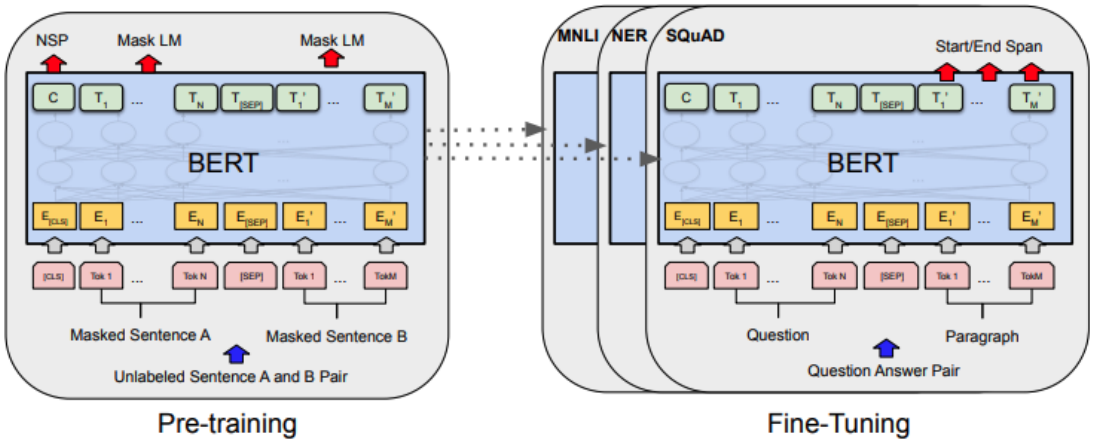


Figure 2-8: The main idea of BERT [Devlin et al., 2019] model: (i) the model is pretrained for MLM and next sentence prediction tasks on big amount of text data; (ii) after that, for a specific task, the model can be easily fine-tuned.

BERT was specifically trained on Wikipedia ($\tilde{2}$.5B words) and Google's BooksCorpus ($\tilde{8}$00M words).[1] It has different versions: **Base** (110M parameters) and **Large** (340M parameters).

[1] https://www.english-corpora.org/googlebooks

RoBERTa RoBERTa (Robustly optimized BERT approach) [Liu et al., 2019a] refers to a new way of training BERT to achieve better performance. The modifications are the following:

1. The size of training batch was increased;
2. The task of next sentence prediction was removed;
3. The sequences' length in training data format was increased;
4. The masking strategy was changed from static to dynamic during training epochs.

The model was trained on bigger corpus than BERT that consists of five datasets: Wikipedia, BookCorpus, CommonCrawl[2], OpenWebText[3], and Stories. The same as BERT, it has two versions: **Base** (125M parameters) and **Large** (355M parameters). Because of the modifications, both versions have more parameters size than corresponding BERT versions.

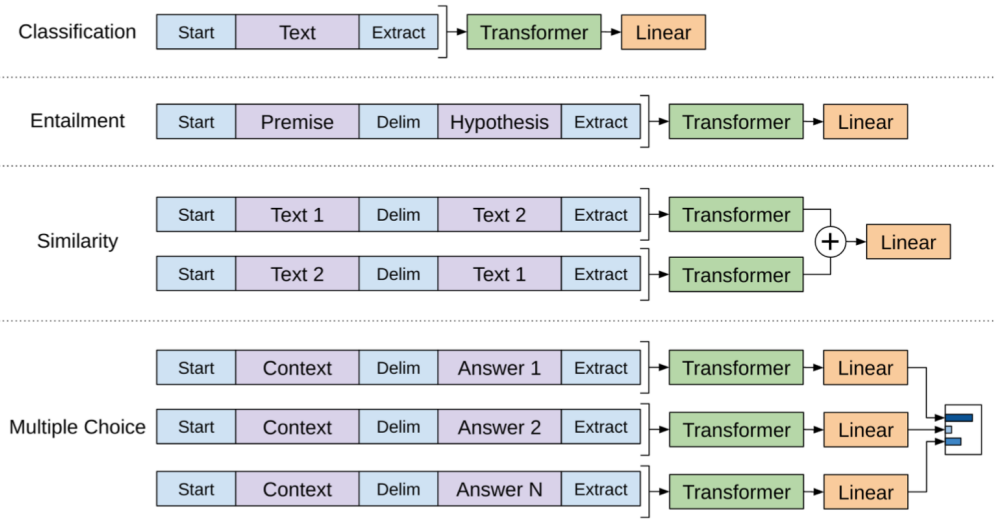


Figure 2-9: Training objectives for GPT model [Radford et al., 2019].

GPT In comparison to previous models, GPT (Generative Pre-training Transformer) consists only of Decoder Transformer blocks. The text representation is taken from

[2] https://commoncrawl.org/2016/10/news-dataset-available
[3] https://github.com/jcpeterson/openwebtext

the last decoder layer for the last token. Then, the classification model takes this representation as an input for specific task. The model was trained in different types of tasks (Figure 2-9). GPT architecture has different generations depending on the training data and the size of the parameters: **GPT** (6GB training data, 117M parameters), **GPT-2** (40GB training data, 1.5B parameters), and **GPT-3** (45TB training data, 175B parameters).

T5 While previous models use only one type of Transformer blocks, T5 (Text-to-Text Transfer Transformer) [Raffel et al., 2020] is based on the original encoder-decoder Transformer idea.

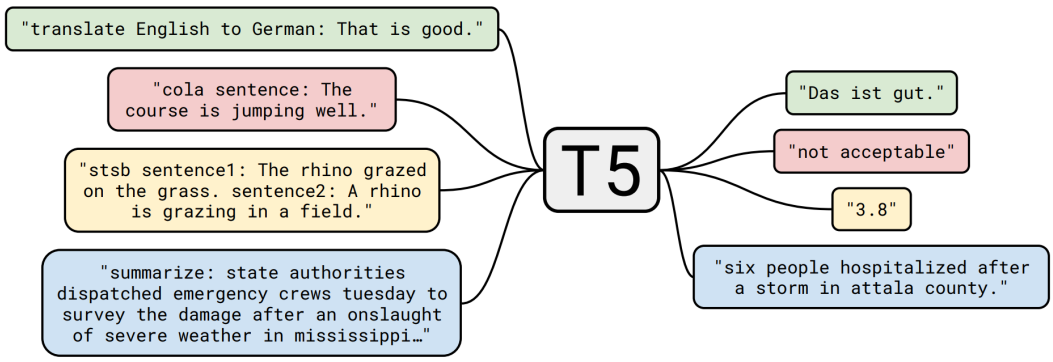


Figure 2-10: The illustration of tasks on which T5 model [Raffel et al., 2020] was pretrained.

The model is pretrained on various tasks (Figure 2-10). T5 uses short task prefixes to distinguish task intentions and fine-tunes the model separately on every individual task. All NLP problems were converted into a text-to-text format. It is trained using teacher forcing. This means that, for training, we always need an input sequence and a corresponding target sequence. The model is trained on Web corpus with various filters applied. T5 was pre-trained on Common Crawl dataset with unsupervised denoising objective and then fine-tuned on SuperGLUE [Wang et al., 2019] task.

The model has several variations: **small** (60M parameters), **base** (220M parameters), **large** (770M parameters), **t5-3b** (3B parameters), and **t5-11b** (11B parameters).

BART BART (Bidirectional and AutoRegressive Transformer) [Lewis et al., 2020] also has encoder-decoder architecture. It combines the features of the BERT and GPT models: jointly training the BERT-like bidirectional encoder and the GPT-like autoregressive decoder (Figure 2-11).

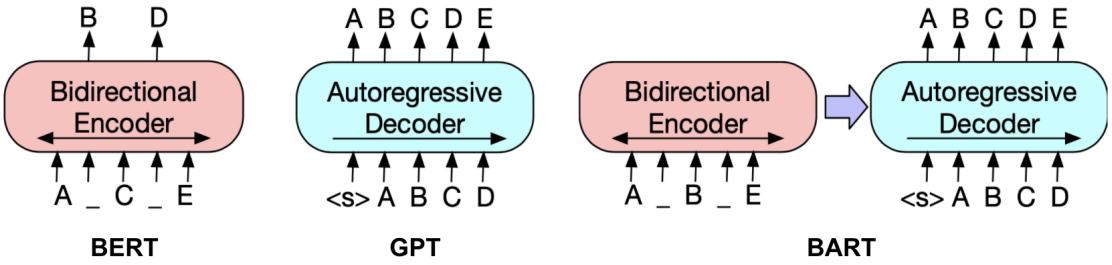


Figure 2-11: The distinguishing feature of BART [Lewis et al., 2020]: (i) it is constructed of both Encoder and Decoder blocks; (ii) it is trained on the task of reconstruction corrupted texts.

The task used for pre-training is a recovering the original text from a randomly corrupted version. In the work, several strategies for text corruption were explored including token masking, token deletion, text infilling, sentence permutation, and documentation rotation. These transformations are applied to 160GB of text from the English Wikipedia and BookCorpus dataset. The versions of the model are: **base** (139M parameters) and **large** (406M parameters).

All models achieve SOTA results in various NLP tasks. One of the key elements is pre-training. The models were trained on vast amount of data with different objectives. That allows them already to incorporate "knowledge" about the language. After pretraining, models can be quickly fine-tuned on the specific task. All these advantages are complemented by a convenient single platform for storing models and datasets – HuggingFace [Wolf et al., 2020]. The majority of all existed due today Transformer-based models and their different versions with weights are available at the platform. In this work, we use these advantages of highly performed models to fine-tune on our presented tasks. In addition, we also released all our fine-tuned models and presented datasets on the HuggingFace platform.

2.3 Multilingual Natural Language Processing

To the state of 2022, 7 151[4] languages are spoken in the world today. However, the distribution of speakers between all languages is quite different. Only 23 languages account for more than half of the world's population. At the same time, 40% of all spoken languages are endangered with fewer than 1 000 speakers remaining.

The development of multilingual NLP techniques is still ongoing. Every year there are more and more datasets and models for different purposes, which cover more and more languages. After the release of the Word2Vec model for English monolingual vector representation [Mikolov et al., 2013], there was introduced distributed word representations for 157 languages [Grave et al., 2018] trained on the mixture of Wikipedia and CommonCrawl datasets.

The recent rise of deep learning models based on the Transformer architecture [Vaswani et al., 2017] made it possible to create Large Language Model (LLM) covering several dozens or even hundreds of languages. Thus, for the transformer-based models discussed above such as BERT [Devlin et al., 2019], RoBERTa [Liu et al., 2019a], T5 [Raffel et al., 2020], BART [Lewis et al., 2020] there exist their multilingual analogues – mBERT, XLM-R [Conneau et al., 2020], mT5 [Xue et al., 2021], mBART [Tang et al., 2020]. Recently, new multilingual models appeared. Thus, one of the biggest multilingual models released in 2022 is No Language Left Behind (NLLB) [Costa-jussà et al., 2022] by Meta AI which is able of delivering high-quality translations directly between any pair of 200+ languages — including low-resource languages like Asturian, Luganda, Urdu and more. More details on which languages and datasets cover each multilingual model are represented in Table 2.2. One of the big advantages of such multilingual models is the ability to get vector representations for texts for the corresponding language for further processing.

At the same time, the quality of such vector representations for various languages can differ. For instance, in Figure 2-12 we can observe the difference in monolingual corpus parts that were used to pre-train mBART. The authors used a re-balancing

[4] https://www.ethnologue.com/guides/how-many-languages

Model	#Parameters	Dataset	#langs.	vocab.
mBERT [Devlin et al., 2019]	172M	Wikipedia	104	110K
mT5-Large [Xue et al., 2021]	1.2B	Common Crawl	101	250K
mBART-Large [Tang et al., 2020]	680M	CommonCrawl	50	250K
XLM-R-Large [Conneau et al., 2020]	559M	CommonCrawl	100	250K
BLOOM [BigScience, 2022]	176B	WuDaoCorpora	46	250K
NLLB [Costa-jussà et al., 2022]	54.5B	Flores-200	204	256K

Table 2.2: A comparison of multilingual models that can be used for various NLP tasks.

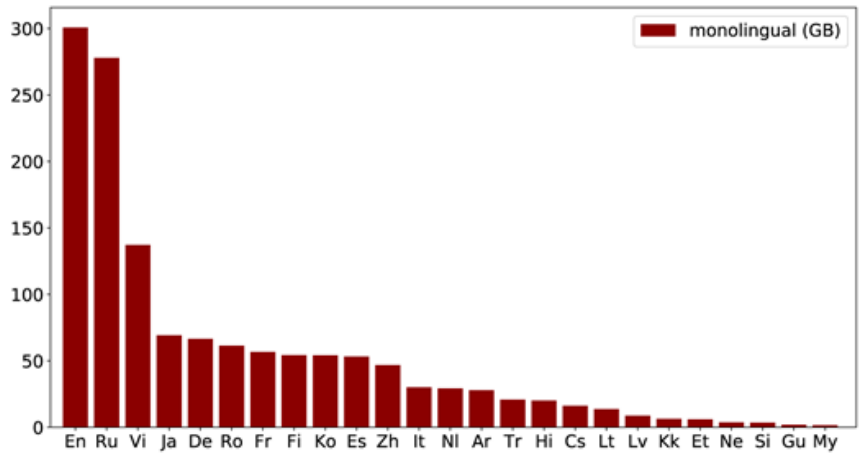


Figure 2-12: The sizes of different languages parts of CC-25 dataset used for mBART training [Liu et al., 2020]. We can see the significant difference between top-used languages and low resource ones.

strategy by up/down-sampling text from each language i with a ratio λ_i:

$$\lambda_i = \frac{1}{p_i} * \frac{p_i^\alpha}{\sum_j p_j^\alpha}, \tag{2.8}$$

where p_i is the percentage of each language in the corpus. At the same time, we can observe the difference in the quality of massive multilingual transformers' performance in cross-lingual transfer in resource-lean scenarios which was studied in [Lauscher et al., 2020]. Thus, improving data accessibility for all languages and multilingual data augmentation studies still have room for improvement. Furthermore, this disbalance in the training data distribution should be considered during the deployment of multilingual NLP transformer-based solutions in applications.

Much interest is also given to **zero-shot** cross-lingual transfer learning. The scenario can be quite realistic: we would like *tto transfer some domain knowledge from a resource-rich language to a low-resource one while the domain knowledge is*

not accessible for the last one. Multilingual large-language models can be very useful for such applications. The possibility of zero-short cross-lingual transfer learning is already possible was reviewed in [Doddapaneni et al., 2021]. There are some findings from the study on when zero-short cross-lingual transfer can succeed:

1. The source and target languages share some vocabulary;
2. There is some similarity between the source and target languages;
3. Enough pretraining data is available in the target languages;
4. The complexity of the task is less.

As can be seen, these requirements are quite strict. For most cases, zero-short cross-lingual transfer learning fails. For example, for the XNLI benchmark [Conneau et al., 2018], it was shown that training with translated data in the target language still generates more profits than when training data are available only for one language.

In the end, we can see that modern multilingual NLP models already have many possibilities for research in multiple languages. But, even the largest multilingual models cover only a small percentage of all existing languages (remainder: 204 out of 7 151 which is 3%) Still, there is a lot of work that needs to be done to make NLP research equally fair and available for all languages. In this work, we focus on exploration of already existing multilingual models for fighting with different types of harmful information. However, we hope that the development of more stable multilingual models will open new horizons in the future for the presented research.

Part I

Methods for Fake News Detection

3

Task Introduction

This part is dedicated to answer research question **Q1**: we explore if fake news detection systems can benefit from signals from news written in multiple languages. The contributions of this part are the follows:

1. We present new **multilingual feature** for fake news classification.
2. We show that the proposed feature significantly improves performance of previous fake news classification systems achieving SOTA results on several multi-domain datasets.
3. We explore new methods for **multilingual and cross-lingual news similarity measurement**.

3.1 Task Motivation

After the manipulation of opinions on Facebook during the 2016 U.S. election [Allcott and Gentzkow, 2017], the interest in the topic of fake news has increased substantially. Unfortunately, the distribution of fakes leads not only to misinformation among readers but also to more severe consequences. There was a case of the spread of rumors about Hillary Clinton leading child sex trafficking led to Washington Pizzeria [Kang and Goldman, 2016]. Moreover, due to the global pandemic in 2020, there was a simultaneous emergence of an infodemic [Alam et al.] that could lead to an even worse epidemiological situation and harm people's health dramatically.

Figure 3-1: The example how one event can be described differently by mass media in different languages.

As a result, fake news received tremendous public attention and drew increasing interest from the academic community. Multiple supervised fake news classification models were proposed based on linguistic features [Pérez-Rosas et al., 2018, Patwa et al., 2020]; deep learning models [Barrón-Cedeno et al., 2019, Glazkova et al., 2020, Kaliyar et al., 2021, Gundapu and Mamid, 2021]; or signals from social networks [Nguyen et al., 2020, Shu et al., 2019a]. One of the directions of the supervised approaches is to use additional information from the Web [Popat et al., 2017, Karadzhov et al., 2017, Ghanem et al., 2018]. However, in these works only monolingual signals were taken into account.

The world-changing situations showed that a single event can be described differently by mass-media in different countries (Figure 3-1). The cross-lingual comparison between such news from different languages can be a strong signal to detect fake news. Such processing of news from different countries in different languages already carries an additional filter and verification of news by several specialists in the field of journalism simultaneously. For this reason, we want to fill the gap of only monolingual evidence from the Web usage and propose a new feature for fake news detection based on cross-lingual news comparison.

3.2 Problem Statement

A lot of systems have been created dedicated to the different steps of the fake news detection pipeline. The general approach for the fake news detection is illustrated in Figure 3-2. Usually, previous works focused on *Information checking* step where all classification tasks are appearing.

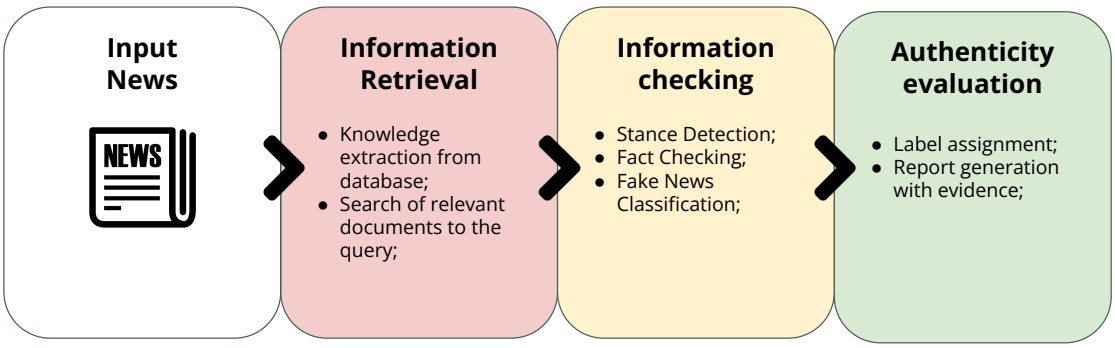


Figure 3-2: High-level illustration of a general pipeline of fake new detection system.

Thus, *Information checking* tasks can be divided into several types:

1. **Stance Detection**: find a classifier

$$f : (d, h) \mapsto s \tag{3.1}$$

that predicts one of four stance labels $s \in S = \{\text{Agree, Disagree, Discuss, Unrelated}\}$ for a document d with respect to a headline h.

2. **Fact Checking**: find a classifier

$$f : (h, D) \mapsto v \tag{3.2}$$

that predicts one of three verdicts $v \in V = \{\text{Supported, Refuted, Unrelated}\}$ for a news headline h given a database of documents D. This database can be used to search for facts and comparison of information with facts in a headline.

3. **Fake News Classification**: find a classifier

$$f : (n, F) \mapsto c \tag{3.3}$$

that predicts class $c \in C = \{\text{Fake, Legit}\}$ for a news n based on feature set F. Features can be different: for instance, linguistic features from news headlines and main content or news propagation graph in some social networks.

In our work specifically, we want to focus on **Fake News Classification** task. However, we want to extend the usual definition of a feature set F that previously only focused on one language and extend it to the multilingual case.

3.3 Related Work

A substantial amount of research has been done in the field of fake news detection, which includes the creation of datasets and methods. In this section, we perform a comprehensive analysis of the prior art related to the subject of this article.

3.3.1 Users Behaviour Towards Fake News Detection

Firstly, before the discussion of automatic machine fake news detection methods, we want to analyze the case of how real-life users react to fake information and in which way they check the veracity of information.

In [Lewandowsky et al., 2012] a very broad analysis of users' behavior was obtained. The authors found out that when people try to check information credibility they rely on a limited set of features, such as:

- Is this information compatible with other things I believe to be true?
- Is this information internally coherent? Do the pieces form a plausible story?
- Does it come from a credible source?
- Do other people believe it?

So, people can rely on the text of the news and its source and their judgment. However, if they get enough internal motivation, they can also refer to some external sources for evidence seeking. These external sources can be some knowledge sources or other people.

The conclusions from [Tandoc Jr et al., 2018] repeat the previous results: individuals rely on both their judgment of the source and the message, and when this does not adequately provide a definitive answer, they turn to external resources to authenticate news. The intentional and institutional reaction was seeking confirmation from institutional sources (some respondents answered simply "Google").

Also, several works have been done to explore the methods to combat received by users fake information and convince them with true facts. In [Ecker et al., 2017] it was shown that explicitly emphasizing the myth and even its repetition with refutation help users to pay attention and memorize the truth. Moreover, participants that received messages across different media platforms [Zhao, 2019] and different perspectives of the information [Geeng et al., 2020] showed greater awareness of news evidence. Consequently, the information from the external search is an important feature for news authenticity evaluation and evidence seeking. Also, a different perspective from different media adds more confidence in the decision-making process.

3.3.2 Fake News Detection Datasets

To leverage the task of automatic fake news detection there have been created several news datasets focused on misinformation, each with a different strategy of labeling.

The Fake News Challenge[1] launched in 2016 was a big step in identifying fake news. The task of FNC-1 was stance detection type task [Hanselowski et al., 2018]. The dataset consists of 300 topics, with 5–20 news articles for each. In general, it consists of 50K labeled claim-article pairs. The dataset is derived from the Emergent project [Silverman, 2017].

Another publicity available dataset is **LIAR** [Wang, 2017]. In this dataset 12.8K manually labeled short statements in various contexts from PolitiFact.com[2] were collected. They covered such topics as news releases, TV or radio interviews, campaign speeches, etc. The labels for news truthfulness are fine-grained in multiple classes: pants-fire, false, barely-true, half-true, mostly true, and true.

[1] http://www.fakenewschallenge.org
[2] https://www.politifact.com

Claim verification is also related to Fact Extraction and VERification dataset (**FEVER**) [Thorne et al., 2018]. 185,445 claims were manually verified against the introductory sections of Wikipedia pages and classified as SUPPORTED, REFUTED, or NOTENOUGHINFO. For the first two classes, the annotators also recorded the sentences forming the necessary evidence for their judgment.

FakeNewsNet [Shu et al., 2018] contains two comprehensive datasets that includes news content, social context, and dynamic information. Moreover, as opposed to all the datasets described above, in addition to all textual information, there is also a visual component saved in this dataset. All news were collected with PolitiFact and GossipCop[3] crawlers. In general, 187014 fake and 415645 real news were crawled.

Another collected for supervised learning dataset is **FakeNewsDataset** [Pérez-Rosas et al., 2018]. The authors did a lot of manual work to collect and verify the data. As a result, they managed to collect 240 fake and 240 legit news on six different domains – sports, business, entertainment, politics, technology, and education. All news samples are for the 2018 year.

One of the latest large datasets is **NELA-GT-2018** [Nørregaard et al., 2019]. In this dataset authors tried to overcome some limitations that can be observed in previous works: *1) Engagement-driven* – the majority of the datasets, both for news articles and claims, contain only data that has been highly engaged with on social media or has received attention from fact-checking organizations; *2) Lack of ground truth labels* – all of the current large-scale news article datasets do not have any form of labeling for misinformation research. To overcome these limitations, they gathered a wide variety of news sources from varying levels of veracity and scraped article data from the gathered sources' RSS feeds twice a day for 10 months in 2018. As a result, a new dataset was created consisting of 713,534 articles from 194 news and media producers.

Due to the events of 2020, the work has been already done in the direction of the creation COVID-19 fake news detection dataset. **COVID-19 Fake News** [Patwa et al., 2020] was built based on the information from public fact-verification

[3] https://www.gossipcop.com

Dataset	Task	Language
FNC-1 [Hanselowski et al., 2018]	Stance Detection	English
Arabic Claims Dataset [Hasanain et al., 2019]		Arabic
FEVER [Thorne et al., 2018]	Fact Checking	English
DanFEVER [Nørregaard and Derczynski, 2021]		Danish
LIAR [Wang, 2017]	Fake News Classification	English
FakeNewsNET [Pérez-Rosas et al., 2018]		
FakeNewsDataset [Pérez-Rosas et al., 2018]		
NELA-GT-2018 [Nørregaard et al., 2019]		
ReCOVery [Zhou et al., 2020b]		
GermanFakeNC [Vogel and Jiang, 2019]		German
The Spanish Fake News Corpus [Posadas-Durán et al., 2019]		Spanish

Table 3.1: The datasets covered in related work. It can be observed that the majority of the data for different fake news detection tasks is for the English language.

websites and social media. It consists of 10,700 tweets (5600 real and 5100 fake posts) connected with the COVID-19 topic. In addition, there was created **ReCOVery** [Zhou et al., 2020b] multimodal dataset. It also incorporates in itself 140,820 labeled tweets as well as 2,029 news articles on coronavirus collected from reliable and unreliable resources.

However, all of the above datasets have one main limitation – they are monolingual and dedicated only to the English language. Talking about other languages other than English, such datasets can be mentioned: *French satiric dataset* [Liu et al., 2019b], *GermanFakeNC* [Vogel and Jiang, 2019], *The Spanish Fake News Corpus* [Posadas-Durán et al., 2019], *Arabic Claims Dataset* [Hasanain et al., 2019]. However, all of these datasets are monolingual as well and mostly cover fake news

classification tasks missing, for instance, fact verification and evidence generation problems. There was only collected *A Multilingual Cross-domain Fact Check News Dataset for COVID-19* [Shahi and Nandini] that covers 40 languages from 105 countries (English, Spanish, French, Portuguese, Hindi languages, and others). However, this dataset is highly imbalanced. Firstly, there is a disbalance in terms of fake (4132 samples) and true (1050 samples) labels. Secondly, the number of English samples is significantly bigger than for other languages: for the top first English language, there are over 2000 samples, for the top second Spanish there are almost 1000 samples, for the top third French language there are only 250 samples, and further data size for other languages decreases dramatically. All these statistics also illustrate the difficulties of collecting multilingual fake news datasets. Consequently, the creation of a supervised dataset for each language and implementation algorithm of fake news detection for each language will be a very resource- and time-consuming task.

3.3.3 Fake News Classification Methods

On the basis of previously described datasets, several solutions were created to tackle the problem of obtaining such a classifier. The feature sets used in all existing methods can be divided into two categories: 1) **internal** features that can be obtained by different preprocessing strategies and linguistic analysis of the input text; 2) **external** features that are extracted from some knowledge base, the Internet or social networks and give additional information about the facts from the news, its propagation in social media and users reactions.

Methods based on Internal Features

One of the types of features that are helpful in fake news classification tasks is linguistic and psycholinguistic features. In [Pérez-Rosas et al., 2018] a strong baseline model based on such a feature set was created based on the FakeNewsDataset. The feature set used in this work looks as follows:

- **Ngrams**: tf-idf values of unigrams and bigrams from a bag-of-words representation of the input text.

- **Punctuation** such as periods, commas, dashes, question marks, and exclamation marks.

- **Psycholinguistic features** extracted with LIWC lexicon. Alongside some statistical information, LIWC also provides emotional and psychological analysis.

- **Readability** that estimates the complexity of a text. The authors use content features such as number of characters, complex words, long words, number of syllables, word types, and others. In addition, they used several readability metrics, including the Flesch-Kincaid, Flesch Reading Ease, Gunning Fog, and Automatic Readability Index.

- **Syntax**: a set of features derived from production rules based on context-free grammar (CFG) trees.

Based on such features, different statistical machine learning models can be trained. In [Pérez-Rosas et al., 2018] the authors trained the SVM classifier according to the set of characteristics presented. Naïve Bayes, Random Forest, KNN, and AdaBoost were also frequently used as fake news classification models [Choudhary and Arora, 2021, Sharma et al., 2019, Gravanis et al., 2019].

In [Ghanem et al., 2020] the perspective of the usage of emotional signals extracted from the news text for detecting fakes was shown. The authors analyzed the set of emotions that are present in true and fake news checking the hypothesis that trusted news does not use emotions to affect the reader's opinion while the fake one does. They found out that such emotions as *negative emotions*, *disgust*, *surprise* have more tendency to appear in fake news and can give a strong signal for fake news classification.

Additionally to linguistic features, feature extraction strategies based on deep learning architectures were also explored. In [Kaliyar et al., 2020] the classical architecture for text classification task based on CNN was successfully applied for the fake news detection task. With the recent growth of the usage of Transformer architectures in the NLP field, such models as BERT [Kaliyar et al., 2021, Jwa

et al., 2019] and RoBERTa [Glazkova et al., 2020] also demonstrated high results for general-topic fakes classification as well as COVID-19 fake news detection task.

As it can be seen, one of the main advantages of models based on internal feature sets is that such models are quite easy to use and they do not require significant additional time for feature extraction. Moreover, such models can be optimal in terms of inference time and memory usage because they only operate with internal information from input news. However, if we consider the explainability aspect for the end users, the evidence generated from such internal features most likely will be not enough to convince the user of the correctness of model performance and to motivate the label decision for the news.

Methods based on External Non-Textual Features

Although internal features-based models can achieve high classification scores in the fake news classification task, the decision of such is hard to interpret. As a result, additional signals from external sources can add more confidence to model decision reasoning.

If the news appears in some social network, the information about the users that liked or reposted the news post and the resulted post propagation can be used as a feature for fake news classification. It was shown in [Zhao et al., 2020] that fake news spread over social networks quicker after the publication than true news. As a result, to combat fake news in the early stages of its appearance, several methods have been created to detect the anomaly behavior in reposts or retweets [Liu and Wu, 2018, Shu et al., 2019b]. In [Shu et al., 2019c] the different information about specific users was explored. The author extracted location, profile image, and political bias to create a feature set.

Another type of information that can be obtained from users and be used as some kind of knowledge base is users' comments related to the news post. This approach was explored in [Shu et al., 2019a]. There was created *dEFEND* system for explainable fake news detection. The information from users' comments was used to find related evidence and validate the fact from the original news. *Factual News Graph (FANG)* system from [Nguyen et al., 2020] was presented to connect

the content of news, news sources, and user interaction to build a full-filled social picture about the inspected news.

Talking about the information verification step in the fake news detection pipeline, there were created several methods for leveraging a fact-checking task. One of the sources for providing a knowledge base with evidence is Wikipedia. The FEVER dataset that was previously discussed in Section 3.3.2 consists of claims and evidence already pre-extracted from Wikipedia. Several works like [Soleimani et al., 2020, Atanasova et al., 2020, Nie et al., 2019] are dedicated to the fact-checking task and evidence generation based on Wikipedia pages.

On the other hand, the knowledge base for obtaining evidence for information verification can be simply the Web. In [Popat et al., 2017, Karadzhov et al., 2017, Ghanem et al., 2018, Li and Zhou, 2020] the authors referred to the Web search (Google or Bing) to collect relevant articles and use such scraped information as an external feature to build fake news classifier. As it was discussed in Section 3.3.1, such a Web-based feature is quite motivated by real-life users' behavior. As a result, the generated evidence based on the Web scraped information can be more persuasive for the users as it automatizes the steps that they take to check the veracity of the news.

However, in all the discussed methods we can also see the usage of only one language for evidence granting. The systems that used Web search for evidence extraction turned to only English search results. In our work, we want to fill this gap to explore cross-lingual Web-based evidence for the fake news classification task.

4

Fake News Detection using Multilingual Evidence

This chapter describes the proposed method for fake news detection based on the usage of multilingual evidence. The contributions of this chapter are the following:

- Multiverse: the new **cross-lingual evidence feature** for fake news detection based on multilingual news verification is proposed.
- The manual experiment based on cross-lingual dataset markup to evaluate if the user can use the such feature for misinformation identification is conducted.
- Fake news classification systems are compared based on the proposed feature with several baselines that achieve SOTA results.
- The best models with the integrated cross-lingual feature are investigated in terms of explainability, showing examples of how extracted cross-lingual information can be used for evidence generation.

The code of the proposed method is available online.[1]

[1] https://github.com/skoltech-nlp/multilingual-fake-news

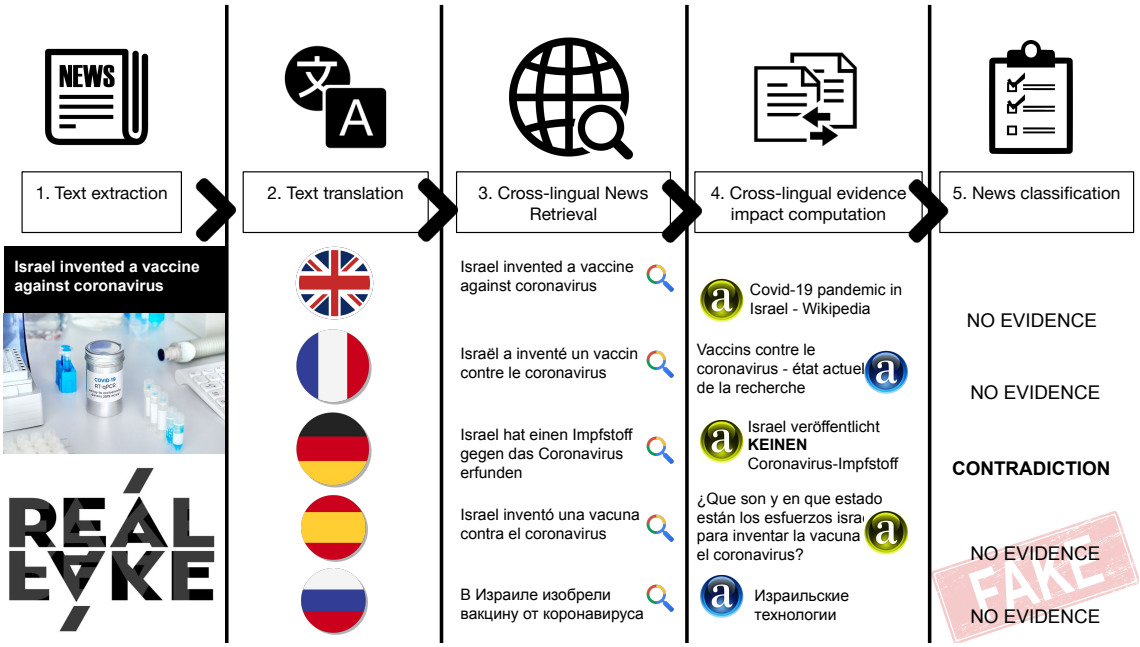


Figure 4-1: Overview of our approach: checking for fake news based on cross-lingual evidence (CE).

4.1 Multiverse: A New Feature for Fake News Classification

We present Multiverse – Multilingual Evidence for Fake News Detection based on extraction from Web search. The idea is motivated by the user experience illustrated in Section 3.3.1 and the lack of multilingualism in automatic fake news detection methods, as discussed in Section 3.3.3. Users quite often refer to the Web search to check news seen in some news feed. However, to show the different points of view and additional information out of a monolingual bubble, the cross-lingual check of original news can be quite persuasive and can give a larger room for rational judgment about information.

Our proposed approach is based on the following hypothesis:

Hypothesis 1 (H1)

- *If the news is true, then it will be widespread in different languages and also across media with different biases, and the facts mentioned should be identical.*
- *If the news is fake, it will receive a lower response in the foreign press than a*

piece of true news.

The step-by-step pipeline of the approach, schematically represented in Figure 4-1, is as follows:

- **Step 1. Text extraction:** As a new article arrives, the title and content are extracted from it.
- **Step 2. Text translation:** The title is translated into target languages and new search requests are generated.
- **Step 3. Cross-lingual news retrieval:** Based on generated cross-lingual request – translated title – the search with a Web search engine is executed.
- **Step 4. Cross-lingual evidence impact computation:** Top-N articles from search results are extracted to assess the authenticity of the initial news. The information described in the news is compared with the information in the articles from the search result. Also, the ranks of the source of the extracted articles are taken into account. The number of articles that confirms or disproves the original news from reliable sources is estimated.
- **Step 5. News classification:** Based on the information from the previous step, the decision is made about the authenticity of the news. If the majority of results support the original news, then it is more likely to be true; if there are contradictions – it is a signal to consider the news as a fake.

As we can see from the example in the scheme in Figure 4-1, for the news *"Israel invented a vaccine against coronavirus"* the majority of the scraped articles provide no evidence that supported incoming news. Moreover, there is an article with high reliability that provides an explicit refutation of the original information. As there is none of the supporting information and a contradiction with the scraped information, the probability that we should believe in the veracity of this news is quite low.

The proposed method based on cross-lingual evidence extraction can work properly with worldwide important news. Indeed, if there is some local event about locally famous parties, in the majority of cases such news will be doubtfully widespread

all over the Internet. As a result, in our future assumptions and experiments, we take into consideration datasets and news that cover worldwide events.

To incorporate the proposed feature into an automatic fake new detection pipeline, firstly, we wanted to lean on user experience and check the following hypothesis:

Hypothesis 2 (H2) *The person can detect fake news using cross-lingual evidence using the pipeline presented in Figure 4-1.*

After this hypothesis confirmation, we can explore the possibilities to automate fake news classification using the cross-lingual evidence feature confirming the next hypothesis:

Hypothesis 3 (H3) *The proposed cross-lingual evidence feature can improve automatic fake news detection.*

To confirm all the above hypotheses we conducted several experiments. For all experiments, we chose top-5 European languages spoken in Europe[2] and used in Internet[3] – English, French, German, Spanish, and Russian – to obtain cross-lingual evidence. For the search engine, we stopped at Google search[4] as it is the top-1 search engine in the world[5] and also claimed to be widely used by users during use case fake news check experiment mentioned in Section 3.3.1.

The first experiment is a manual small-scale study confirming Hypothesis 1 and Hypothesis 2. After that, we tested several approaches to automatize the pipeline and compared them with manual markup (Section 4.2). The final step (Section 4.3) of the confirmation of Hypothesis 3 is an automated fake news detection system tested on several fake news datasets: we implemented our cross-lingual evidence feature and compared it with several baselines achieving SOTA on all datasets.

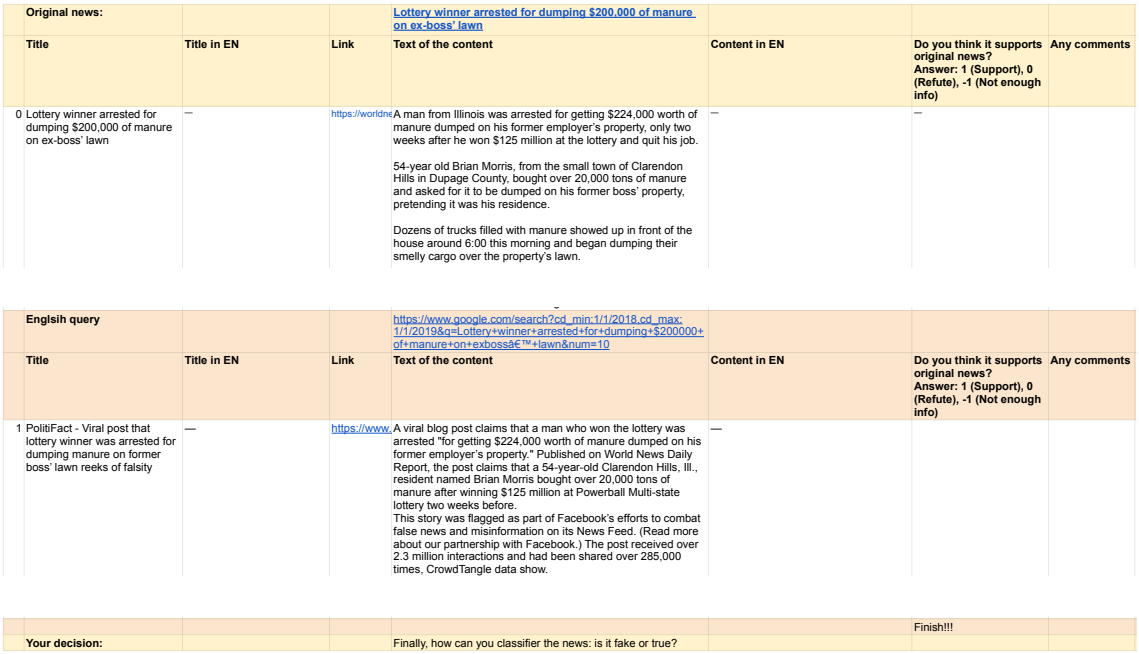
[2] https://www.justlearn.com/blog/languages-spoken-in-europe
[3] https://www.statista.com/statistics/262946/share-of-the-most-common-languages-on-the-internet
[4] https://www.google.com
[5] https://www.oberlo.com/blog/top-search-engines-world

News title	URL	Label
Lottery winner arrested for dumping $200,000 of manure on ex-boss' lawn	https://worldnewsdailyreport.com/lottery-winner-arrested-for-dumping-200000-of-manure-on-ex-boss-lawn/	Fake
Woman sues Samsung for $1.8M after cell phone gets stuck inside her vagina	https://worldnewsdailyreport.com/woman-sues-samsung-for-1-8m-after-cell-phone-gets-stuck-inside-her-vagina/comment-page-58/	Fake
BREAKING: Michael Jordan Resigns From The Board At Nike-Takes 'Air Jordans' With Him	https://www.newsbreak.com/news/944830700924/breaking-michael-jordan-resigns-from-the-board-at-nike-takes-air-jordans-with-him	Fake
Donald Trump Ends School Shootings By Banning Schools	https://www.8shit.net/donald-trump-ends-school-shootings-banning-schools/	Fake
New mosquito species discovered that can get you pregnant with a single bite	https://thereisnews.com/new-mosquito-species-discovered-can-make-you-pregnant/	Fake
Obama Announces Bid To Become UN Secretary General	https://www.pinterest.com/pin/465630048969491948/	Fake
Lil Tay Rushed To Hospital After Being Beat By Group Of Children At A Playground	https://www.huzlers.com/lil-tay-rushed-to-hospital-after-being-beat-by-group-of-children-at-a-playground/	Fake
Post Malone's Tour Manager Quits Says Post Malone Smells Like Expired Milk And Moldy Cheese	https://www.huzlers.com/post-malones-tour-manager-quits-says-post-malone-smells-like-expired-milk-and-moldy-cheese/	Fake
Putin: Clinton Illegally Accepted $400 Million From Russia During Election	https://newspunch.com/putin-clinton-campaign-400-million-russia/	Fake
Elon Musk: 99.9% Of Media Is Owned By The 'New World Order'	https://newspunch.com/elon-musk-media-owned-new-world-order/	Fake
Scientists Develop New Method to Create Stem Cells Without Killing Human Embryos	https://www.christianpost.com/news/scientists-develop-new-method-to-create-stem-cells-without-killing-human-embryos.html	Legit
Luis Palau Diagnosed With Stage 4 Lung Cancer	https://cnnw.com/luis-palau-diagnosed-with-stage-4-lung-cancer/	Legit
1st black woman nominated to be Marine brigadier general	https://edition.cnn.com/2018/04/12/politics/marine-corps-brigadier-general-first-black-female/index.html	Legit
Disney CEO Bob Iger revealed that he seriously explored running for president	https://www.businessinsider.com/disney-ceo-bob-iger-says-he-considered-running-for-president-oprah-pushed-2018-4	Legit
Trump Has Canceled Via Twitter His G20 Meeting With Vladimir Putin	https://www.buzzfeednews.com/article/emilytamkin/trump-g20-putin-russia	Legit
US Mexico and Canada sign new USMCA trade deal	https://www.dw.com/en/us-mexico-canada-sign-usmca-trade-deal/a-51613992	Legit
Afghanistan Women children among 23 killed in US attack UN	https://www.aljazeera.com/news/2018/11/30/afghanistan-women-children-among-23-killed-in-us-attack-un	Legit
UNESCO adds reggae music to global cultural heritage list	https://www.aljazeera.com/features/2018/11/29/unesco-adds-reggae-music-to-global-cultural-heritage-list	Legit
The Saudi women detained for demanding basic human rights	https://www.aljazeera.com/news/2018/11/29/the-saudi-women-detained-for-demanding-basic-human-rights/	Legit
Georgia ruling party candidate Zurabishvili wins presidential runoff	https://www.aljazeera.com/news/2018/11/30/ex-envoy-wins-georgia-presidency-vote-to-be-challenged	Legit

Table 4.1: The manually selected 20 news dataset (10 fake and 10 true news) for manual experiment. Fake news were selected from the top 50 fake news of 2018 according to BuzzFeed. Legit news were selected from NELA-GT-2018 dataset.

4.2 Experiment 1: Manual Verification

To confirm Hypothesis 1 and Hypothesis 2 we conducted an experiment with manual markup where the annotators were asked to classify fake news based on cross-lingual



	Original news:			Lottery winner arrested for dumping $200,000 of manure on ex-boss' lawn			
	Title	**Title in EN**	**Link**	**Text of the content**	**Content in EN**	**Do you think it supports original news? Answer: 1 (Support), 0 (Refute), -1 (Not enough info)**	**Any comments**
0	Lottery winner arrested for dumping $200,000 of manure on ex-boss' lawn	—	https://worldne	A man from Illinois was arrested for getting $224,000 worth of manure dumped on his former employer's property, only two weeks after he won $125 million at the lottery and quit his job. 54-year old Brian Morris, from the small town of Clarendon Hills in Dupage County, bought over 20,000 tons of manure and asked for it to be dumped on his former boss' property, pretending it was his residence. Dozens of trucks filled with manure showed up in front of the house around 6:00 this morning and began dumping their smelly cargo over the property's lawn.	—	—	

	Englsih query			https://www.google.com/search?cd_min:1/1/2018,cd_max:1/1/2019&q=Lottery+winner+arrested+for+dumping+$200000+of+manure+on+exbossâ€™+lawn&num=10			
	Title	**Title in EN**	**Link**	**Text of the content**	**Content in EN**	**Do you think it supports original news? Answer: 1 (Support), 0 (Refute), -1 (Not enough info)**	**Any comments**
1	PolitiFact - Viral post that lottery winner was arrested for dumping manure on former boss' lawn reeks of falsity	—	https://www.	A viral blog post claims that a man who won the lottery was arrested "for getting $224,000 worth of manure dumped on his former employer's property." Published on World News Daily Report, the post claims that a 54-year-old Clarendon Hills, Ill., resident named Brian Morris bought over 20,000 tons of manure after winning $125 million at Powerball Multi-state lottery two weeks before. This story was flagged as part of Facebook's efforts to combat false news and misinformation on its News Feed. (Read more about our partnership with Facebook.) The post received over 2.3 million interactions and had been shared over 285,000 times, CrowdTangle data show.	—		

						Finish!!!	
	Your decision:			Finally, how can you classifier the news: is it fake or true?			

Figure 4-2: User interface that was used for annotators answer collection for manual verification. The annotator was provided with original news and the link to the source. After that he was given the results of cross-lingual search results with translation into English if needed. For each news from search result the title, link to the source, and text of the content were provided. The task of the annotator was to identify if the scraped news supported, refuted the original news or provided not enough information to make a decision. As a final step, the annotator was asked to do the classification of the original news into fake or true.

evidence.

4.2.1 Dataset

For fake news examples, we used the list of top 50 fake news from 2018 according to BuzzFeed.[6]. For true news, we used NELA-GT-2018 dataset [Nørregaard et al., 2019]. We manually selected 10 fake and true news. We tried to cover several topics in this dataset: celebrities, science, politics, culture, and the world. The full dataset featuring 20 news used for the manual markup is provided in Table 4.1.

4.2.2 Experimental Setup

As nowadays Google provides personalized search results[7], we precalculated **Step 2** and **Step 3** for annotators convenience and reproducibility. We generated cross-lingual requests in five languages – English, French, German, Spanish, and Russian. For translation from English, the Google Translation service was used. As the news are of 2018, the time range of every search was limited only to this year. For the cross-lingual search, the translated titles were used. From search results, we used the first page of the search which consisted of 10 news. As a result, for 20 news for each of all languages we got 1000 pairs of "original news ↔ scraped news" to markup.

We asked 6 annotators to take part in the experiment: manually conduct **Step 4**: cross-lingual evidence impact computation. For this, we created an interface for the markup presented in Figure 4-3. For each piece of news, we provide information about its title, content, and link to the source. As a result, every annotator could evaluate the quality of the text, the credibility of the source, and cross-lingual evidence for each sample from the dataset.

Every annotator got 10 randomly selected news, as a result, we got each news cross-checked by 3 annotators. All non-English pieces of news were translated into English. For each pair "original news ↔ scraped news" the annotator provided one of three answers: 1) **Support**: the information in the scraped news supports the original news; 2) **Refute**: the information is opposite or differ from the original news or there is an explicit refutation; 3) **Not enough info**: the information is not relevant or not sufficient to support/refute the original news. Finally, at the end of the annotation of a news, the annotator was asked to conduct **Step 5** of the pipeline and classify the news as fake or true.

4.2.3 Discussion of Results

Based on the collected annotations, for each news, we chose the final label based on the majority voted. We estimated confidence in the annotators' agreement with

[6]https://github.com/BuzzFeedNews/2018-12-fake-news-top-50
[7]http://googlepress.blogspot.com/2004/03/google-introduces-personalized-search.html

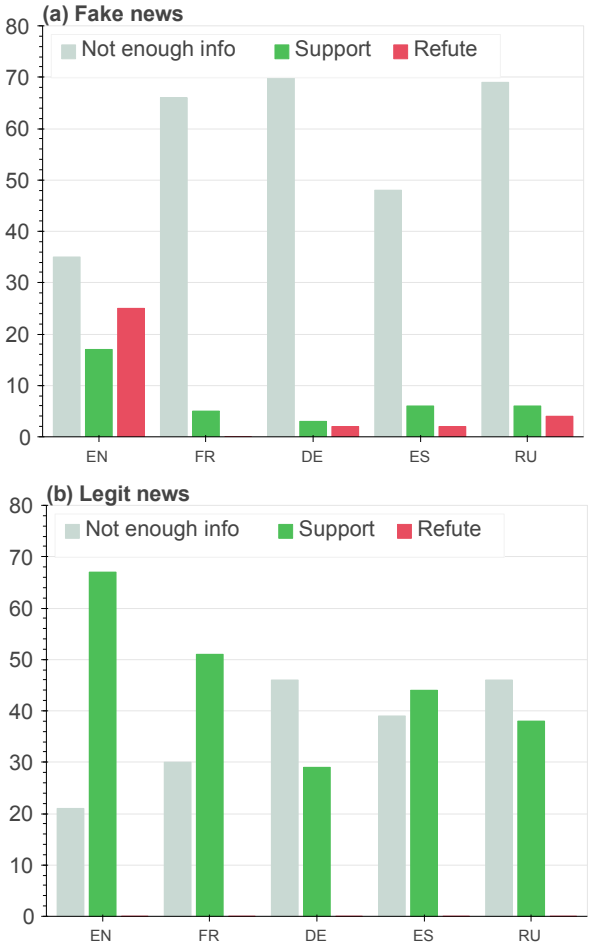


Figure 4-3: The results of manual annotation: the distribution of annotators answers for fake (a) and legit (b) news. As we can see, the amount of Support news from search results for every language for legit news incredibly overcome the amount for fake news. At the same time, there is almost none of Refute news for legit news while Refute news appeared in the search results for fake news across all languages.

Krippendorff's alpha ($\alpha = 0.83$). After that, we calculated the distribution of each type of annotator's answers for the top 10 search results by languages for fake and true news separately. The results are provided in Figure 4-3.

As we can see, the distribution of labels for true news significantly differs from the distribution for fake ones: the number of supporting articles is enough for almost every language. At the same time, for fake news, we got more refuting signals than supporting the English language and little or no evidence or relevant information dissemination for other languages. The obtained result can be used for Hypothesis 1 confirmation: the fake news indeed received less spread over different languages, while for true news we can see supportive information from multilingual sources.

Finally, the average accuracy of annotators classification is 0.95. That confirms our Hypothesis 2: a person can distinguish fake news based on cross-lingual evidence.

4.3 Experiment 2: Automatic Verification

After the manual verification of the proposed feature, we conducted the chain experiments to validate Hypothesis 3. To achieve that, we automated all the steps of the pipeline presented in Section 4.1. We experimented with several approaches for cross-lingual evidence feature computation and compared the implementations with annotators markup obtained in Section 4.2. After that, we incorporated our feature in an automated fake news detection pipeline comparing with baseline methods.

4.3.1 Automatic Cross-lingual Evidence Feature

Firstly, we implemented the cross-lingual evidence feature according to the steps of the pipeline described in Section 4.1. We implemented Algorithm 1 that automatically extracts cross-lingual evidence features for input news.

Cross-lingual evidence retrieval

To automate **Step 2**: *Text translation*, we used Googletrans[8] library. For the translation, we used five languages as well: English, French, German, Spanish, and Russian. To execute **Step 3**: *Cross-lingual News Retrieval*, the Google Search API[9] was used. As in the manual experiment, we generated the queries as the translated titles of the original news and extracted only the first page of the search result which gave us 10 articles for each language.

Content similarity computation

The goal of **Step 4**: *Cross-lingual evidence impact computation* is to figure out if the information in scraped articles supports or refutes the information from the original news. To compute this measurement we tested two strategies: 1) similarity

[8] https://pypi.org/project/googletrans
[9] https://pypi.org/project/Google-Search-API

Algorithm 1 Multilingual Evidence for Fake News detection: feature extraction.
Input: news information n, languages to use for comparison $l \in L$ the maximum amount of news from Web search to compare with N
Output: cross-lingual evidence feature set (s_i, a_i) of similarity with the original news and source credibility rank for each news w_i from multilingual web search.

function COSINE_DISTANCE_NEWS_SIMILARITY(n, w, l)
  if $type(w)$ isnot $text$ **then**
    $news\_pair\_similarity = 0$
  end if
  if $[l(\text{“fake”}), l(\text{“false”}), l(\text{“lie”})] \in w$ **then**
    $news\_pair\_similarity = 0$
  end if
  $news\_pair\_similarity = \text{cosine\_distance}(\text{mBERT}(n), \text{mBERT}(w))$
  return $news\_pair\_similarity$
end function

function NLI_NEWS_SIMILARITY(n, w, l)
  $news\_pair\_similarity = \text{XNLI-RoBERTa}(n, w)$
  return $news\_pair\_similarity$
end function

function MULTIVERSE(n, L, N)
  $cross\_lingual\_evidence := []$
  for $l \in L$ **do**
    $headline_l = \text{Translate}(n[headline], lang = l)$
    $W = \text{Search}(headline_l, top = N)$
    for $w \in W$ **do**
      $source\_rank = \text{AlexaRank}(w)$
      # For similarity score cosine- or nli-based function can be chosen
      $similarity = \text{cross\_lingual\_news\_similarity(n, w, l)}$
      $cross\_lingual\_evidence.\text{append}(similarity, source\_rank)$
    end for
  end for
  return $cross\_lingual\_evidence$
end function

computation based on cosine distance between text embeddings; 2) scores based on Natural Language Inference (NLI) model.

Cosine distance Firstly, we evaluated the similarity between two news based on their texts' embeddings. As the similarity between text embeddings can be interpreted as the similarity between text content, we assumed that such a strategy for content similarity computation can correlate with the fact that one news support

information from another one. However, there can be cases when the contents of the news can be very close or even duplicated, but the special remarks such as “Fake", “Rumor", etc. indicate the refutation of the original facts. We took into account such situations. As a result, the algorithm for this approach of content similarity computation looks as follows:

1. If the link from the search leads to the file and not to the HTML page, then the news at this link is automatically considered dissimilar to the original one;

2. If there are signs of disproof of news such as the words “fake", “false", “rumor", “lie" (and their translations to the corresponding language), negations, or rebuttal, then the news is automatically considered dissimilar to the original one;

3. Finally, we calculate the similarity between the news' title and the translated original one. For a similarity measure, we choose cosine similarity between sentence embeddings. To get sentence vector representation we average sentence's tokens' embeddings extracted from Multilingual Bert (mBERT) released by [Devlin et al., 2019]. If the similarity measure overcomes the threshold θ, then the information described in scraped news and original news is considered the same.

Natural Language Inference (NLI) On the other hand, the task of estimating similarity between news contents can be reformulated as Natural Language Inference task. Natural Language Inference (NLI) is the problem of determining whether a natural language hypothesis h can reasonably be inferred from a natural language premise p [MacCartney and Manning, 2009]. The relations between hypothesis and premise can be *entailment*, *contradiction* and *neutral*. The release of the large NLI dataset [Bowman et al., 2015] and later multilingual XNLI dataset [Conneau et al., 2018] made possible the development of different deep learning system to solve this task.

The number of classes and their meaning of them in the NLI task is very similar to the labels “Support", “Refute" and “Not enough info" that are used for the

Premise p	Hypothesis h	Label
Israel invented a vaccine against coronavirus	Israel is not releasing a coronavirus vaccine – The Forward	contradiction
Israel invented a vaccine against coronavirus	Covid-19 pandemic in Israel – Wikipedia	neutral
Israel invented a vaccine against coronavirus	Israel's vaccine has 90% efficacy in trial	entailment

Table 4.2: Example how Natural Language Inference (NLI) model can be used to extract relations between news.

stance detection task in the fake news detection pipeline and that we used in the manual markup. Moreover, in [Sadeghi et al.] the usage of NLI features for stance detection task based was tested. The best model based on NLI features showed a 10% improvement in accuracy over baselines on the FNC-1 dataset. The example of the usage of the NLI model on news titles is presented in Table 4.2.

We used XLM-RoBERTa-large model pretrained on multilingual XNLI dataset[10] to obtain NLI scores for pairs "original news as premise $p \leftrightarrow$ scraped news as hypothesis h". Also, we generated input in a special format: 1) the premise was formulated as "The news "<news title + first N symbols of content>" is legit"; 2) the hypothesis was only "<news title + first N symbols of content>". The size N of the used content was a hyperparameter of this NLI-based approach for the news content similarity computation.

Additional features

Source credibility As it was discussed in Section 3.3.1, one of the aspects to which users pay attention during news verification is the credibility of the news source. In addition, such a feature about external sources was widely used in methods described in Section 3.3.3. We as well took into account the credibility of the source from which the piece of news comes. Following [Popat et al., 2016], we used AlexaRank for source assessment.

Named Entity frequency During the manual experiment, it was discovered that cross-lingual check is more relevant for news about worldwide important events, peo-

[10] https://huggingface.co/joeddav/xlm-roberta-large-xnli

ple, or organizations and not the local ones. As a result, to evaluate the worthiness of the news to be cross-lingual checked we: 1) extracted Named Entity (NE) from the title and the content of news; 2) found the most relevant page on Wikipedia; 3) evaluate AlexaRank of corresponding Wikipedia page to estimate the popularity of the NE.

4.3.2 Comparison with Manual Markup

To understand the validity of chosen approaches for content similarity computation between news, we conducted a small case study on a manually marked-up dataset. For each approach of news similarity estimation, we calculated the accuracy of such an experimental setup: the classification task if the scraped news supports the original news. So, from manually marked-up data we got a dataset of labeled 1000 pairs “original news ↔ scraped news". For each pair, we transferred from a three-person annotation to a single label by the voting of the majority.

Taking such a setup, we fined-tuned hyperparameters for both approaches. We fine-tuned threshold θ for the embeddings-based similarity. We conducted hyper-parameter search on the segment $[0.1, 0.9]$ with a step $\delta = 0.1$. The best result was achieved with the $\theta = 0.5$ threshold for decision making if the scraped news supports or not the original news. For NLI based approach, we fine-tuned the length of the text passed as the input to the NLI model. We got the best hyperparameters setup for the NLI approach is 500 symbols length of news text which is equal to the title of the news with the first two paragraphs of the content. For the NLI model, we united “neutral" and “contradiction" classes to have a similar setup as for the embeddings-based approach.

Finally, for *cosine distance* approach we achieved 82% accuracy, while for *NLI* approach 70% accuracy on 1000 pairs dataset. Although the models are not ideal, we believe that they can be used as baseline approximations of human judgments.

4.3.3 Automatic Fake News Detection

Finally, we conducted a set of experiments to validate Hypothesis 3: if the presented cross-lingual evidence feature can improve automatic fake news detection systems. We integrated the automated cross-lingual evidence feature into the fake news classification pipeline tested on three datasets.

Datasets

In tested datasets for our automated experiment, we tried to cover several world-wide spread topics – politics, famous people and events, entertainment as well as the most recent event connected with COVID-19.. Firstly, we evaluate the systems on a multi-domain dataset by [Pérez-Rosas et al., 2018] which consist of two parts: *FakeNewsAMT* dataset (240 fake and 240 legit articles) and *CelebrityDataset* dataset (250 fake and 250 legit articles). *FakeNewsAMT* dataset consists of news from six topics: sports, business, entertainment, politics, technology, and education. *CelebrityDataset* is dedicated to rumors, hoaxes, and fake reports about famous actors, singers, socialites, and politicians. Secondly, we ran experiments on COVID-19 fake news dataset *ReCOVery* [Zhou et al., 2020b]. It consists of 2029 (665 fake and 1364 true news). All datasets are originally in English.

Dataset	# Fakes	# Legit	Covered topics
FakeNewsAMT	240	240	sports, business, entertainment, politics, technology, and education
CelebrityDataset	250	250	rumors, hoaxes, and fake reports about famous actors, singers, socialites, and politicians
ReCOVery	665	1364	rumors, hoaxes, and fake news about COVID-19

Table 4.3: Statics of datasets that were used to test fake news classification with proposed cross-lingual evidence feature.

We used 70%-20%-10% proportion for train-test-dev validation split.

Baselines

We compared our approach with several baselines. For the baseline, we chose the fake news systems based on internal features computed either via linguistic analysis or neural networks.

Linguistic Features: In [Pérez-Rosas et al., 2018] a baseline fake news classification model was trained based on Ngrams, punctuation, psycholinguistic features extracted with LIWC, readability, and syntax. In [Zhou et al., 2020b] LIWC features were also used as one of the proposed baselines. We tested these features separately, grouped them all, and in combination with our proposed feature. We experimented with SVM, RandomForest, LogRegression, and LightGBM. We used standard hyperparameters set for the models. The results of the best models based on LightGBM are presented. We call the model based on the concatenation of all listed above linguistic features as **All linguistic**.

Text-CNN, LSTM: Following [Zhou et al., 2020b], we tested classical model for text categorization TextCNN and LSTM on all datasets.

BERT, RoBERTa: BERT [Devlin et al., 2019] based models were used for fake news detection by [Kaliyar et al., 2021] and specifically for COVID-19 fake news classification [Gundapu and Mamid, 2021, Glazkova et al., 2020]. We used pretrained models – bert-base-uncased[11] and roberta-base[12] – and fine-tuned them.

Only monolingual evidence (ME): In addition, we compared our feature with the case when only monolingual English evidence was used. For this baseline, the LightGBM model was used as well.

Results

To evaluate the performance of fake news classification models, we use three standard metrics: F_1, *precision*, *recall*. The formulas are provided bellow.

$$precision = \frac{TP}{TP + FP}, \quad recall = \frac{TP}{TP + FN} \quad F_1 = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{4.1}$$

We experimented with both types of content similarity measurements – either cosine similarity between embeddings (Emb.) or NLI scores – concatenated with the source credibility rank (Rank) of the scraped news. Both Emb. and NLI features were presented as a vector of similarity scores for the pairs "original news ↔ scraped news".

[11] https://huggingface.co/bert-base-uncased
[12] https://huggingface.co/roberta-base

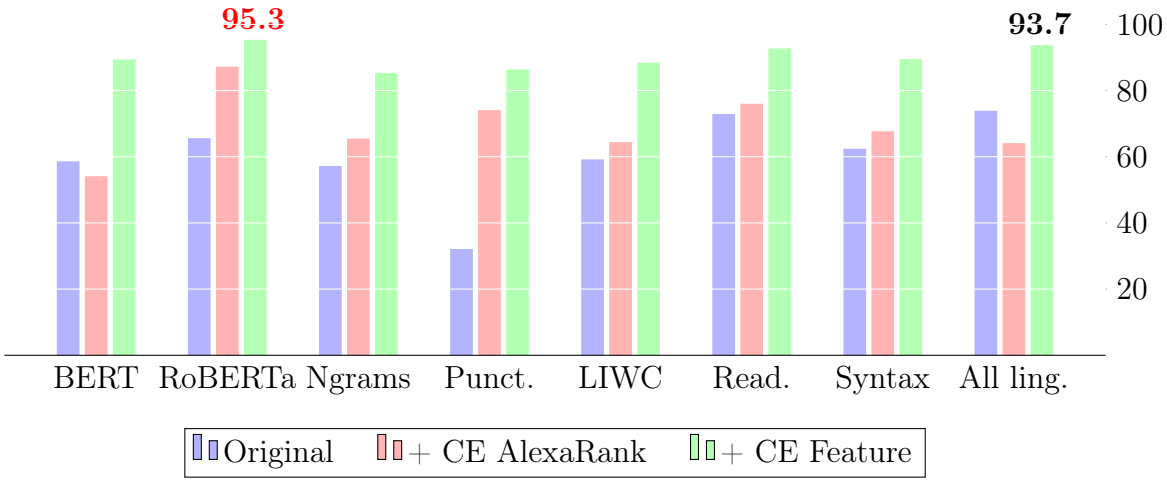


Figure 4-4: Results on FakeNewsAMT dataset (F_1 score): adding proposed Cross-lingual Evidence (CE) improves various baseline systems and yields state-of-the-art results with RoBERTa model.

Table 4.5 compares the results of our model based on cross-lingual evidence (CE) with the baselines on three datasets. To prove the statistical significance of the result we used paired t-test on 5-fold cross-validation. All improvements presented in the results are statistically important. Additionally, we provide histogram view of F_1 scores comparison for all three datasets: FakeNewsAMT (Figure 4-4), Celebrity (Figure 4-5), and ReCOVery (Figure 4-6).

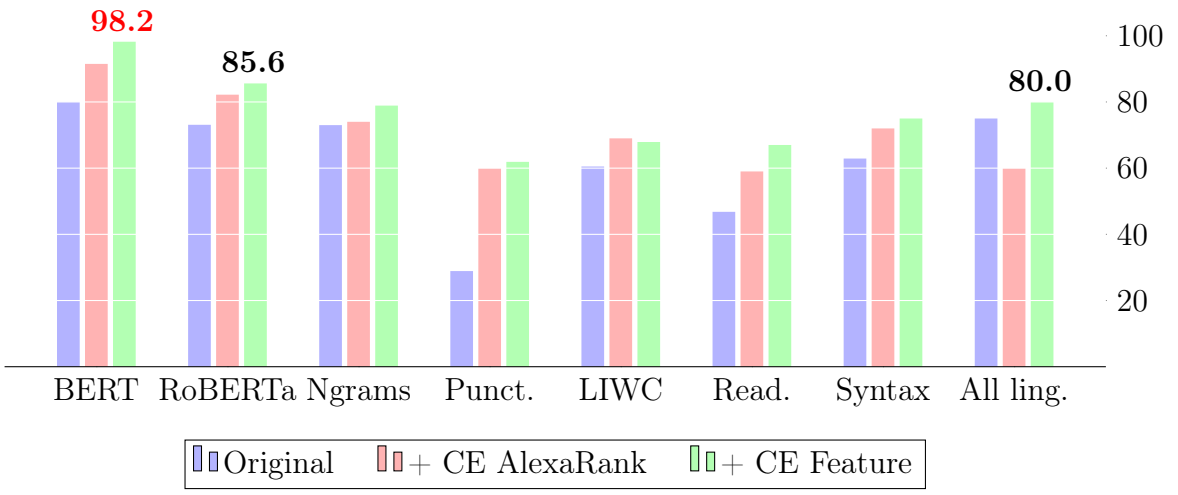


Figure 4-5: Results on Celebrity dataset (F_1 score): adding our Cross-lingual Evidence (CE) improves various baseline systems and yields state-of-the-art result with BERT model.

CE features along slightly outperform the baselines or show almost the same

results as linguistic features. As it was expected, only ME based fake news detection system shows worse results than the usage of CE features. NLI based CE features show generally worse results than embeddings based approach. For further improvements, the NLI model can be additionally trained specifically for the task of detection of confirmation or refutation specifically in news content.

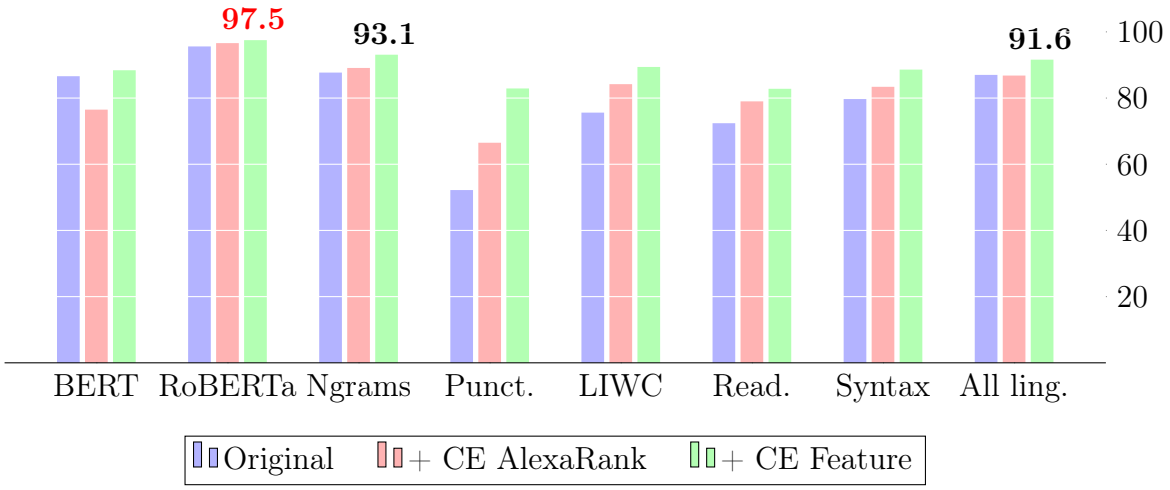


Figure 4-6: Results on ReCOVert dataset (F_1 score): adding our Cross-lingual Evidence (CE) improves various baseline systems and yields state-of-the-art result with RoBERTa model.

The addition of the CE feature improves all baseline models. For *FakeNewsAMT*, the best $F_1 = 0.973$ score is achieved with BERT embeddings in combination with CE features. For *Celebrity* dataset, BERT again with CE features shows the best results achieving the best $F_1 = 982$ result. In spite RoBERTa showing the highest $F_1 = 0.975$ score for *ReCOVery*, the combination of all linguistic and CE features and specifically Ngrams with CE features show competitive results achieving $F_1 = 0.916$ and $F_1 = 0.931$ respectively.

The importance of the proposed features in the model's decision-making is also confirmed by the feature's importance. The top-30 features' importance for best models for all datasets based on embeddings similarities is reported in Appendix A.1. For all *FakeNewsAMT*, *Celebrity*, and *ReCOVery* dataset we can see the presence not only English, but indeed cross-lingual evidence features in the top important features. Although English evidence features for the top-3 news from the search results got the highest importance, the similarity scores and rank of the source from

other languages (French, German, Spanish, Russian) contribute as well.

Title	English translation
Original news (FAKE)	
Kate Middleton & Prince William Try To Save Crumbling Marriage?	–
English search results	
Prince William and Kate Middleton's Love Through the Years	–
French search results	
Le jour où le prince William a demandé Kate Middleton en mariage	The day Prince William proposed to Kate Middleton
German search results	
Elternschaft, Babynamen, Prominente und königliche Nachrichten | CafeMom.com	Parenting, Baby Names, Celebrities, and Royal News | CafeMom.com
Spanish search results	
Príncipe William – Clarín.com	Prince William - Clarín.com
Russian search results	
Факты о свадьбе Кейт Миддлтон и принца Уильяма, о которых вы могли не знать	Kate Middleton and Prince William's wedding facts you might not know
Title	**English translation**
Original news (LEGIT)	
Amazon Prime Air drone completes its first US public delivery	–
English search results	
Amazon Prime Air drone completes its first US public delivery	–
French search results	
E-commerce. Amazon autorisé à livrer par drone aux États-Unis	E-commerce. Amazon authorized to deliver by drone to the United States
German search results	
Prime Air: FAA erteilt Amazons Liefer-drohnen die Starterlaubnis	Prime Air: FAA gives Amazon's delivery drones permission to take off
Spanish search results	
Amazon hace su primera entrega por dron en Estados Unidos	Amazon makes its first delivery by drone in the United States
Russian search results	
Amazon запускает дроны Prime Air для быстрой доставки	Amazon launches Prime Air drones for fast delivery

Table 4.4: The example of output that can be produced by Multiverse.

Additionally, we explored if the cross-lingual feature can add explainability to the fake news classification system. Thus, the user can enter the headline of news and get not only the class probability as an answer from the fake news detection system, but also the list of multilingual news that was used for feature calculation, their similarity to original news, and source credibility. An example of such output is provided in Table 4.4. The extended version of it with scores and the case for legit news can be found in Appendix A.2. Such output can allow the user to have a look

at the situation from different perspectives and critically relate to the information stated in the news.

	FakeNewsAMT			**Celebrity**			**ReCOVery**		
	Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1
TextCNN	0.276	0.250	0.260	0.641	0.703	0.664	0.733	0.913	0.805
LSTM	0.614	0.614	0.614	0.745	0.740	0.740	0.800	0.803	0.793
ME Emb. + Rank	0.539	0.593	0.592	0.552	0.550	0.550	0.794	0.798	0.793
ME NLI + Rank	0.637	0.633	0.634	0.554	0.550	0.550	0.756	0.761	0.752
CE Emb. + Rank	0.872	0.864	0.864	0.631	0.620	0.619	0.829	0.829	0.829
CE NLI + Rank	0.837	0.833	0.834	0.625	0.620	0.620	0.767	0.771	0.762
BERT	0.586	0.586	0.586	0.800	0.800	0.800	0.868	0.868	0.866
BERT + CE Emb + Rank	**0.884**	**0.885**	**0.894**	**0.982**	**0.982**	**0.982**	**0.870**	**0.863**	**0.884**
RoBERTa	0.895	0.548	0.656	0.856	0.690	0.731	0.986	0.936	0.956
RoBERTa + CE Emb + Rank	**0.973**	**0.938**	**0.953**	**0.952**	0.784	**0.856**	**0.992**	0.960	**0.975**
Ngrams	0.573	0.572	0.572	0.730	0.730	0.730	0.878	0.879	0.877
Ngrams + CE Emb. + Rank	**0.864**	**0.854**	**0.853**	**0.789**	**0.790**	**0.789**	**0.931**	**0.932**	**0.931**
Ngrams + CE NLI + Rank	0.844	0.844	0.844	0.690	0.690	0.690	0.862	0.860	0.856
Punctuation	0.239	0.489	0.321	0.211	0.460	0.289	0.433	0.658	0.522
Punctuation + CE Emb. + Rank	**0.872**	0.864	0.864	0.631	0.620	0.619	**0.829**	**0.829**	**0.829**
Punctuation + CE NLI + Rank	0.870	**0.865**	**0.865**	**0.690**	**0.690**	**0.690**	0.767	0.771	0.762
LIWC	0.597	0.593	0.592	0.630	0.610	0.605	0.768	0.771	0.756
LIWC + CE Emb. + Rank	**0.894**	**0.885**	**0.884**	**0.692**	**0.680**	**0.679**	**0.894**	**0.894**	**0.894**
LIWC + CE NLI + Rank	0.850	0.844	0.844	0.650	0.650	0.650	0.816	0.815	0.808
Readability	0.729	0.729	0.729	0.478	0.470	0.468	0.732	0.741	0.724
Readability + CE Emb.+ Rank	**0.928**	**0.927**	**0.927**	**0.674**	**0.670**	**0.670**	**0.828**	**0.829**	**0.828**
Readability + CE NLI + Rank	0.854	0.854	0.854	0.601	0.600	0.599	0.772	0.773	0.762
Syntax	0.626	0.625	0.624	0.639	0.630	0.629	0.812	0.809	0.797
Syntax + CE Emb. + Rank	**0.902**	**0.895**	**0.895**	**0.754**	**0.750**	**0.750**	**0.886**	**0.886**	**0.886**
Syntax + CE NLI + Rank	0.505	0.500	0.501	0.525	0.520	0.519	0.840	0.837	0.832
All linguistic	0.739	0.739	0.739	0.750	0.750	0.750	0.875	0.874	0.870
All linguistic + CE Emb. + Rank	**0.940**	**0.937**	**0.937**	**0.801**	**0.800**	**0.800**	**0.916**	**0.917**	**0.916**
All linguistic + CE NLI + Rank	0.886	0.885	0.886	0.737	0.732	0.732	0.864	0.865	0.862

Table 4.5: Results of integration of cross-lingual evidence (CE) feature into automated fake news classification systems. The proposed feature is used in two way based on content similarity computation strategy: (i) based on text embeddings (Emb.) (ii) based on NLI scores (NLI). It is also combined with the rank of the news articles source (Rank). The CE feature alongside showed worse results then baseline methods. All the improvements of the results were statistically proven by t-test on 5-fold cross-validation. However, in combination with linguistic features the SOTA results are achieved.

4.4 Summary

We presented Multiverse: an approach for fake news detection based on cross-lingual evidence (CE) from the Web search that is motivated by user behavior and overcomes the limitations of external monolingual features of previous work.

Firstly, we conducted a manual study on 20 news datasets to test the hypothesis of whether the real-life user can use cross-lingual evidence to detect fake news. The annotators successfully passed the task of such news verification providing also the markup of 100 pairs "original news ↔ scraped news".

After the first hypothesis confirmation, we tested our approach for the automated detection of fake news. We experimented with two strategies for content similarity estimation: (i) based on cosine distance between news texts embeddings; (ii) based on Natural Language Inference (NLI) scores where the original news used as premise p and the scraped news as hypothesis h. We compared the proposed strategies with human assessments of 1000 pairs of marked news showing that these methods can be used for news similarity estimation. Finally, we integrated the proposed cross-lingual feature into an automated fake news detection pipeline. To this point, the cross-lingual feature itself showed the performance only at the baseline level. However, in combination with linguistic features based on original text of the news, it outperformed both statistical and deep learning fake news classification systems. Furthermore, we provided an ablation study in which the necessity of using cross-lingual evidence with source rank was proven compared to only monolingual features.

The proposed cross-lingual evidence feature can have several limitations. Firstly, the usage of Google services for search and translation steps can bring bias to the personalized system. We tried to avoid personalization in search by using incognito mode during experiments to hide search history and location parameters. Nevertheless, Google search can use meta information and adjust the resulting feed. On the one hand, the usage of Google services is motivated by user search experience. On the other hand, the reproduction of such experiments can be quite difficult. In our future work, we plan to overcome such an issue in the experiments by using already pre-saved snapshots of searches on the Internet for an exact period of time. Also,

it should be taken into account that the proposed cross-lingual signal will be useful for identifying fake news not immediately after the news appears, but with a slight delay. Naturally, journalists from different countries need time to react to the news.

As we used automated translation to get the queries for cross-lingual search, there can be another side of this automated translation application – some Internet editions can use automated translation to get the duplication of the news in the target language. Moreover, the method of machine translation is becoming more and more advanced each year. As a result, we can get the repetition of the news in search results in different languages. However, we believe that our proposed pipeline can handle such cases as we incorporated in our feature the source rank of the news. But in future work, the addition of detection of machine-generated texts can be considered.

One of the future extensions of the proposed research can be the cross-lingual check of the news not only via Web search but additionally with the information from comments section in social media. User-generated texts can bring a strong signal to news verification. We can group news posts on social media based on their cross-lingual similarity and compare the comments left by users.

5

Multilingual Text News Similarity Metrics

In the previous Section 4, we introduced a new feature for fake news detection based on cross-lingual news evidence. One of the parts of this feature is the similarity measure between news in different languages. In previous experiments, we explored only two types of metrics for such cross-lingual news similarity measurement. In this chapter, we want to extend our research and explore new metrics for multilingual and cross-lingual news similarities. The research is based on the SemEval-2022 competition *"Multilingual News Article Similarity"* [Chen et al., 2022]. The contributions of this chapter are the follows:

1. We explore new multilingual and cross-lingual news similarity measures based on several ideas: Transformer-based embeddings, addressing this task as NLI task, and extracting additional signals as Named Entity (NE);

2. We incorporate new metrics in the already proposed fake news detection pipeline.

3. We provide a demonstration of how such cross-lingual similarity measurements can be shown to a user for news credibility evaluation.

The code of the proposed similarity models is available online.[1]

[1] https://github.com/s-nlp/multilingual_news_similarity

5.1 Problem Statement

The aim of the SemEval 2022 Task 8 competition [Chen et al., 2022] is to develop systems that identify multilingual news articles that provide similar information. This is a document-level similarity task in the applied domain of news articles, rating them pairwise on a 4-point scale from most to least similar. The example of markup is presented in Figure 5-1.

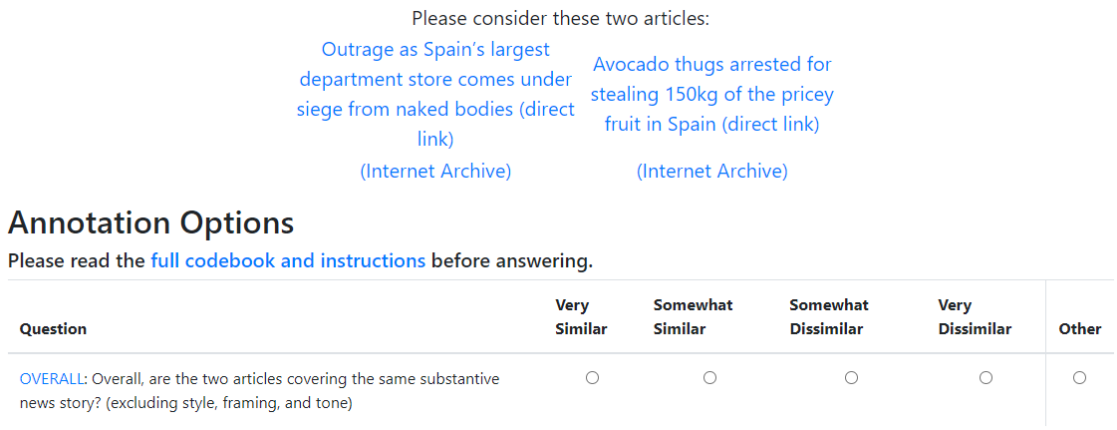


Figure 5-1: Example of data markup for the SemEval-2022 competition *"Multilingual News Article Similarity"* [Chen et al., 2022].

The dataset consists of 4 818 news pairs for training and 4 956 pairs for evaluating the results. The news pairs can be written in the same language as well as in different languages. In addition, news on 3 of 10 languages are not provided in the train part and test data has 19% more cross-lingual pairs. The quantitative statistics of the dataset parts are listed in Table 5.1.

To evaluate the performance of the approaches under consideration, Pearson Correlation was used. This metric is for two vectors representing ground true and predicted similarity scores. It can be calculated with the following formula:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{5.1}$$

where in our case we consider predicted similarity score x are the results from the proposed models for similarity measurement and ground true scores y as scores provided from manual annotation.

language pairs	train	eval	Mean Distance Score
ar-ar	274	298	2.41
de-de	857	608	2.57
de-en	531	185	3.18
de-fr	-	116	1.88
de-pl	-	35	1.69
en-en	1800	236	2.86
es-es	570	243	2.34
es-en	-	496	2.79
es-it	-	320	2.29
fr-fr	72	111	2.39
fr-pl	-	11	2.00
it-it	-	411	2.65
pl-pl	349	224	2.35
pl-en	-	64	2.35
ru-ru	-	287	2.78
tr-tr	465	275	2.74
zh-zh	-	769	2.22
zh-en	-	213	3.07
Totals	4918	4902	2.62

Table 5.1: Quantitative statistics of Training and Evaluation parts of the dataset used for a research in this chapter.

5.2 Baselines

Baseline approaches are built upon token-based similarity measures. The most simple metric of this type is **Word Count**. It is just a difference in the number of tokens in the first and second texts. It can be calculated with the formula:

$$WC = \frac{|n_1 - n_2|}{max(n_1, n_2)} \tag{5.2}$$

Another measure that is commonly used is **Jaccard Similarity**. This measure is the intersection of tokens sets divided by the union of these sets:

$$JS(\mathbf{A}, \mathbf{B}) = \frac{|\mathbf{A} \cap \mathbf{B}|}{|\mathbf{A} \cup \mathbf{B}|}, \quad 0 \leq JS(\mathbf{A}, \mathbf{B}) \leq 1 \tag{5.3}$$

Baseline approaches are the composition of simple features extracted from the text and one of the four ML classifiers: Logistic Regression, SVC with linear kernel, Random Forest, and XGBoost [Chen and Guestrin, 2016]. There are three sets of

features based on the statistical texts similarity metrics:

1. **Set A**. The first set has only one feature — Jaccard similarity of named entities extracted from both news texts which need to be compared.

2. **Set B**. The second set is set A with the addition of text Jaccard Similarity.

3. **Set C**. The third set is set B with the addition of word count difference.

Additional attention should be paid to the way how the cross-lingual pairs are treated since Jaccard Similarity for texts in different languages is often equal to zero. As a consequence named entities and also words are linked with the help of WikiData [Vrandečić and Krötzsch, 2014]. The idea is based on the fact that Wikipedia articles in different languages dedicated to the same entity have the same identification number. In addition, such an approach allows filtering of incorrectly extracted named entities, because they won't be found in the Wikipedia database.

5.3 Transformer-based Pre-trained Encoders

Pre-trained neural masked language models like BERT [Devlin et al., 2019] and RoBERTa [Liu et al., 2019a] have shown superior performance on a wide range of NLP tasks both in monolingual and multilingual settings. During the work on the task of the competition, the approach for fine-tuning Transformers was developed. The following multilingual models were tested: DistilBERT[2], BERT[3] RoBERTa[4], XLM[5]. All these models support all the languages included in the competition dataset. Two different architectures were chosen for fine-tuning the language models. The first one is based on the approach for the BERT Next Sentence Prediction problem described in the original article [Devlin et al., 2019]. We will call it **TransformerEncoderCLS**. The second approach is inspired by the articles [Reimers and

[2]https://huggingface.co/distilbert-base-multilingual-cased
[3]https://huggingface.co/bert-base-multilingual-cased and https://huggingface.co/bert-base-multilingual-uncased
[4]https://huggingface.co/xlm-roberta-base and https://huggingface.co/xlm-roberta-large
[5]https://huggingface.co/xlm-mlm-17-1280

Gurevych, 2019, Sergei et al., 2021]. It will be labeled as **TransformerEncoder-CosSim** from now on.

5.3.1 TransformerEncoderCLS

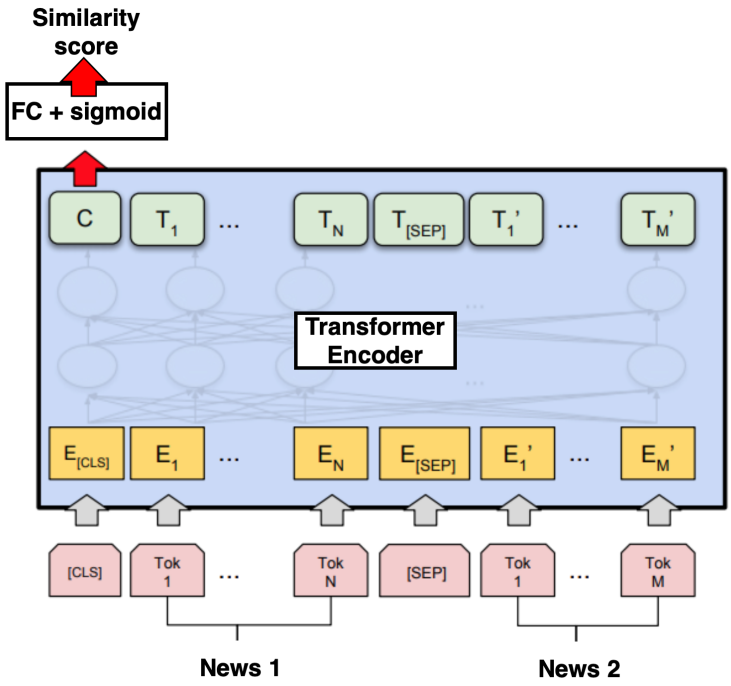


Figure 5-2: TransformersEncoderCLS architecture, depicted from the original paper [Devlin et al., 2019].

The general scheme of the approach is shown in Figure 5-2. The Transformer model takes as input two tokenized news texts separated by [SEP] token, which is needed for the model to distinguish words from different texts. Also, this sequence of tokens has a special [CLS] token in the beginning. Passing through the layers of the model, each token results in the embedding vector. All the information from the sequence is aggregated in the [CLS] token embedding. That is why we use it as the input to the regression head, which is the combination of a fully-connected layer and Sigmoid nonlinearity. The linear layer dimensions are $emb\_len \times 2$, where emb_len is the dimension of the hidden layer. We use the output probability of the first class as the similarity score. Together with mapped to $[0, 1]$ range ground true similarity scores, the predicted scores are passed to the MSE loss function. Transformer's

weights are not frozen while training and initialized from the aforementioned pre-trained multilingual models.

5.3.2 TransformerEncoderCosSim

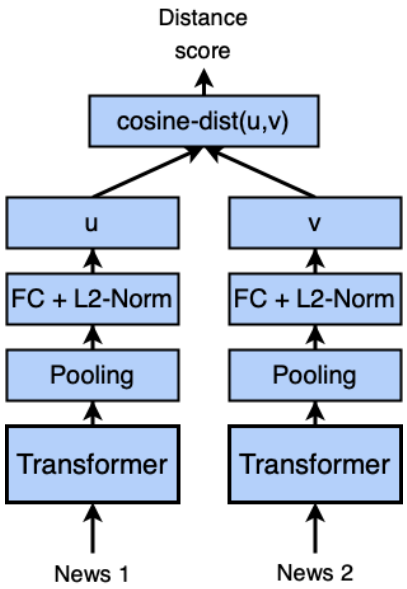


Figure 5-3: TransformerEncoderCosSim architecture.

The general scheme of the approach is shown in Figure 5-3. The pre-trained Transformer model takes as input the tokenized news text. Then, Transformer output embeddings are passed through the average pooling followed by a fully-connected layer and L2 normalization layer. This procedure is applied for both compared news. Then, the resulting text embeddings are passed in the cosine distance function which is computed with the equation below to produce a distance score:

$$cosine\_dist = 1 - |cosine\_sim| \tag{5.4}$$

We use the absolute value of the cosine similarity function because it takes values from -1 to 1. Together with mapped to $[0, 1]$ range ground true scores, the predicted scores are passed to the MSE loss function. The Transformer's weights are not frozen while training and initialized from the aforementioned pre-trained multilingual models.

5.4 Natural Language Inference

The same as in Chapter 4 (Section 4.3.1), we address the task of multilingual news similarity as NLI task. We again use XLM-RoBERTa model pre-trained on multilingual XNLI dataset[6] to obtain NLI scores. NLI model outputs the probabilities of news pair to be classified as entailment, contradiction, or neutral. Hence, it's 3 real numbers from the $[0, 1]$ range. These extracted NLI features are passed as input to the Machine Learning (ML) model, which predicts the similarity score for the pair of news under consideration. In our work, we compared the performance of several regression models: Linear Regression, Support Vector Machine for regression, Decision Trees, Random Forest, and Gradient Boosting. The last one gave the best results. The general scheme of the approach is shown in Figure 5-4. Also, several improvements to this pipeline are applied:

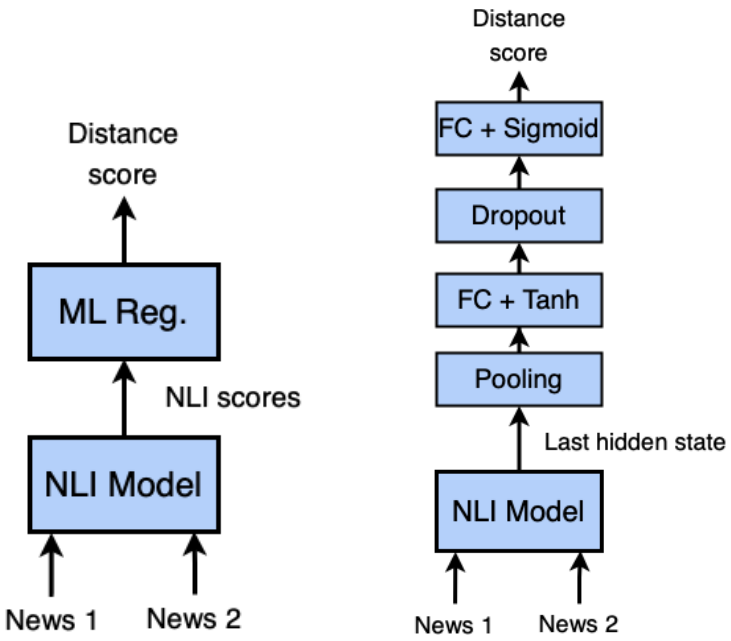
1. **Both pairs**. Each piece of news is used as a premise and hypothesis. As a result, we get twice more features for training.

2. **Subject-Verb-Object triplets**. We extract syntactic dependencies from the sentences of a text to make triplets consisting of subjects, verbs, and objects. These triplets are passed to the model. Such an approach shortens the input data, which makes the process of extracting NLI features faster and doesn't have a significant influence on the quality of the method. To extract syntactic dependencies Spacy library is used [Honnibal et al., 2020].

3. **Fine-tune**. We fine-tune the NLI model on the data of the competition. The approach is based on the one proposed by [Martín et al., 2021]. We add the regression head to the NLI model, which has global average pooling of the last hidden state of the transformer model, a linear layer with 768 neurons and tanh activation, a 10% dropout for training, and a classifier linear layer with sigmoid. The output probability is treated as a similarity score, and MSE loss is used. This regression head is trained, freezing the XLM-RoBERTa-large weights to preserve the previous pre-training. This is optimized using Adam

[6] https://huggingface.co/joeddav/xlm-roberta-large-xnli

optimizer [Kingma and Ba, 2015] with 10^{-3} learning rate. The general scheme of the approach is shown in Figure 5-4.



(**a**) Basic architecture. (**b**) Fine-tuning architecture.

Figure 5-4: The schema of NLI approach with two settings.

5.5 Named Entity Recognition

Transformers have great performance but almost no interpretability. In search of interpretability, the Named Entity Recognition (NER) based approach has been developed. The general scheme of the approach is shown in Figure 5-5. News texts are pre-processed and forwarded to the NER extractor to extract locations (`LOC`), organizations (`ORG`), and person entities (`PER`). For this task we've tested and compared several tools:

1. **Transformer for named entities tagging.** We used BERT[7] pre-trained model. It is a Named Entity Recognition model for 10 high-resource languages (Arabic, German, English, Spanish, French, Italian, Latvian, Dutch, Portuguese, and Chinese) based on a fine-tuned mBERT base model.

[7] https://huggingface.co/Davlan/bert-base-multilingual-cased-ner-hrl

2. **Polyglot for Named Entity Extraction.** The models from this package [Al-Rfou et al., 2015] were trained on datasets extracted automatically from Wikipedia. Polyglot currently supports 40 major languages, including all presented in the dataset of the competition.

3. **Spacy.** Spacy library [Honnibal et al., 2020] provides huge variety of NLP tools, including NER extractor. We used multi-language model,[8] trained on Wikipedia.

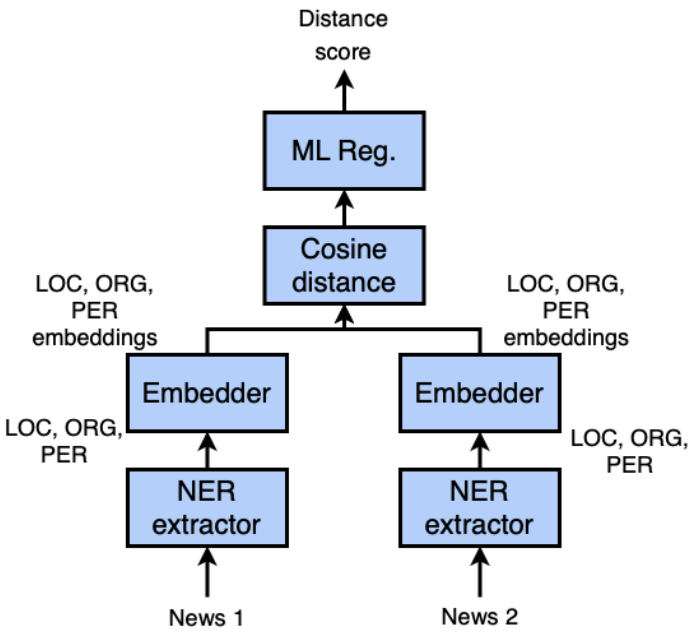


Figure 5-5: The schema of NER approach.

In the next step, we vectorize extracted entities with Bag of Words, Tf-Idf, fastText [Bojanowski et al., 2017], and BERT embeddings[9] for comparison. Then we average all the word vectors. As a result, we obtain 3 vectors (one for each of `LOC`, `PER`, `ORG` entities) for each text. Corresponding vectors for `LOC`, `ORG`, `PER` for two texts are compared with cosine distance to get 3 distance scores for every pair of news under consideration. Then, these scores are passed in the Machine Learning model to get the final distance score. We test several regression models: Linear Regression, Support Vector Machine for regression, Decision Trees, Random Forest, and Gradient Boosting.

[8] xx_ent_wiki_sm
[9] bert-base-multilingual-uncased pre-trained model was used

5.6 Additional study

To improve the quality of the prediction the following two techniques were tested:

1. **Augmentation.** Testing part of the dataset has a lot of language pairs that are not presented in the training part of the dataset. To test the influence of unseen language pairs on the results, we added pairs of news for the missing language pairs. Such augmentation was performed with the help of the Google Translator, which was accessed with the help of the Deep Translator python library. The pairs of news were selected randomly from the pairs written in English and then translated to the target languages. Samples were added to the training part of the dataset in the same proportion they are presented in the testing part of the dataset. As a result, the training dataset was extended to 7505 samples.

2. **Stacking.** Ensembling different models is a common way to improve the scores. To aggregate the dependencies caught by several models, we exploited the technique called stacking. To form the ensemble, we used TransformerEncoderCosSim, TransformerEncoderCLS, fine-tuned NLI model, and NER model[10] which has shown the best results in the experiments described below. All the models were trained on three-quarters of the training dataset. And one-quarter of the dataset was used to train the aggregation model. We used the Linear Regression model with L_2 regularization as an aggregation model.

5.7 Results

Final results for all separate methods are provided in Table 5.2. Also, we provide the results for ensembles of models in Table 5.8. The application of ensembling and augmentation techniques improved the best result to a $0,763$ correlation. In addition to the test set, the performance of the developed systems was evaluated

[10]We used the following combination: BERT NER tagger, BERT embeddings, Gradient Boosting ML model.

on the validation set. The validation set was randomly sampled from the training data in the case of TransformerEncoder methods, including fine-tuned NLI model. For other methods, the results on validation are the results obtained with 5-fold cross-validation.

	Validation	Evaluation
TransformerEncoderCLS	**0.813**	0.706
TransformerEncoderCosSim	0.793	**0.734**
NLI	0.478	0.477
NLI fine-tuned	0.670	0.632
NER	0.496	0.395
NLI + NER	0.615	0.546

Table 5.2: The comparison of proposed approached for both validation and evaluation sets by Pearson correlation with manual annotations.

Transformer models. As it has already been said, the TransformerEncoderCosSim model has shown the best result. It was the one with XLM[11] pre-trained model. The worst score was given by the DistilBert model. We provide the comparison of different encoders from Transformers for two proposed models in Table 5.3. As for the TransformerEncoderCLS model, its performance has dropped by 12% on the evaluation part of the dataset in comparison to the validation part. And it's become worse than the TransformerEncoderCosSim model, although it showed better results on the cross-validation.[12] In general, the transformer-based models have a lower correlation on the evaluation data. You can see a similar behavior for the NLI fine-tuning approach.

NLI. Firstly, we provide the results for different regression models for NLI pairs ↔ titles in Table 5.4. The best score for the NLI approach was given by the Gradient Boosting model.

[11] https://huggingface.co/xlm-mlm-17-1280

[12] Model which has shown the best result: https://huggingface.co/xlm-roberta-large

	Transformer-EncoderCLS	Transformer-EncoderCosSim
distilbert	0.591	0.679
bert-base-cased	0.644	0.704
bert-base-uncased	0.678	0.714
xlm-roberta-base	0.656	0.643
xlm-roberta-large	**0.706**	0.718
xlm-mlm-17-1280	0.650	**0.734**

Table 5.3: Comparison of performance of different pre-trained encoders from Transformers on evaluation dataset.

	Validation	Evaluation
LinearRegression	0.290	0.364
SVR	0.288	0.356
DecisionTreeRegressor	0.228	0.273
RandomForestRegressor	0.477	0.469
GradientBoostingRegressor	**0.483**	**0.480**

Table 5.4: Comparison of the performance of different ML models for NLI pairs ↔ titles approach.

The comparison of the results of the best NLI-based model with different setups is provided in Table 5.5. We experimented with classical ML models to gain not only good performance score but also explainability of the model's decision. Also, such models are fairly lightweight.

The fine-tuning approach has given the best correlation here. Also, there is a tendency for smaller input text to have better scores. The highest correlation was achieved when only titles were given as input. The reason for that could be that the NLI model was trained on the XNLI dataset, composed of short phrases. That is why it was decided to try to shorten the news with the extraction of SVO triplets from them. The extracted triplets were joined to form a text which was forwarded to the input of the NLI model.

	Validation	Evaluation
NLI tiltes	0.453	0.438
NLI pairs - titles	0.478	0.477
NLI pairs - titles + text	0.354	0.310
NLI pairs - SVO	0.154	0.107
NLI fine-tuned - titles	**0.670**	**0.632**
NLI fine-tuned - titles + text	0.627	0.589
NLI fine-tuned - SVO	0.495	0.422

Table 5.5: Comparison of different setups of NLI approach.

As you can see from Table 5.5 the quality of both methods (with fine-tuning and without) has dropped significantly. Hence, the conclusion is that despite SVO triplets giving a good summary of the given text, they are not applicable, at least without any complex processing, for the task of comparing the news. Also, it could mean that the source of similarity of articles is not contained in Subjects, Verbs, and Objects. Last, it is worth mentioning that the resulting summary for big texts still has quite a large size in comparison to titles.

The idea to extract NLI scores from both pairs, as was described in devoted Section 5.4, gave an improvement. Also, it can be noticed that the NLI approach without fine-tuning is quite robust to adding new languages. The score for "NLI pairs - titles" has only a slight decrease on the evaluation dataset. Although the correlation for single NLI features is low, it becomes significantly better in combination with features with the NER-based method.

NER. We present a comprehensive comparison of different NER taggers, various embedding techniques, and different Machine Learning models for the prediction of distance scores in Table 5.6.

You can see that the best correlation was shown by combination: BERT-based NER tagger, BERT embeddings, and Gradient Boosting ML model. In general, Gradient Boosting has shown superior scores for all combinations of NER taggers and embedders. Also, BERT embeddings in combination with this model have

Tagger	Embedding	Linear Regression	SVR	Decision Tree	Random Forest	Gradient Boosting
BERT-based	BOW	0.202	0.200	0.154	0.244	0.246
	Tf-Idf	0.195	0.191	0.135	0.229	0.239
	Fasttext	0.194	0.194	0.157	0.320	0.326
	BERT	0.250	0.250	0.200	0.385	**0.395**
Polyglot	BOW	0.228	0.227	0.146	0.240	0.244
	Tf-Idf	0.220	0.218	0.143	0.227	0.226
	Fasttext	0.206	0.205	0.151	0.309	0.310
	BERT	0.211	0.211	0.180	0.334	**0.342**
Spacy	BOW	0.227	0.227	0.147	0.230	0.235
	Tf-Idf	0.223	0.223	0.154	0.224	0.231
	Fasttext	0.184	0.183	0.146	0.254	0.259
	BERT	0.219	0.220	0.152	0.278	**0.279**

Table 5.6: Comparison of different NER taggers, embeddings and ML models on evaluation dataset.

shown the highest results for all embedding methods listed in the Methodology section. However, in comparison to NLI and Transformers approaches, the results for NER models are significantly lower.

We present an example of NER-based approach performance in Table A.3 in Appendix A.3. The following behaviors can be noticed. In our method in cases when no named entities were found for the `PER`, `ORG` or `LOC` classes, the distance score was set to 0.5, because it is not clear whether the absence of named entities is an indicator of similarity or not. These 0.5 scores confuse the model, increasing its generalization error. The second problem is that when there is no overlap of named entities in one of the classes, it could lead to two bad outcomes. When the other two distance scores correctly reflect the ground true similarity, like in the second example in Table A.3, the one with no overlap could be large, which spoils the overall prediction.

The second behavior happens when the extracted entities have no straight overlap but happen to be similar in vector space. For example, two different news about

	Not Augmented	Augmented
TransformerEncoderCLS	0.706 ± 0.032	0.712 ± 0.051
TransformerEncoderCosSim	$\mathbf{0.734 \pm 0.001}$	$\mathbf{0.746 \pm 0.002}$
NLI fine-tuned - titles	0.630 ± 0.003	0.637 ± 0.002
NER Hug.-Hug.-GB	0.395 ± 0.000	0.397 ± 0.000

Table 5.7: Influence of augmentation technique on the results in evaluation dataset. Pearson correlation with 0.95% confidence intervals. The pre-trained models for TransformerEncoder approaches are xlm-roberta-large and xlm-mlm-17-1280 respectively.

the close locations. In this case, the model can output a small distance, which is not correct. Also, the errors of the NER tagger make the model performance worse. As a result, the model tends to predict values from the middle of the $[1, 4]$ range, avoiding its edges. In addition, the problems described make the results even worse on unseen evaluation data.

NER + NLI. As you can conclude from Table 5.2, NER features, having poor single performance, add significant improvement in correlation being combined with NLI features. To obtain this result we have taken the features used in the best-scored NLI and NER models. For classification Gradient Boosting ML model was used as it had given the highest results for both approaches.

Additional study Additionally to the comparison of proposed models on the given datasets, we experiment with several techniques to improve the performance. The application of augmentation to the training part of the dataset improved the result of the best-performing model from 0.734 to 0.746, which is a slight improvement. It can be concluded that the performance of this model is not highly affected by unseen language pairs. The increase in score may be caused just by the increase in the number of training samples. A comparison of the results with and without augmentation can be found in Table 5.7.

The results for stacking of the models can be found in Table 5.8. Also, in this table, the results for the combination of the two improvement techniques are

	Stacking	Stacking + Augm.
TrEncCLS, TrEncCosSim	0.749 ± 0.022	0.752 ± 0.019
TrEncCLS, TrEncCosSim, NLI	0.757 ± 0.023	**0.763** ± 0.015
TrEncCLS, TrEncCosSim, NLI, NER	0.755 ± 0.021	**0.763** ± 0.020

Table 5.8: Comparison of the results for different ensembles with and without augmentation on the evaluation dataset. Pearson correlation with 0.95% confidence intervals. The names of TransformerEncoder models were shortened. The pre-trained models for TransformerEncoder approaches are xlm-roberta-large and xlm-mlm-17-1280 respectively.

provided.

The stacking technique in combination with augmentation showed a significant score improvement. It can be noticed that the addition of the predictions obtained with the NER model gives no increase in score. Overall, the augmentation together with stacking gave the average 4% improvement to the result of the TransformerEncoderCosSim model. There is no overlap in confidence intervals.

5.8 Fake News Detection using New Multilingual Text Similarity

We incorporate the proposed metric for multilingual news similarity in this Chapter into fake news detection systems proposed in the previous Chapter 4. We take several baselines from previous experiments: monolingual evidence compared with cosine similarity with rank (ME Emb.+Rank), cross-lingual compared with cosine similarity evidence with rank (CE Emb.+Rank), all linguistic features (All ling.), and combination of all linguistics features with cross-lingual evidence compared with cosine similarity (All ling.+CE Emb.+Rank). We substitute the previously used metrics for cross-lingual news comparison with the best one explored in this Chapter – TransformerEncoderCosSim (TrCosSim). The results are presented in Table 5.9.

The usage of cross-lingual evidence again improves over monolingual baselines. The cross-lingual evidence based on new TrCosSim metric outperforms the baseline

FakeNewsAMT			
	Pre.	Rec.	F1
ME Emb.+Rank	0.539	0.593	0.592
CE Emb.+Rank	**0.872**	**0.864**	**0.864**
TrCosSim+Rank	0.851 ± 0.052	0.850 ± 0.041	0.846 ± 0.053
All ling.	0.739	0.739	0.739
All ling.+CE Emb.+Rank	**0.940**	**0.937**	**0.937**
All ling.+TrCosSim+Rank	0.854 ± 0.062	0.851 ± 0.048	0.847 ± 0.041
Celebrity			
	Pre.	Rec.	F1
ME Emb.+Rank	0.552	0.550	0.550
CE Emb.+Rank	0.631	0.620	0.619
TrCosSim+Rank	**0.761** ± 0.042	**0.780** ± 0.051	**0.775** ± 0.039
All ling.	0.750	0.750	0.750
All ling.+CE Emb.+Rank	**0.801**	0.800	**0.800**
All ling.+TrCosSim+Rank	0.780 ± 0.046	**0.801** ± 0.052	0.787 ± 0.039
ReCOVery			
	Pre.	Rec.	F1
ME Emb.+Rank	0.794	0.798	0.793
CE Emb.+Rank	0.829	0.829	0.829
TrCosSim+Rank	**0.851** ± 0.024	**0.897** ± 0.006	**0.878** ± 0.015
All ling.	0.875	0.874	0.870
All ling.+CE Emb.+Rank	**0.916**	0.916	0.916
All ling.+TrCosSim+Rank	0.895 ± 0.012	**0.956** ± 0.012	**0.924** ± 0.007

Table 5.9: Results of integration of new metrics into cross-lingual evidence (CE) features for fake news detection. Scores for all the methods studied in this thesis are provided with 95% confidence intervals.

based on only monolingual evidence for all datasets and previous cross-lingual news comparison based on cosine similarity between work embeddings for Celebrity and ReCOVery datasets.

The comparison with the strong baseline based on all linguistic features baseline, the additional usage of TrCosSim cross-lingual evidence also give performance improvement. However, for FakeNewsAMT dataset, the usage of embeddings-based CE feature still shows the best result. For Celebrity dataset, both cross-lingual

evidence features show almost the same Recall. Finally, the results in ReCOVery dataset illustrates that fake news classification can be significantly improve by the addition of CE feature based on TrCosSim measurement.

As a result, we can claim that the usage of TrSocSim metric for cross-lingual news similarity measurement is more beneficial then the usage of only cosine similarity between multilingual embeddings. As previous results showed, TrSocSim metric is more scalable to different languages and more stable for zero-shot set up.

5.9 Demonstration System

As previously discussed the cross-lingual comparison of news can be used to demonstrate a user's different point of view on some event in different languages, help a user critically asses the news, and explain the decision of the automated fake news classification system.

We make a system demonstration of such a platform where a user can enter in a text field his or her request (some news title) and receive a comparison of information across several languages. The title page of such a system is provided in Figure 5-6.

Figure 5-6: Starting page of a system for cross-lingual news comparison.

After the user enters the request, the system translates the provided news title into several preassigned languages and scrapes search results. As a result, the user

will get a table with scrapped news contents, their translation into English (for unification), and similarity score as it is shown in Figure 5-7.

Figure 5-7: Comparison of cross-lingual news according to the user's request.

We believe that the proposed system will allow indifferent users to read several pieces of news in multiple languages and form a more informed opinion about the information he or she has found.

5.10 Summary

We presented a comprehensive comparison of several approaches to address the problem of measurement of similarity between multilingual and cross-lingual news pairs. The dataset of news pairs is constructed of texts from 10 languages from different language families. Moreover, 3 out of 10 presented languages appear only in the evaluation set pushing to develop metrics that can be easily scaled to new unseen languages.

Firstly, we tested the approach based on text embeddings from Transformer-based models. Secondly, we addressed the task as Natural Language Inference (NLI) task and applied corresponding models. Thirdly, we thought about more interpretable metrics based on Named Entities that incorporate the most important information in news text: location (`LOC`), organizations (`ORG`), and person entities (`PER`). We evaluated all proposed approaches based on Pearson correlation with

manual annotations.

The best results of 0.73 on the evaluation set showed the TransformerEncoderCosSim approach. NLI-based approaches showed compatible results when specifically fine-tuned on the data. However, NER-based approaches looked like quite promising models with a high possibility to interpret the result, they performed poorly in comparison to TransformerEncoderCLS and NLI-based approaches. The reason for such performance can be still not accurate named entities extraction for different languages. There is room for improvement in this approach with the development of more stable named entities extractors for a more diverse set of languages.

The performance of all metrics drops on the evaluation set because of the new unseen languages. However, Transformer-based embeddings showed the best stability. Modern Transformer-based multilingual models were pre-trained in a big amount of languages as it was discussed in Chapter 2. There is still room for improvement as well to make these models equally well-performed for all languages.

All the metrics benefit from data augmentation and stacking. That shows that these techniques should be included in such multilingual and cross-lingual news similarity metrics development.

Finally, we integrated new metrics into fake news detection systems proposed in Chapter 4. Baseline fake news detection benefit from the usage of cross-lingual evidence feature based on TransformerEncoderCosSim approach. Moreover, for ReCOVery dataset the substitution of CE metric with the proposed approach helps to achieve the highest F_1 score for fake news classification. In future research, the diversity of languages for fake news detection should be explored. All discussed in this Part metrics are worse to be continued to work on with more data available with more language presented.

Part II

Methods for Texts Detoxification

6

Task Introduction

In this part, we provide broad research for transferring the style of texts from toxic to neutral or in other words text detoxification task answering the research question **Q2**. The contributions of this part are the follows:

1. The **new method for parallel dataset collection** for detoxification is proposed.
2. The dataset collection method is tested for **two languages**.
3. The **new detoxification methods** based on parallel detoxification dataset for monolingual detoxification are explored achieving SOTA results.
4. **Multilingual** and **cross-lingual detoxification** methods are explored.
5. The study about **correlation between automatic and human evaluation** of detoxification models is conducted.

The collected parallel datasets dataset[1,2] and SOTA detoxification models together with multilingual experiments[3] are available online.

[1] https://github.com/s-nlp/paradetox
[2] https://github.com/s-nlp/russe_detox_2022
[3] https://github.com/s-nlp/multilingual_detox

6.1 Task Motivation

Global access to the Internet has enabled the spread of information throughout the world and has offered many new possibilities. On the other hand, alongside the advantages, the exponential and uncontrolled growth of user-generated content on the Internet has also facilitated the spread of toxicity and hate speech. Much work has been done in the direction of offensive speech detection [D'Sa et al., 2020, Schmidt and Wiegand, 2017, Pamungkas and Patti, 2019]. However, it has become essential not only to detect toxic content but also to combat it. While some social networks block sensitive content, another solution can be to detect toxicity in a user's text while the user types it and offer a non-offensive version of this text. This task can be considered as a Text Style Transfer (TST), where the source style is toxic, and the target style is neutral/non-toxic. Examples of such rewriting are shown in Table 6.1.

Toxic Text	**Detoxified Text**
*After all it's hard to get a job if your st**id.*	After all it's hard to get a job if you are incompetent.
*Go ahead ban me, i don't give a s**t.*	It won't matter to me if I get banned.
*Well today i f**king fr**king learned something.*	I have learned something new today.

Table 6.1: Examples of how real-life toxic comments can be detoxified.

The task of style transfer is the task of transforming a text so that its content and the majority of properties stay the same, and one particular attribute (style) changes. This attribute can be the sentiment [Shen et al., 2017, Melnyk et al., 2017], the presence of bias [Pryzant et al., 2020], the degree of formality [Rao and Tetreault, 2018], etc. The survey by Jin et al. [2020] provides more examples of style transfer applications. The detoxification task has already been tackled by different groups of researchers [Nogueira dos Santos et al., 2018, Tran et al., 2020], as well as a similar task of transforming text to a more polite form [Madaan et al., 2020].

There are multiple real-life cases of major commercial companies fighting offen-

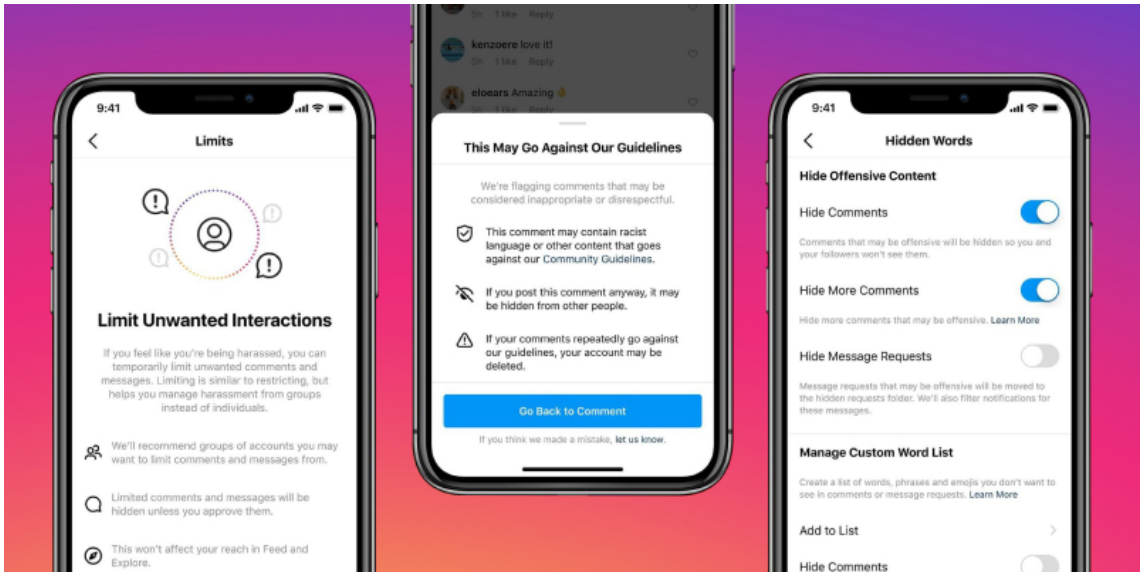


Figure 6-1: The example from Instagram how social networks are handling the fight ith toxic speech.

sive and toxic content. For instance, Facebook is testing models that can identify arguments in groups so that group administrators can help to alleviate such situations.[4] The group administrator will receive an alert about a conflict as it starts and can limit the maximum frequency of comments for some group members or posts. Instagram has also presented tools to filter abusive messages (Figure 6-1).[5] They can help to filter the direct messages based on a list of offensive words, phrases, and emojis. The Russian social network VK[6] has also presented[7] a way to not only detect offensive language but also prevent offensive messages from being posted. The proposed technique makes suggestions for users to replace rude words with more neutral stickers.

As we can see, the task of fighting toxic speech is quite important and relevant today. The methods that we propose in this work can be used in several scenarios. While, in VK, users are already asked to replace rude words with stickers, our methods can suggest a more neutral version of a message instead of a toxic message written by a user (see Figure 6-2a). In this case, the user will be able to choose

[4] https://edition.cnn.com/2021/06/16/tech/facebook-ai-conflict-moderation-groups/index.html
[5] https://about.instagram.com/blog/announcements/introducing-new-tools-to-protect-our-community-from-abuse
[6] https://vk.com
[7] https://tjournal.ru/internet/371142-instagram-vnedrit-filtr-oskorbitelnyh-soobshcheniy-funkciya-nacelena-na-znamenitostey

whether they would like to send a toxic message or a neutral one. Thus, the user can first express their emotions in a toxic text and, after their anger has been reduced, they can choose a more civil paraphrase of the toxic message. However, the final decision will be up to the user. We should also note that the notions of toxicity and civility are not hard-coded in our methods. The acceptability is fully data-driven—our detoxification methods can be trained in a different language or a specific dialect, where the criteria of toxicity can be different from the results reported in this work.

(a) (b)

Figure 6-2: Example of use cases where the detoxification technology can be applicable. (**a**) Offering the user a more civil version of a message. (**b**) Preventing chatbots from being rude to users when trained on open data.

Another field of application of our models is the development of chatbots. Nowadays, many companies are using chatbots for automating answers to frequently asked

user questions. Some of these chatbots can be constantly fine-tuned on the open user-generated data (e.g., posts from social media). There exist multiple cases of such chatbots becoming rude, e.g., the Oleg chatbot by Tinkoff Bank suggested that a user should have her fingers cut off.[8] Such situations cause both user frustration and damage to the company's reputation. To prevent this, our detoxification techniques can be used to filter the offensive messages generated by a chatbot and replace them with more civil messages conveying the same sense (see Figure 6-2b).

6.2 Problem Statement

In this section, we first look into the various definitions of toxicity and then formally define the task of text style transfer.

6.2.1 Definition of Toxicity

There exists a large body of work on toxicity detection in NLP. "Toxicity" is used as an umbrella term for almost any undesirable behavior on the Internet. It is intuitively understood as behavior that can offend, insult, or cause harm. This definition is too vague since the same message can be considered insulting or benign to different people depending on their preferences and background. Therefore, researchers usually further divide toxicity into subtypes.

The Jigsaw dataset [Jigsaw, 2018] contains six non-exclusive classes: *toxic*, *severe toxic*, *obscene*, *threat*, *insult*, *identity hate*. Other works partially adopt this typology. However, the semantics of classes may differ. Zampieri et al. [2019] call a message "offensive" if it contains profanities or targeted offenses. On the other hand, the Jigsaw dataset [Jigsaw, 2018] does not consider a message offensive if it contains obscenities but they are not targeted at any person or group of people. Some other datasets also distinguish between using obscene words for insulting someone and simply for expressiveness. One such example is the dataset collected by Wiegand et al. [2018]. It has a label, *offense*, that stands for any insult or use of obscene words.

[8] https://vc.ru/flood/71460-za-pervyy-den-raboty-pomoshchnik-oleg-ot-tinkoff-banka-nauchilsya-rugatsya

This class is further divided into three subclasses, *abuse*, *insult*, and *profanity*, where *profanity* is a non-toxic use of obscene words, and *insult* and *abuse* are both toxic messages that differ in gravity.

This gravity-based division can be found in other works. Unlike Wiegand et al. [2018], in the majority of works, a grave insult is referred to as *hate speech*. Fortuna and Nunes [2018] define hate speech as having a particular target (groups of people of a particular race, ethnicity, gender, and other innate characteristics) and aiming at attacking and diminishing the target groups. Other works on hate speech [Waseem and Hovy, 2016, Davidson et al., 2017, Basile et al., 2019] provide similar definitions. Many research works concentrate solely on hate speech, because, on one hand, it is one of the gravest and most dangerous types of undesired behavior. On the other hand, due to its salient features, it is relatively easy to identify, and the agreement of annotators is usually high.

Input text	**Toxicity type**
*clearly the french are a nation of m**ogynists*	direct racism ✘
*lying anti american m*slim m***thpiece*	direct racism ✘
*your *gnorance makes me laugh.*	passive aggressiveness ✘
*i think sen ron johnson need to *xamine his brain.*	passive aggressiveness ✘
*you s*ck sand n***er p***y!*	severe toxic ✘
*f*ck off you stupid *spy a***ole*	severe toxic ✘
*f**k you, i wont do what you tell me.*	obscene ✔
*what a chicken c**p excuse for a reason.*	toxic ✔

Table 6.2: Examples of different types of toxicity and specification of that one which we are handling in this work.

In contrast, several works deal with microaggressions [Breitfeller et al., 2019]—the "mildest" toxicity, which is not even recognized as such by a large percentage of respondents. Breitfeller et al. [2019] build upon a classification of microaggressions presented by DW et al. [2007] and defines some themes of microaggressions, such as using stereotypes, objectification, denial of a lived experience, etc. The authors of works on microaggressions often use a data-driven approach—in particular, Bre-

itfeller et al. [2019] and Han and Tsvetkov [2020] report using the website[9], which contains self-reports on microaggressions. Lees et al. [2021] explain microaggressions to crowd workers by contrasting them with open aggression. They also provide examples of different types of microaggressions and suggest trying to imagine the emotions of dialogue participants.

Other types of toxicity are not as well agreed upon as hate speech. Although many datasets of toxic messages have detailed annotation guidelines, the annotation remains subjective. The reason is that the guidelines sometimes have to appeal to the annotators' intuition regarding what is toxic, and this intuition differs for people with different backgrounds.

Our approach to defining toxicity is somewhat similar to that of Breitfeller et al. [2019]. We adopt the data-driven approach. In other words, we consider a message toxic if it is considered toxic by annotators. Since we have toxic datasets at hand, we simply follow the labeling provided there. Although there is no information on the labeling process for these datasets, we suggest that they were labeled using the same "intuitive" guidelines as the majority of other datasets. Similarly, when creating a parallel dataset, we rely on our intuition of what is offensive. We provide the comparison of the examples of different types of toxicity in Table 6.2 to provide intuition with which types we do not work and which cases we want to handle in the detoxification task described in this work.

6.2.2 Definition of Text Style Transfer

The definition of *textual style* in the context of NLP is vague [Tikhonov and Yamshchikov, 2018]. One of the first definitions of style refers to how the sense is expressed [McDonald and Pustejovsky, 1985]. However, in our work, we adhere to the data-driven definition of style. Thus, the style simply refers to the characteristics of a given corpus that are distinct from a general text corpus [Jin et al., 2020]. The style is a particular characteristic from a set of categorical values: {`positive`, `negative`} [Shen et al., 2017], {`polite`, `impolite`} [Madaan et al., 2020], {`formal`, `informal`} [Rao and Tetreault, 2018]. It is commonly assumed that this textual characteristic is mea-

[9] https://www.microaggressions.com

surable using a function $\sigma(x_i) \rightarrow s_i$ that obtains as input text x_i and returns the corresponding style label s_i. For instance, it can be implemented using a text classifier.

Let us assume a discrete set of styles $S = \{s_1, ..., s_k\}$. For simplicity, let us assume that S contains only two mutually exclusive styles (source and target, e.g., toxic/neural or formal/informal): $S = \{s^{src}, s^{tg}\}$. Let us consider two text corpora $D^{src} = \{d_1^{src}, d_2^{src}, ..., d_n^{src}\}$ and $D^{tg} = \{d_1^{tg}, d_2^{tg}, ..., d_m^{tg}\}$ belonging to the source and target styles s^{src} and s^{tg}, respectively. For each text d_i, let us assume that it has a style s_i measurable with the function $\sigma : D \rightarrow S$. There also exists a binary function $\delta : D \times D \rightarrow [0, 1]$ that indicates the semantic similarity of two input texts and a unary function $\psi : D \rightarrow [0, 1]$ that indicates the degree of the text fluency. In general, the sizes of the source and the target corpora D^{src} and D^{tg} are different ($n \neq m$) and the texts in them are not aligned, i.e., in general, $\delta(d_i^{src}, d_i^{tg}) \neq 1$. If $n = m$ and $\delta(d_i^{src}, d_i^{tg}) = 1$ for all texts, this is a special case of a parallel style-aligned corpus. Given the introduced notations, we define the task of textual style transfer (TST) as follows:

Definition 2 *A **text style tranfer** (TST) model is a function* $\alpha : S \times S \times D \rightarrow D$ *that, given a source style* s^{src}*, a target style* s^{tg}*, and an input text* d^{src}*, produces an output text* d^{tg} *such that:*

- *The style of the text changes from the source style* s^{src} *to the target style* s^{tg}*:* $\sigma(d^{src}) \neq \sigma(d^{tg})$*,* $\sigma(d^{tg}) = s^{tg}$*;*
- *The content of the source text is saved in the target text as much as required for the task:* $\delta(d^{src}, d^{tg}) \geq t^{\delta}$*;*
- *The fluency of the target text achieves the required level:* $\psi(d^{tg}) \geq t^{\psi}$*,*

where t^{δ} *and* t^{ψ} *are the threshold values for the content preservation* (δ) *and fluency* (ψ) *functions. They can be adjusted to the specific task.*

For instance, when removing the toxicity from a text, we inevitably change its meaning, so full content preservation cannot be reached. However, we should attempt to save the content as much as possible and adjust t^{δ} to the needs of this

task. At the same time, it is not always important for the resulting text to be ideally fluent and grammatically correct so that $\psi(d^{tg}) = 1$. When writing messages on the Internet, people often make grammatical mistakes or typos. Therefore, it is enough for the fluency score $\psi(d^{tg})$ to be better than some threshold $t^{\psi} > 0$.

Thus, the task of obtaining a TST model with the best parameters set may be viewed as maximizing the probability $P(d^{tg}|d^{src}, s^{src}, s^{tg})$ given the three above-mentioned constraints based on parallel or non-parallel text corpora D^{src} and D^{tg}.

6.3 Related Work

Style transfer was first proposed and widely explored for images [Gatys et al., 2016]. However, the task of text style transfer has gained less attention, partly due to the ambiguity of the term "style" for texts. Nevertheless, there exists a large body of work on textual style transfer for different styles. All the existing methods can be divided into techniques that use parallel training corpora and those using only non-parallel data. The latter category is larger because pairs of texts that share content but have different styles are usually not available. At the same time, it is relatively easy to find non-parallel texts of the same domain with different styles (e.g., positive and negative movie reviews, speeches by politicians from different parties, etc.).

6.3.1 Unsupervised TST approaches

A relatively easy yet efficient style transfer method is to leave the sentence intact and manipulate only individual words associated with the style. **Delete-Retrieve-Generate** (DRG) framework [Li et al., 2018a] was the first effort to perform such a transfer. It proposes four methods based on this principle. **Delete** part separates words in the sentence into style markers and content words and in order to do that n-grams that affect the style of the sentence the most are deleted. Formally, for any style, $s \in S$ impact of a concrete n-gram $g \in D$ is defined as:

$$imp(g, s) = \frac{\text{count}(g, D_s) + \lambda}{\sum_{\hat{s} \in S; \hat{s} \neq s} \text{count}(g, D_{\hat{s}}) + \lambda} \tag{6.1}$$

Here λ is a smoothing parameter, $\text{count}(g, D_s)$ is a counter of presence of n-gram g in a text corpus D_s. Marker s is considered as a style marker if and only if $imp(g, s) \geq \theta$, where θ is a threshold that can be manually specified. The text x with deleted n-gram can be depicted as $\text{del}(x, s_{source})$.

The next method that was introduced is called **Retrieve**. It locates a text x_{target} in corpus D which is nearly identical to the removed one that has the same target style:

$$x_{\text{target}} = \underset{x' \in D_{\text{target}}}{\text{argmin}} \, \text{dist}\left(\text{del}(x, s_{\text{source}}), \text{del}(x', s_{\text{target}})\right) \tag{6.2}$$

DRG-RetrieveOnly retrieves a sentence with the opposite style which is similar to the original sentence and returns it, and **DRG-TemplateBased** takes the style attributes from it and plugs them into the original sentence. Here, the performance depends on the methods for the identification of style markers and retrieval of replacements. Words associated with style are typically identified either based on their frequencies as in the original paper, some works use attention weights as features [Sudhakar et al., 2019].

Alternatively, style transfer can use Masked Language Modelling (MLM). An MLM trained on a dataset with style labels picks a replacement word based not only on the context but also on the style label. An example of such model is **Mask & Infill** [Wu et al., 2019b]. A masking step finds potential attribute markers in a text by selecting tokens with higher attention weights and selecting the best with a pre-trained classifier. At the infill step, masked tokens are replaced with tokens conditioned both on a context and target label.

To do that, a language model is trained with respect to the minimum reconstruction error of the replaced tokens. If $\hat{a}$ is the target attribute, $\boldsymbol{x} = [x_1, \ldots, x_n]$ - an input sequence of n tokens, $\boldsymbol{m} = [m_{i_1}, \ldots, m_{i_k}]$ - a set of masked tokens, then, $\overline{\boldsymbol{x}} = \boldsymbol{x} \backslash \boldsymbol{m}$ - is a set of context tokens. Finally, the language model is trained to recreate the original sentence based on the context $\overline{\boldsymbol{x}}$ and target style characteristic

$\hat{a}$:

$$\mathcal{L} = - \sum_{a \in \mathcal{A}, m_{i_k} \in \boldsymbol{m}} \log p\left(m_{i_k} | \overline{\boldsymbol{x}}, a\right) \tag{6.3}$$

Here $a \in \mathcal{A}$ is a certain style (class) a from a set of styles (classes) $\mathcal{A}$, m_{i_k} is k^th masked token. After training it is assumed that language model takes sentence with masked token $\overline{\boldsymbol{x}}$ and a style (class) label $\hat{a}$ and output token probabilities for all masked tokens: $\{p(m_i|\hat{a}, \overline{\boldsymbol{x}})\}_{i=1}^{k}$.

Another similar model of this type is described by [Malmi et al., 2020]. It has a more complicated structure: there, two MLMs trained on corpora of different styles perform replacements jointly.

Improving the idea introduced for MLM, in our work [Dale et al., 2021] for unsupervised TST there is presented **ParaGedi** – a modification of a GeDi [Krause et al., 2020] model modified for a style-specific text generation. The intuition behind the idea of this model is presented in Figure 6-3.

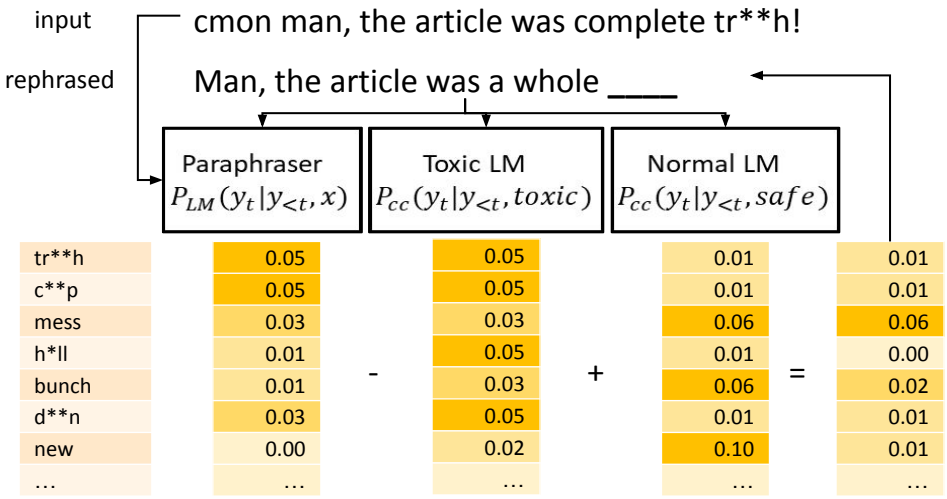


Figure 6-3: Visualization of the idea behind ParaGedi for unsupervised TST [Dale et al., 2021].

GeDi improves conditional generation by using a small language model as a discriminator that controls the generation [Krause et al., 2020]. Being trained with respect to style labels for each sentence, the discriminator can model style-conditioned word distributions and, hence, control the generation of style-specific text. The gen-

eration is also done in a specific way: for each new token distribution is first predicted by the main language model and then modified by a discriminator via Bayes Rule:

$$P(x_i|x_{<i}, c) \propto P_{\text{LM}}(x_i|x_{<i})P_{\text{D}}(c|x_i, x_{<i}), \tag{6.4}$$

where x_i in the formula above is the current token that is being generated, c is desired style (class) attribute, and $x_{<i}$ is the prefix (e.g. already generated text). On the right-hand side, the last term stands for style-conditional (class-conditional) discriminator and the first term (LM) is the generative language model itself. The probability distribution given by a discriminator model is conditioned both on the desired style (class) and on the undesired one.

ParaGeDi follows the concept of GeDi, but replaces the original generative language model with a model trained to paraphrase the text with respect to meaning preservation (Figure 6-3). Thus, the following probability is being modeled:

$$\begin{aligned} P(x_i|x_{<i}, x_{\text{input}}, c) &\propto P_{\text{LM}}(x_i|x_{<i}, x_{\text{input}})P(c|x_i, x_{<i}, x_{\text{input}}) \\ &\approx P_{\text{LM}}(x_i|x_{<i}, x_{\text{input}})P_{\text{D}}(c|x_i, x_{<i}) \end{aligned} \tag{6.5}$$

Here x_i is an i^{th} token being generated, $x_{<i}$ is a prefix (already generated text), x_{input} is input text. The last approximation is an assumption that, however, allows to train the paraphraser and the generative language model independently. Also, Krause et al. [2020] suggests ranking generation candidates similar to condBERT in order to improve generation. In this case, a ranker is a toxicity classifier that allows a selection least toxic candidates for a generation.

Training of ParaGeDi mostly follows original training procedure of GeDi: loss function is viewed as a linear combination (with adjustable hyperparameter λ) of

generative $\mathcal{L}_{\text{G}}$ and discriminative $\mathcal{L}_{\text{D}}$ losses.

$$\begin{cases} \mathcal{L}_{\text{G}} = -\frac{1}{n}\sum_{j=1}^{n}\frac{1}{T_j}\sum_{j=1}^{T_j}\log P(x_i^j|x_{<i}^j, c) \\ \mathcal{L}_{\text{D}} = -\frac{1}{n}\sum_{i=1} n \log P(c^i|x_{1:T_i}^i) \\ \mathcal{L} = \lambda\mathcal{L}_{\text{G}} + (1-\lambda)\mathcal{L}_{\text{D}} \end{cases} \tag{6.6}$$

In contrast to previous point-wise editing models, there exist end-to-end architectures for style transfer. They encode the source sentence, then manipulate the resulting hidden representation in order to incorporate a new style, and then decode it. Some of them disentangle the hidden representation into the representation of content and style [John et al., 2019]. The others force the encoder to represent style-independent content [Hu et al., 2017]. Alternatively, the model **DualRL** by [Luo et al., 2019] performs a direct transfer from the source to the target style. The task is paired with the dual task (back transfer to the source style) which allows models to train without parallel data.

The Deep Latent Sequence Model (**DLSM**) model by [He et al., 2020] uses amortized variational inference to jointly train models for the primal and dual tasks. The authors assume that each observed sentence is generated from an unobserved parallel sentence in the opposite domain. So, if we have observed data from domain D_1 as $X = \{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$ and from domain D_2 as $Y = \{y^{(m+1)}, y^{(m+2)}, ..., y^{(n)}\}$, then pseudo-parallel unobserved samples to each domain will be $\hat{Y} = \{\hat{y}^{(1)}, \hat{y}^{(2)}, ..., \hat{y}^{(m)}\}$ generated from prior $p_{D_1}(\hat{X})$ and $\hat{X} = \{\hat{x}^{(m+1)}, \hat{x}^{(m+2)}, ..., \hat{x}^{(n)}\}$ generated from prior $p_{D_2}(\hat{Y})$. Let us name $\theta_{x|\hat{y}}$ and $\theta_{y|\hat{x}}$ represent the parameters of the two transduction distributions respectively. That all give us a joint likelihood:

$$p(X, \hat{X}, Y, \hat{Y}; \theta_{x|\hat{y}}, \theta_{y|\hat{x}}) = \\ \Big(\prod_{i=1}^{m} p(x^{(i)}|\hat{y}^{(i)}; \theta_{x|\hat{y}}) p_{D_2}(\hat{y}^i)\Big)\Big(\prod_{j=m+1}^{n} p(y^{(j)}|\hat{x}^{(j)}; \theta_{y|\hat{x}}) p_{D_1}(\hat{x}^j)\Big) \tag{6.7}$$

The log marginal likelihood of the data, which we will approximate during train-

ing, is:

$$\log p(X, Y; \theta_{x|\hat{y}}, \theta_{y|\hat{x}}) = \log \sum_{\hat{X}} \sum_{\hat{Y}} p(X, \hat{X}, Y, \hat{Y}; \theta_{x|\hat{y}}, \theta_{y|\hat{x}}) \tag{6.8}$$

The Stable Style Transformer (**SST**) method [Lee, 2020] trains a pair of sequence-to-sequence transformers for primal and dual tasks using the cross-entropy of a pretrained style classifier as an additional discriminative loss. In SST Delete process is independent of any predefined vocabulary (frequency-ratio method) or attention scores like in Li et al. [2018a]. Instead, an *Importance Score (IS)* is calculated for each token. Given an input sequence $\boldsymbol{x}$, the probability is given by a style classifier is

$$P(\boldsymbol{x}) = P_{\mathrm{C}}(c|\boldsymbol{x}) \tag{6.9}$$

Here c is certain style (class) label. That exact probability for a sequence $\boldsymbol{x}$ without a token x_i would be

$$P(\boldsymbol{x} \backslash x_i | x_i) = P_{\mathrm{C}}(c|\boldsymbol{x} \backslash x_i, x_i) \tag{6.10}$$

After calculating that probability for deleting every token in a sequence $\boldsymbol{x}$ we have a set of *importance scores* calculated in the following form:

$$IS(\boldsymbol{x} \backslash x_i) = P(\boldsymbol{x}) - P(\boldsymbol{x} \backslash x_i | x_i) \tag{6.11}$$

Importance Score indicates how each token x_i from a sequence $\boldsymbol{x}$ affects the overall style (class) of a sequence $\boldsymbol{x}$ predicted by a style classifier. Therefore, only tokens with the highest IS should be deleted. In order to control the deletion of tokens, α and β constraints are introduced. α is a threshold for the probability $P(\boldsymbol{x})$ and is used as an indicator that a sequence $\boldsymbol{x}$ is no longer of an original style (class). β is used to control the content of the sentence: if the threshold β is exceeded, a token could not be deleted. During generation special tokens $< start >$ and $< style >$ are used to start generation and follow a specific style. Since this method is positioned as unsupervised, reconstruction and style losses are optimized.

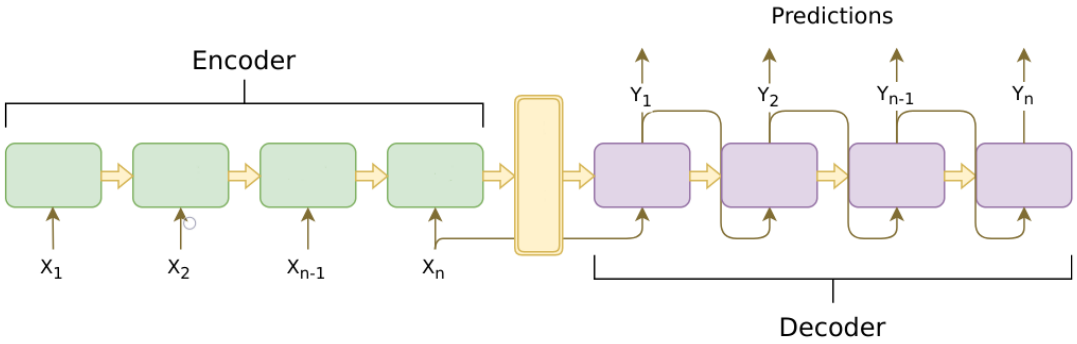


Figure 6-4: High-level illustration of a sequence-to-sequence architecture.

6.3.2 Supervised TST approaches

On the other hand, if there exists a corpus with parallel sentences $\{(d_1^{src}, d_1^{tg}), (d_2^{src}, d_2^{tg}), ..., (d_N^{src}, d_N^{tg})\}$ where $\delta(d_i^{src}, d_i^{tg}) = 1 \ \forall i \in [1, N]$, then style transfer can be formulated as a Sequence-to-sequence (seq2seq) task, analogously to supervised Neural Machine Translation (NMT), summarization, paraphrasing, etc.

Sequence-to-sequence (seq2seq) model usually consists of encoder and decoder (Figure 6-4). Both encoder and decoder are usually either RNNs [Rumelhart et al., 1985] or Transformer blocks [Vaswani et al., 2017]. Encoder is used to transform an input text sequence $\boldsymbol{x} = (x_1, \ldots, x_n)$ of length n first to a hidden representation z which is expected to be smaller than original sequence and preserve the content of an input. Decoder takes a hidden representation z as an input and then transforms to an output sequence $\boldsymbol{y} = (y_1, \ldots, y_n)$. Formally, the goal of a sequence-to-sequence language model is to estimate the probability:

$$p(y_1, y_2, \ldots, y_n|\boldsymbol{x}) = \prod_{i=1}^{n} p(y_i|y_{<i}, \boldsymbol{x}) \tag{6.12}$$

We previously discussed Transformer-based models in Section 2.2.3. The TST task can highly benefit from the usage of pre-trained LLM for Natural Language Generation (NLG) task.

One of the first works that used the advantages of Transformer for sequence generation tasks is **GPT** [Radford et al., 2019]. The model consists of a multi-layer Transformer decoder. Firstly, the model is trained in unsupervised mode with

standard language modeling objective to maximize the following likelihood:

$$L_1(X) = \sum_i \log P(x_i|x_{i-k}, ..., x_{i-1}; \theta) \tag{6.13}$$

where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters θ.

After training the model with the unsupervised objective, its parameters are adapted to the supervised target task. We assume a labeled dataset C. The inputs are passed through our pre-trained model to obtain the final transformer block's activation h_l^m, which is then fed into an added linear output layer with parameters W_y to predict y:

$$P(y|x^1, ..., x^m) = \text{softmax}(h_l^m W_y) \tag{6.14}$$

This gives us the following objective to maximize:

$$L_2(C) = \sum_{x,y} P(y|x^1, ..., x^m) \tag{6.15}$$

As such a LLM is already pretrained for different tasks, it can be used in *zero-shot* setup even for TST task (Figure 6-5). For instance, several examples of parallel data can be passed as a prompt in prefix specify in suffix that we want to solve paraphrasing task. Another approach can be indeed to fine-tune LLM for a specific dataset and task.

Text-to-Text Transfer Transformer (**T5**) [Raffel et al., 2020] is a large Encoder-Decoder transformer model. We follow the proposed text-to-text approach and formulate the task of supervised detoxification as a task of translation the toxic input sequence $\boldsymbol{d^{src}} = (x_1, \ldots, x_n)$ to the polite output sequence $\boldsymbol{d^{tg}} = (y_1, \ldots, y_m)$, optimizing cross-entropy loss:

$$\mathcal{L}_{CE}(\boldsymbol{d^{src}}, \boldsymbol{d^{tg}}) = \frac{1}{n} \sum_{i=1}^{n} -\log p_\theta(y_i|\boldsymbol{d^{src}}, \theta) \tag{6.16}$$

Where n is a length of an input sequence and θ are model parameters (weights).

Bidirectional and Auto-Regressive Transformer (**BART**) [Lewis et al., 2020] is

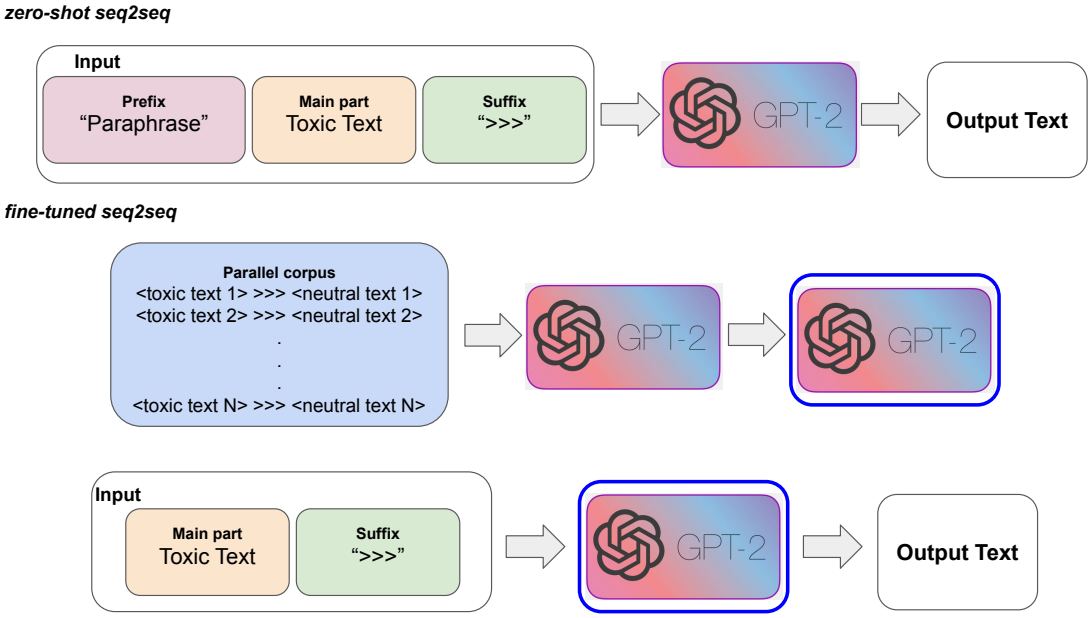


Figure 6-5: Pretrained seq2seq models (such as, for instance, GPT [Radford et al., 2019]) can be used in different setups: i) the model is taken as it is and the task is described only as textual prompts; ii) when there a parallel corpus exists, the model can be fine-tuned on a specific task.

itself a combination of bidirectional encoder that was introduced in BERT [Devlin et al., 2019] and an autoregressive decoder that was introduced in GPT Radford et al. [2019]. Originally BART was pre-trained on a denoising task and then fine-tuned on other downstream tasks. We use **BART** similarly to **T5** in a neural machine translation sequence-to-sequence manner by trying to translate the text written in toxic language to the text written in polite language.

Due to the reason that there are only a few parallel datasets for the TST task, there is not so much work dedicated to the development of seq2seq approaches for TST. For example, in the work by Krishna et al. [2020], a GPT-based model was fine-tuned on an automatically generated parallel corpus to transfer between multiple styles. Another example of addressing the TST task as a seq2seq generation task can be found for the formality transfer task in [Rao and Tetreault, 2018] and [Briakou et al., 2021b]. There was presented a parallel corpus of `formal` $\leftrightarrow$ `informal` pairs for four languages: English, French, Italian, and Brazilian Portuguese.

6.3.3 Detoxification

Detoxification of text is a relatively new style transfer task. The majority of the previous work is based on unsupervised TST approaches described in the section above. The first work on this topic by dos Santos et al. [2018] is an end-to-end seq2seq model trained on a non-parallel corpus with autoencoder loss, style classification loss, and cycle-consistency loss. More recent work by Tran et al. [2020] uses a pipeline of models: a search engine finds non-toxic sentences similar to the given toxic ones, an MLM fills the gaps that were not matched in the found sentences, and a seq2seq model edits the generated sentence to make it more fluent. Finally, Laugier et al. [2021] detoxify sentences by fine-tuning T5 as a denoising autoencoder with additional cycle-consistency loss. Dathathri et al. [2020] and Krause et al. [2020] approach a similar problem: preventing a language model from generating toxic text. They do not need to preserve the meaning of the input text.

Most approaches tested on detoxification rely only on unsupervised methods, i.e. models trained without parallel corpus so far. Moreover, all previous works were dedicated to solving the detoxification task only for the English language.

7

ParaDetox: A Parallel Detoxification Dataset

In this section, we present a new automated method for parallel dataset collection for the detoxification task. We tested the pipeline and in a result we present ParaDetox – a new Parallel detoxification dataset for English and Russian languages. We describe the details of tasks' design for annotators, design of markup quality control, and analyze the delivered data.

7.1 Task Definition

As it was stated in Section 6.3, the majority of style transfer methods and all previous detoxification methods are unsupervised. That means, that they are trained on non-parallel data with separate corresponding classes that are usually available for the classification task. We want to overcome this gap and suggest a new parallel detoxification dataset. The general motivation of parallel dataset collection can be expressed as follows:

Hypothesis 4 (H4) *Trained machine learning models on* ***parallel corpus*** *of detoxification samples will gain higher performance on a detoxification task than trained on non-parallel ones.*

Alongside this hypothesis, we want to tackle the problem of large amounts of

manual work that are usually required for collecting parallel data. For this reason, in the design of our proposed data collection pipeline, we pay attention to the automation of quality control of collected samples.

As a result, we want to create such a pipeline for parallel corpus collection that meets the following requirements (following the Definition 2 of Text Style Transfer):

R1: For each `toxic` input we get 1-3 `non-toxic` paraphrases.

R2: The content of `toxic` input and its `non-toxic` paraphrase is the same as much as it is possible.

R3: The style of created paraphrases is indeed `non-toxic`.

R4: Generated `non-toxic` paraphrases are fluent texts.

R5: The quality control of all the above statements is made without a manual check of experts.

7.2 Related Work

When collecting non-parallel style transfer corpora, style labels often already exist in the data (e.g. positive and negative reviews [Li et al., 2018b]) or its source serves as a label (e.g. Twitter, academic texts, legal documents, etc.). Thus, data collection is reduced to fetching the texts from their sources, and the corpus size depends only on the available amount of text.

Conversely, parallel corpora are usually more difficult to get. There exist parallel style transfer datasets fetched from "naturally" available parallel sources: the Bible dataset [Carlson et al., 2018] features multiple translations of the Bible from different epochs, and biased-to-neutral Wikipedia corpus [Pryzant et al., 2020] uses the information on article edits.

Besides these special cases, there exists a large style transfer dataset that was created from scratch. This is the GYAFC dataset [Rao and Tetreault, 2018] of informal sentences and their formal versions. While the task of generation of formal

Bible corpus [Carlson et al., 2018]	
Bible in Basic English	*Moses output*
Then Samuel gave him an account of everything, keeping nothing back. And he said, It is the Lord; let him do what seems good to him.	Then Samuel told him of all things not. And he said, It is Jehovah; let him do that which seemeth him good.
His legs were covered with plates of brass and hanging on his back was a javelin of brass.	His legs were covered with flakes of brass and hanged on his shoulder was a javelin of brass.
GYAFC [Rao and Tetreault, 2018]	
Informal	*Formal*
I'd say it is punk though	However, I do believe it to be punk.
Gotta see both sides of the story	You have to consider both sides of the story.

Table 7.1: Examples of existed parallel corpora for different text style transfer tasks: i) Bible corpus was collected naturally over centuries; ii) GYAFC corpus was generated via crowdsourcing, however verification was made manually.

sentences was given to crow workers, the validation of samples was done manually by the authors of the paper.

Since toxic-neutral pairs also do not occur in the wild and the manual validation of samples can gain a lot of resources, we follow the data collection setup with a notable difference – we replace expert validation of crowdsourced sentences with crowd validation and additionally optimize the cost. As a result, we get an automated pipeline for parallel detoxification collection that can be easily scaled to different languages and theoretically to different text style transfer tasks.

7.3 Crowdsourcing Tasks

We ask crowd workers to generate paraphrases and then evaluate them for content preservation and toxicity. Each task is implemented as a separate crowdsourcing project. We use the crowdsourcing platform Yandex.Toloka.[1]

[1] https://toloka.yandex.com

7.3.1 Task 1: Generation of Paraphrases

The first crowdsourcing task asks users to eliminate toxicity in a given sentence while keeping the content (see the task interface in Figure 7-1).

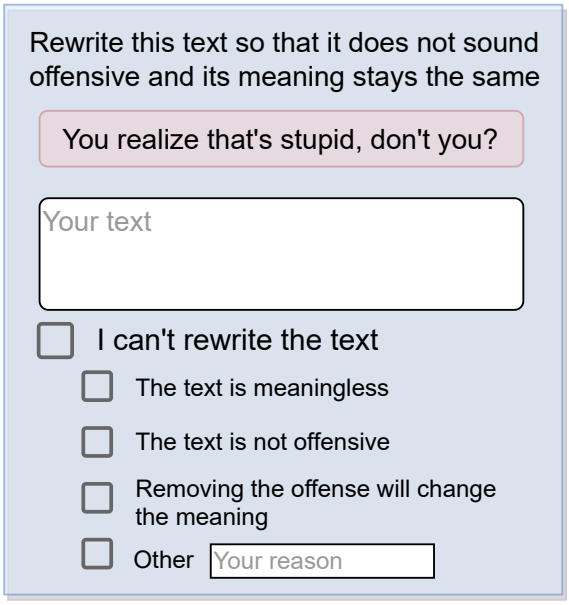


Figure 7-1: Interface of Task 1 (paraphrases generation).

Text that can be detoxified One of the main struggles in this task is to explain to annotators which type of toxicity we work with and which want to eliminate. We use the example-based approach. Namely, instead of definitions of what can be detoxified, we give users examples of pairs of `toxic` ↔ `non-toxic` sentences that we prepared by ourselves. In Table 7.2 the examples with explanations used for instructions are presented.

One of the important points to pay attention to is that it is crucial not to mix up sentiment and toxicity. Non-toxic sentences, in the paraphrasing result, can be negative by their sentiment (as in example *"You are an **iot."* it should not be changed to the opposite sentiment with *"You are a great guy."*). As we want to preserve the content as much as possible, it is important to save sentiment even if it is negative but to eliminate rude words.

Text that cannot be detoxified However, detoxification is not always possible. We talked about different toxicity types in Section 6.2.1 showing appropriate toxicity

Input Text	Paraphrase	Hint
*A st*pid society does stupid things and votes for stupid politicians.*	*The decisions of society are not always correct.*	Good paraphrase ✔
*A president who is an *diot.*	*An unsuitable president.*	Good paraphrase ✔
*How naive, s*lly rabbit.*	*You are naive.*	Good paraphrase ✔
*Just like that *diot nanakuli.*	*This isn't a nice example.*	Bad paraphrase ✘: Major change of sense
*You are an **iot.*	*You are a great guy.*	Bad paraphrase ✘: Change of sentiment
*Get f*cked b*tch sl*t h*re h*e sk*nk.*	*Get f*cked*	Bad paraphrase ✘: Preservation of toxicity

Table 7.2: Task 1 (Paraphrase Generation) examples used to provide understanding of style change requirement to crowd workers.

that we can handle in Table 6.2. Some sentences cannot be detoxified, because they do not contain toxicity, because they are meaningless, or because they consist of toxic intent. Thus, in some cases toxicity cannot be removed. Consider the examples:

- *Are you that d**b you can't figure it out?*
- *I've finally understood that wiki is nothing but a bunch of American r**ists.*

Not only the form but also the content of the messages are offensive, so trying to detoxify them would inevitably lead to a substantial change of sense. We prefer not to include such cases in the parallel dataset.

If workers have to detoxify all inputs without the possibility to skip them, a large proportion of the generated paraphrases will be of low quality. Thus, we add the control "I can't rewrite the text" and optional controls to indicate the reasons. You can refer to more examples of samples that could not be detoxified in Appendix B.6. More detailed instructions for this task with additional training examples you can find in Appendix B.2.1.

However, in this task, we cannot control in any way the quality of generated paraphrases directly – annotators can write in the text field any text. We can filter

obvious copypasted or random texts, however, the criteria that paraphrases should meet are way stricter. To control the quality of generated paraphrases, we pass them via the next tasks described below. The most important trick we use to ensure the quality is ***delayed payment***: the annotators are paid for this task not immediately after task page completion, but only *the quality of samples is confirmed by the next tasks.*

7.3.2 Task 2: Content Preservation Check

We show users the generated paraphrases along with their original variants and ask them to indicate if they have close meanings. Besides ensuring content preservation, this task implicitly filters out senseless outputs, because they do not keep the original content. The task interface is shown in Figure 7-2.

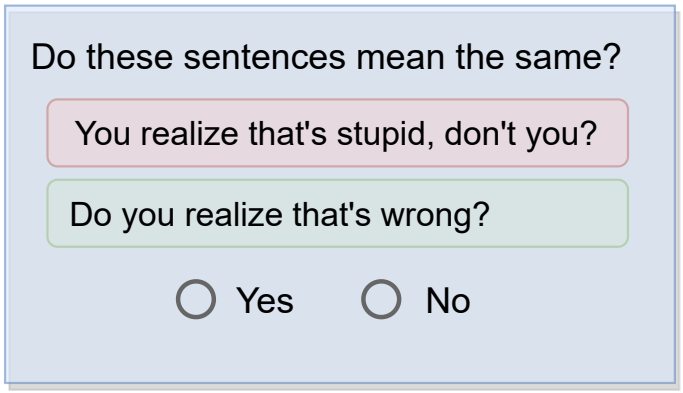


Figure 7-2: Interface of Task 2 (evaluation of content match).

As well as for Task 1, we demonstrate the task for annotators by example (Table 7.3). Detailed instructions and more training examples for this task you can find in Appendix B.2.2.

7.3.3 Task 3: Toxicity Check

Finally, we check if the workers succeeded in removing toxicity. We ask users to indicate if the paraphrases contain any offense or swear words (see Figure 7-3).

Here, it is again important to illustrate for crowd workers what we understand by toxic and non-toxic samples to ensure that we indeed get non-toxic samples in

Input Text 1	Input Text 2	Match	Hint
*and why the f**k did the manager tell the cop that?*	*And why the manager tell the cop that?*	✔	In the second sentence the word "f**k" was removed, this doesn't distort the sense.
*good , i m tired of seeing people ab*sing that s**t.*	*I'm tired of seeing people take advantage of that*	✔	"Take advantage" means the same as "ab*se" in this context, removing the word "s**t" doesn't distort the sense. This is a match.
*parliament is a l*me d*ck.*	*Parliament is unsuitable*	✔	The word "unsuitable" from sentence 2 is another way of saying "a l*me d*ck", the meaning of the sentences is the same.
*just another left*e c*ward with their panties in a twist.*	*kjhsdf uywoeur 87kjhdf.*	✘	The texts have nothing in common. This is not a match.
*after all most of these d**ks are millionares and can pay their own way.*	*these people are millionaires and they don't know how to do anything*	✘	The content of sentences differs: the first says millionaires can pay their own way, and the second states they don't know how to do anything. This is not a match.
*i'm vacation in the san juans, and f**k the winds were crazy.*	*i'm vacation in the san juans*	✘	The second sentence lacks some of the information from the first sentence (about the winds in San Juans). This is not a match.

Table 7.3: Task 2 (Content Preservation Check) examples used to provide understanding of content preservation requirement to crowd workers.

the final dataset. In Table 7.4, we provide the examples that are shown to crowd workers to understand toxic style. Detailed instruction and more training examples for this task you an find in Appendix B.2.3.

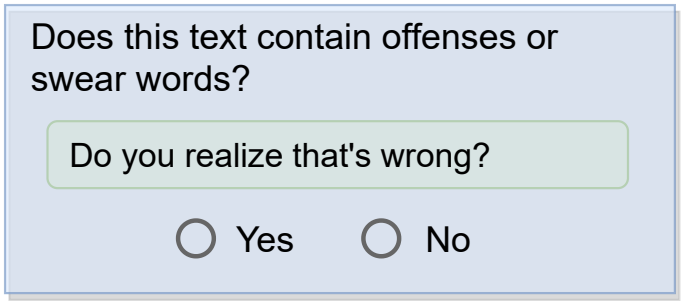


Figure 7-3: Interface of Task 3 (evaluation of toxicity).

Input Text	Is toxic	Hint
Last winter, we had really tough times.	✘	The text doesn't contain any offence or aggression.
The decisions of society are not always correct.	✘	The text criticises the society but doesn't offend anyone.
Hey, what is wrong with you?	✘	This is a neutral question which doesn't offend anyone.
*how the f**k do you manage that?*	✔	The text contains a swear word ("f**k").
*Can't wait to get my hands around all their n*cks.*	✔	The text contains a threat (the speaker wants to strangulate someone).
*a**holes , this is bulls**t!*	✔	The text contains rude words ("a**holes", "bulls**t").

Table 7.4: Task 3 (Toxicity Check) examples used to provide understanding of toxic style to crowd workers.

7.4 Crowdsourcing Settings

To ensure the good quality of collected dataset, we additionally pay attention to: i) input text characteristics; ii) training and examination of crowd workers; iii) fair payment for both annotators and us as stakeholders; iii) accumulation of markup answers and final selection of samples to the dataset. All the details of these steps are described below.

7.4.1 Preprocessing

To pre-select toxic sentences, we need a toxicity classifier. We consider a sentence toxic if the classifier confidence is above 0.8. To make the sentences easier for reading and rewriting, we choose the ones consisting of 5 to 20 tokens.

7.4.2 Quality Control

To perform paid tasks, users need to pass *training* and *exam* sets of tasks. Each of them has a corresponding *skill* – the percentage of correct answers. It is assigned to a user upon completing training or exam and serves for filtering out low-performing users. Besides that, users are occasionally given control questions during labeling. They serve for computing the *labeling skill* which can be used for banning low-performing and rewarding well-performing workers. The overall training and control pipeline is shown in Figure 7-4. It is used in **Tasks 2** and **3**.

In **Task 1**, we perform different quality control. We ban users who submit answers which are:

1. a copy of the input;
2. too short ($<$ 3 tokens) or too long (more than doubled original length);
3. contains too many rare words or non-words. The latter condition is checked as follows.

We compute the ratio of the number of whitespace-separated tokens and the number of tokens identified by the BPE tokeniser [Sennrich et al., 2016].[2] The rationale behind this check is that the BPE tokenizer tends to divide rare words into multiple tokens. If the number of BPE tokens in a sentence is two times more than the number of regular tokens, it might indicate the presence of non-words. We filter out these answers and ban users who produce them.

In addition to that, we ban malicious workers using built-in Yandex.Toloka tools:

1. **captcha**;

[2]We use the tokenizer of the BERT base uncased model (https://huggingface.co/bert-base-uncased)

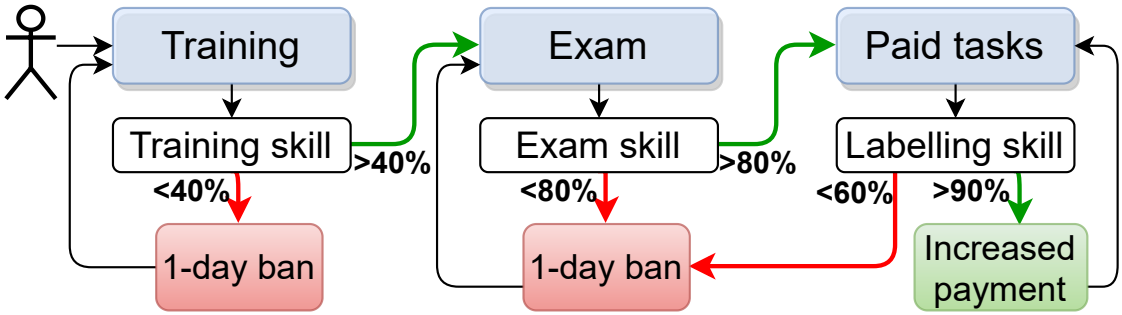


Figure 7-4: Training and quality control pipeline for Tasks 2 and 3.

2. number of **skipped questions**: we ban users who skip 10 task pages in a row;

3. **task completion time**: we ban those who accomplish tasks too fast (this usually means that they choose a random answer without reading)

7.4.3 Payment

In Yandex.Toloka, a worker is paid for a page that can have multiple tasks (the number is set by the customer). In Task 1, a page contains 5 tasks and costs \$0.02. In Tasks 2 and 3, we pay \$0.02 and \$0.01, respectively, for 12 tasks. In addition to that, in these tasks, we use skill-based payment. If a worker has the *labeling skill* of above 90%, the payment is increased to \$0.03 (Task 2) and \$0.02 (Task 3).

Tasks 2 and 3 are paid instantly, whereas in Task 1 we check the paraphrases before paying. If a worker indicated that a sentence cannot be paraphrased, we pay for this answer only if at least one other worker agreed with that. If a worker types in a paraphrase, we send it to Tasks 2 and 3 and pay only for the ones approved by both tasks. The payment procedure is shown in Figure 7-5.

7.4.4 Postprocessing

To ensure the correctness of labeling, we ask several workers to label each example. In Task 1, this gives us multiple paraphrases and also verifies the "I can't rewrite" answers. For Tasks 2 and 3, we compute the final label using the Dawid-Skene aggregation method Dawid and Skene [1979] which defines the true label iteratively giving more weight to the answers of workers who agree with other workers more

often. The number of people to label an example ranges from 3 to 5 depending on the workers' agreement.

Dawid-Skene aggregation returns the final label and its confidence. To improve the quality of the data, we accept only labels with the confidence of over 90% and do not include the rest in the final data.

7.5 Data Collection Pipeline

Summarizing all of the above, the final algorithm for parallel detoxification dataset collection is presented in Algorithm 2. The detailed schema of tasks connection and payment granting is illustrated in Figure 7-5.

Our proposed algorithm ensure all the requirements stated in Section 7.1:

R1: We get several paraphrases for one input as in every task, and Task 1 as well, the annotation pipeline provides overlap of several crowd workers for each sample.

R2: We check the content similarity with Task 2 saving into dataset samples only with high scores and high confidence.

R3: We check the change of style with Task 3 saving into dataset samples only with high scores and high confidence.

R4: The check of text fluency is done implicitly in Task 2 – when the text is non-fluent, it will be discarded from the dataset as the content is not similar to the original text.

R5: We create a pipeline where a check of all text style transfer requirements is done not with experts but with crowd workers. We ensure a high quality of markup execution by strict selection of annotators with training, examination, and control steps.

We test the proposed pipeline for two languages – English and Russian. That shows the scalability of the proposed approach in several languages. The results of these datasets' collection are presented in the next sections.

Algorithm 2 Parallel Detoxification Dataset Collection Pipeline.
Input: Collection of texts labeled as `toxic` for toxicity classification task.
Output: dataset of pairs `toxic` ↔ `non-toxic` texts.

```
function DATASET_PREPROCESSING(dataset_toxic)
    sentences := []
    for sample ∈ dataset_toxic do
        sentences.extend(sample.split())
    end for
    input_toxic_texts := []
    for sentence ∈ sentences do
        if toixicty_score(sentence) ≥ 0.8 AND 5 ≤ len(sentence) ≤ 20 then
            input_toxic_texts.append(sentence)
        end if
    end for
    return input_toxic_texts
end function
```

```
function PARADETOX_COLLECTION(dataset_toxic)
    input_toxic_texts := dataset_preprocessing(dataset_toxic)
    generated_paraphrases ← Task1(input_toxic_texts)
    content_input := []
    for input_toxic_text ∈ input_toxic_texts do
        for paraphrase ∈ generated_paraphrases.get(input_toxic_text) do
            content_input.append(input_toxic_text, paraphrase)
        end for
    end for
    content_similarities ← Task2(content_input)
    toxicity_input := []
    for (content_similarity, pair) ∈
          zip(content_similarities, content_input) do
        if content_similarity ≥ 90 AND
          Dawid-Skene(content_similarity) ≥ 90 then
            toxicity_input.append(pair)
        end if
    end for
    nontoxicity_scores ← Task3(toxicity_input)
    paradetox := []
    for (nontoxicity_score, pair) ∈
          zip(nontoxicity_scores, toxicity_input) do
        if nontoxicity_score ≥ 90 AND
          Dawid-Skene(nontoxicity_score) ≥ 90 then
            paradetox.append(pair)
        end if
    end for
    return paradetox
end function
```

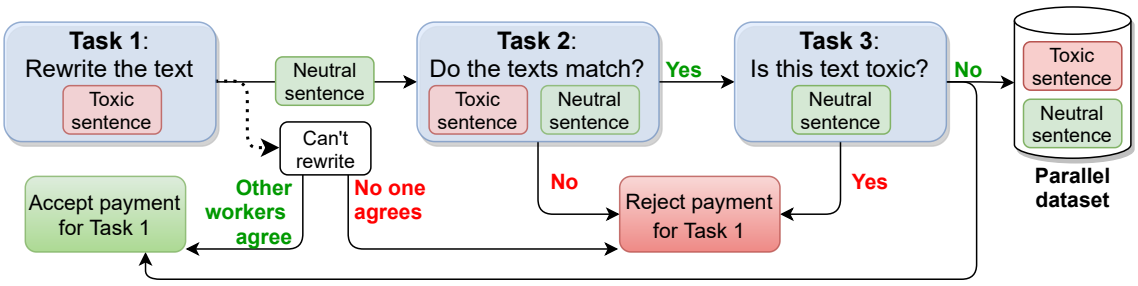


Figure 7-5: The pipeline of crowdsourcing for generation of detoxifying paraphrases.

7.6 English ParaDetox

Firstly, we collected English ParaDetox – a parallel detoxification dataset with 1–3 paraphrases for over 12,000 toxic sentences. The whole dataset is available online.[3] the interfaces used for English dataset collection are the same as used in previous Section 7.3 for crowdsourcing tasks description.

7.6.1 Data Analysis

We fetched toxic sentences from three sources: Jigsaw dataset of toxic sentences Jigsaw [2018], Reddit and Twitter datasets used by Nogueira dos Santos et al. [2018]. We selected 7,000 toxic sentences from each source and gave each of the sentences for paraphrasing to 3 workers. We get paraphrases for 12,610 toxic sentences (on average 1.66 paraphrases per sentence), 20,437 paraphrases total. Running 1,000 input sentences through the pipeline costs $41.2, and the cost of one output sample is $0.07. The overall cost of the dataset is $811.55.

The examples from dataset are shown in Table 7.5. We provide additional examples of gained samples in Appendix B.4.1. In addition to that, we provide some samples which could not be detoxified in Appendix B.6. The statistics of the paraphrases written by crowd workers are presented in Table 7.6.

The distribution of sentences from different datasets in the final data is not equal. Jigsaw turned out to be the most difficult to paraphrase. Fewer sentences from it are successfully paraphrased, making it the most expensive part of the collected corpus

[3] https://github.com/skoltech-nlp/paradetox

Original	as an american who thought it was a d*ck joke , thanks.
Paraphrases	as an american who thought it was a joke, thanks As an American who thought it was a bad joke, thanks
Original	for whatever reason , your comment just blew my f*cking mind.
Paraphrases	For whatever reason, your comment just blew my mind. for whatever reason, your comment just amazes me.
Original	what exactly is your f**king problem here?
Paraphrases	What exactly is your problem here?
Original	who the f**k are you gona call when that happens.
Paraphrases	Who are you gonna call when that happens?
Original	some idiots no longer believe in national sovereignty.
Paraphrases	Some people no longer believe in national sovereignty
Original	i was f**kin bored as s**t
Paraphrases	I was bored

Table 7.5: Examples of detoxified sentences from the collected English ParaDetox.

Source Dataset	Input Samples	Unique Inputs Paraphrased	Paraphrases per Inputs	Paraphrases Total	Cost per 1,000 inputs	Cost per unique sample
Jigsaw	7,000	3,054	1.34	4,082	$36.65	$0.08
Reddit	7,000	4,947	1.75	8,681	$47.77	$0.06
Twitter	7,000	4,609	1.55	7,674	$42.30	$0.06
Total	21,000	12,610	1.62	20,437	$41.18	$0.07

Table 7.6: Statistics of the crowdsourcing experiments and final version of English ParaDetox dataset.

($0.08 per sample). Figure 7-6 shows that the number of untransferable sentences in the Jigsaw dataset is larger than that of other corpora.

Out of all crowdsourced paraphrases, only a small part was of high quality. We plot the percentage of paraphrases which were filtered out by content and toxicity checks in Figure 7-7. It also corroborates the difficulty of the Jigsaw dataset. While

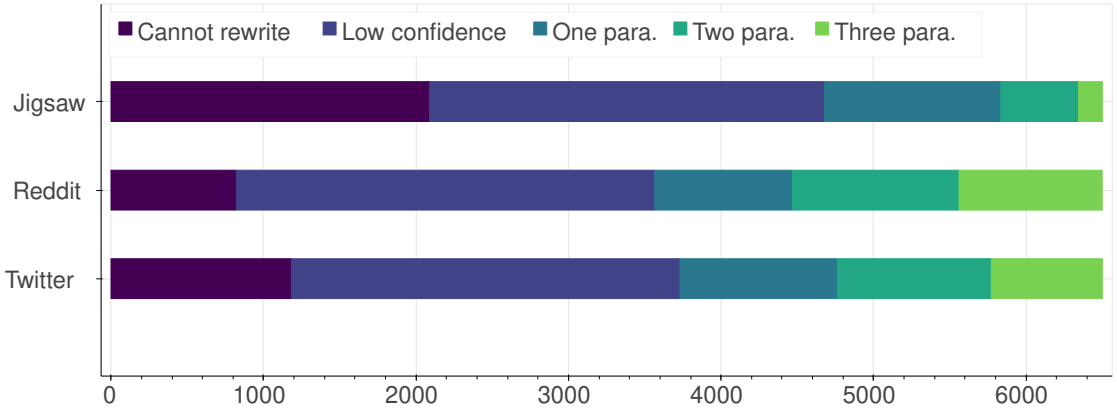


Figure 7-6: Number of paraphrases per input.

the overall number of generated paraphrases was slightly higher for it, much more of them were discarded.

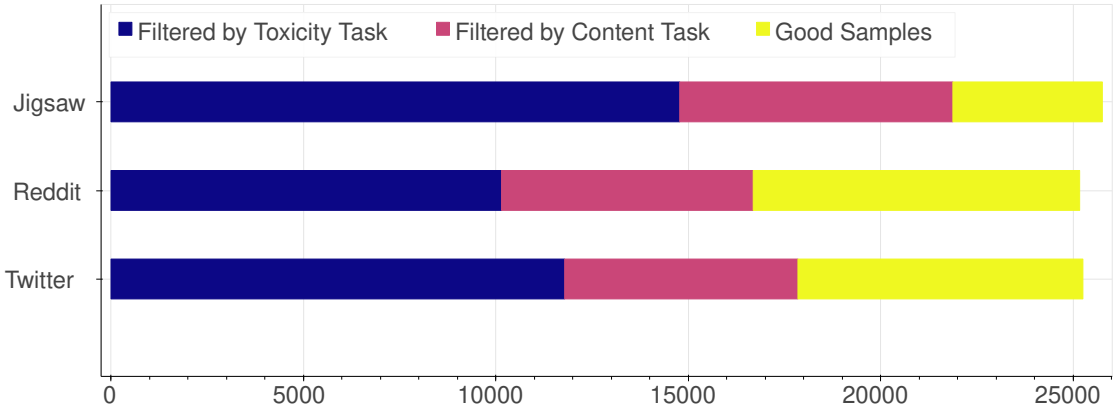


Figure 7-7: Data filtering output.

7.6.2 Analysis of Edits

Although we did not give any special instructions to workers about editing, they often followed the minimal editing principle, making 1.36 changes per sentence on average. A change is the deletion, insertion, or rewriting of a word or multiple adjacent words. Many of the changes are supposedly deletions because the average sentence length drops from 12.1 to 10.4 words after editing.

The nature of editing differs for the three datasets. We compute the percentage of edits which consisted of removing the most common swear words or replacing them with neutral words. We first define the differences between the original and

	Swear words		Other phrases		
Dataset	Del	Rep	Del	Rep	Ins
Jigsaw	2.3%	0.6%	30%	60%	6.8%
Reddit	19%	9.1%	26%	41%	5.7%
Twitter	15%	7.1%	23%	47%	8.2%
ParaNMT	1.6%	1.2%	19%	64%	14%

Table 7.7: Percentage of common swear words (f**k, s**t, a** and their common variants) and other words **Del**eted, **Rep**laced, or **Ins**erted by crowd workers.

transformed string with the `difflib` Python library and then compute the percentage of differences that consist of editing swear words and other (non-offensive) words. We use a small manually compiled list of swear words which includes words *f**k*, *s**t*, *a***, *b***h*, *d**n* and their variants. Table 7.7 shows that the deletion or replacements of the most common swearing constituted a large part of all edits for Reddit and Twitter datasets (22% and 30%), while for Jigsaw it was only 3%.

Another surprisingly common type of editing is the normalization of sentences. The users often fixed casing, punctuation, typos (e.g. *dont* → *don't*, *there's* → *there is*). They also tended to replace colloquial phrases with more formal and standard language. Finally, some users overcorrected the sentences. For example, they replaced neutral words such as *dead*, *murder*, *penis* with euphemisms. This tendency indicates that workers consider any sensitive topic to be inappropriate content and try to avoid it as much as possible.

7.7 Russian ParaDetox

After the English dataset collection, we extended the interface of tasks and instruction to the Russian language.

7.7.1 Pipeline Adaptation

As the original crowdsourcing tasks were designed for the English language, we had to adapt the data collection pipeline to the Russian language. Here are the steps that we completed to make the dataset collection pipeline transferred to a new language:

1. We found already labeled *non-parallel* dataset for toxicity classification. For this step we used a **RuToxic** dataset – a combination of two dataset from Kaggle competitions [Kaggle, 2019, 2020] that covers toxic data from Odnoklassniki[4] and Pikabu[5].

2. We trained Russian toxicity classifier[6] that helps us in preprocessing step to select indeed toxic sample to be an input for the pipeline.

3. We translated the tasks interface. The Russian versions of tasks are illustrated in Figure B-1.

4. In the requirements to crowdworkers, we changed the language requirement – we showed the task only that crowdworkers that are from Russian-speaking regions.

5. From RuToxic dataset we created new training and exam tasks that are required to control quality of labeling process. Additionally, we used **RuTwitter** dataset[7] to gain toxic input samples. Detailed instruction with training examples for each task for Russian language can be found in Appendix B.3.

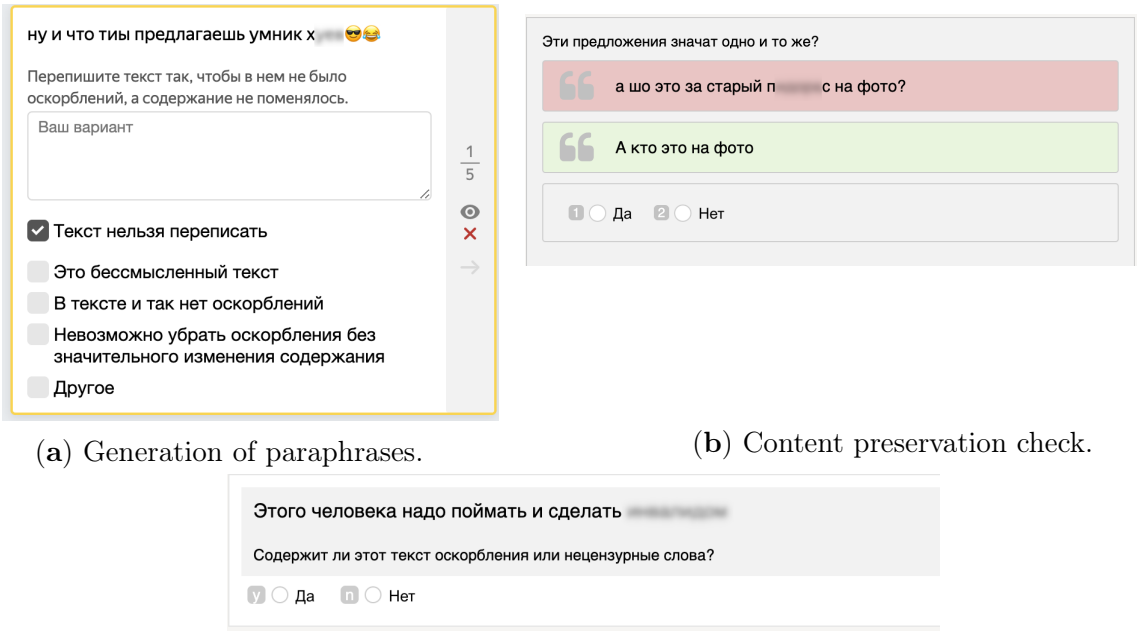
The other setting for the quality control remained the same. As a result, those were the only steps that allowed us already to reuse the pipeline for the Russian language.

7.7.2 Data Analysis

As a result, we collected Russian ParaDetox – a parallel detoxification dataset with 1–3 paraphrases for 8,500 toxic sentences. The whole dataset is available online.[8]

The examples from dataset are shown in Table 7.8. We provide additional examples of gained samples in Appendix B.4.2. The statistics of the paraphrases written by crowd workers are presented in Table 7.9.

[4] https://www.kaggle.com/blackmoon/russian-language-toxic-comments
[5] https://www.kaggle.com/alexandersemiletov/toxic-russian-comments
[6] https://huggingface.co/SkolkovoInstitute/russian_toxicity_classifier
[7] http://study.mokoron.com
[8] https://github.com/skoltech-nlp/russe_detox_2022



(**a**) Generation of paraphrases.

(**b**) Content preservation check.

(**c**) Toxicity classification.

Figure 7-8: Original Russian interfaces in Yandex Toloka platform for labeling.

We selected 20,000 toxic sentences from the RuToxic dataset and 10,000 toxic samples from the RuTwitter dataset. The overlap for paraphrasing task was 3 annotators. We get paraphrases for 8,500 toxic sentences (on average 1.83 paraphrases per sentence), 15,560 paraphrases total. Running 1,000 input sentences through the pipeline costs $31.77, and the cost of one output sample is $0.07. The overall cost of the dataset is $680.

In comparison to the English version, we got smaller amount of samples while we got more sentences as the input. A significant part of Russian input sentences were filtered at Task 1 as non-detoxified ones. The original toxic sentence contained quite a lot of rude phrases that did not have any specific content.

7.8 The Pipeline Credibility

After datasets collection, we evaluated samples manually if we got in the result valid datasets for the further training of detoxification models. We asked pre-selected randomly 200 pairs from both English and Russian ParaDetox for evaluation. We asked three annotators to do manual markup. All annotators are NLP researchers with

Original	ты мужик о**ел что пишешь *(dude, what the f**k are you writing)*
Paraphrases	Зачем ты такое пишешь *(why are you writing this)* Ты мужчина обалдел что пишешь *(man, are you nuts, what are you writing)*
Original	что за п**ик в судейской коллегии??? *(what a f**got is in the judiciary???)*
Paraphrases	кто в судейской коллегии??? *(who is in the judiciary???)* Что за человек в судейской коллегии? *(what a man is in the judiciary?)*
Original	дэтих с**лочей надо расстрелять за жестокое обращение с животным! *(these b**tards should be shot for cruelty to animals)*
Paraphrases	Этих людей нужно наказывать за жестокое обращение с животными! *(these people should be punished for cruelty to animals)*
Original	на х** такое выкладывать, зто и дети будут смотреть д*****бы *(what the h**l do you need to post this, the kids will watch it, motherf****s)*
Paraphrases	Зачем такое выкладывать, это и дети будут смотреть *(Why do you need to post this, the kids will watch it)*
Original	как ур*д такую х**ню пишет??? *(what kind of b*stard writes such bulls**t???)*
Paraphrases	кто такую ерунду пишет *(who writes such a nonsense)*

Table 7.8: Examples of detoxified sentences from the collected Russian ParaDetox.

Source Dataset	Input Samples	Unique Inputs Paraphrased	Paraphrases per Inputs	Paraphrases Total	Cost per 1,000 inputs	Cost per unique sample
RuToxic	20,000	5,340	1.53	8,180	$31.30	$0.06
RuTwitter	10,000	3,160	2,33	7,380	$32.25	$0.07
Total	30,000	8,500	1.83	15,560	$31.77	$0.07

Table 7.9: Statistics of the crowdsourcing experiments and final version of Russian ParaDetox dataset.

a good command of English. We asked annotators to evaluate each pair of `toxic` ↔ `non-toxic` texts if it is valid pair for the dataset or not. For both languages, we got the result that the amount of non-valid pairs is $\leq 10\%$. The inter-annotator agreement (Kripperndorff's α) reaches 0.8. That allows us to state that the proposed parallel dataset collection pipeline is credible for such dataset collection and scalable to different languages.

7.9 The Pipeline Scalability

The Yandex.Toloka platform has an interface in English and workers from a large number of countries. Workers can be filtered by their location and asked to pass built-in language tests (available for many languages) to ensure the knowledge of a particular language. This enables the use of Toloka for the creation of NLP resources in many languages.

In our work, crowd workers manually rephrase sentences from non-parallel datasets. The pipeline does not require any specific data format and can be applied to any text. The only prerequisites are to define the source and target styles and to formulate the task of transferring between them. Thus, we believe that the pipeline is suitable for creating parallel datasets for any other style transfer tasks, at least those which have non-parallel datasets and clear definitions of style (positive ↔ negative, complex ↔ simple, impolite ↔ polite, etc.).

We should admit that our pipeline suggests the availability of (non-parallel) datasets in the chosen styles or at least publicly available sources of such data (e.g. social networks, question answering platforms). However, this is also a prerequisite for any style transfer model trained on non-parallel data. Therefore, any work on style transfer suggests that there exists enough data in the chosen style pair and language. This should not be considered a specific limitation of the pipeline.

7.10 Summary

In this chapter, we presented a new pipeline for parallel detoxification dataset collection. We overcame the issues of previous parallel text style transfer datasets collection – the quality of samples was confirmed with a manual check by experts. We replaced manual checks by experts with crowdsource setup. Current crowdsourcing platforms allow control of the quality of crow workers' performance and scale the annotation to any language and input data size. We reused this advantage by presenting a new totally automated pipeline for parallel detoxification dataset collection. We provided the detailed description of Paradetox collection algorithm as well as each task setup.

After the dataset collection pipeline design, we applied it to two languages presenting English and Russian Paradetox. Both datasets are available for public use. The transfer of the presented data collection pipeline to another language crowdsourcing shows that it can be scaled to any language. Moreover, theoretically, it can be reused for any other type of text style transfer task with the only condition to have an already available dataset for input text sampling and creating training and examination tasks.

Also, we analyzed the edits that did crowd workers to detoxify texts. The statistic showed that while several samples can be detoxified with just the elimination of rude words, a significant part of toxic texts should be rephrased. That confirms at the dataset level the necessity of the development of not only point-wise editing detoxification methods but `seq2seq` text generation methods that will be described in the next chapter.

8

Detoxification Methods

In this section we confirm the Hypothesis 4 stated in previous Chapter 7. Firstly, we present the Conditional BERT Model (condBERT) – a new method for unsupervised text style transfer. Then we present EN-Detox and RU-Detox – monolingual detoxification models trained on the parallel detoxification corpora presented in Chapter 7. We describe evaluation setups, baselines used for comparison, and analysis of the results for the English and Russian detoxification tasks separately. For the Russian language, such kind of study of detoxification task is performed for the first time. In addition, we test proposed approaches for multilingual and cross-lingual setups.

8.1 condBERT: Conditional BERT Model for TST

BERT [Devlin et al., 2019] has been trained on the task of filling in gaps ("masked LM"), we can use it to insert non-toxic words instead of toxic ones. This approach has been suggested by [Wu et al., 2019a] as a method of data augmentation. The authors identify words belonging to the source style, replace them with the `[MASK]` token, and the BERT model then inserts new words of the desired style in the designated places. To push BERT towards the needed style, the authors fine-tune BERT on a style-labeled dataset by replacing segmentation embeddings of the original BERT with trainable style embeddings.

We perform some changes to this model to adapt it for the detoxification task. While in the original conditional BERT model the words are masked randomly, we

select the words associated with toxicity. This can be done in different ways, e.g. by training a word-level toxicity classifier or manually creating a vocabulary of rude and toxic words. We use a method that does not require any additional data or human effort. We train a logistic bag-of-words toxicity classifier. This is a logistic regression model which classifies sentences as toxic or neutral and uses their words as features. As a byproduct of the training process, each feature (word) yields a weight that roughly corresponds to its importance for classification. The words with the highest weights are usually toxic. We use the normalized weights from the classifier as the toxicity score. The overview of CondBERT is shown in Figure 8-1.

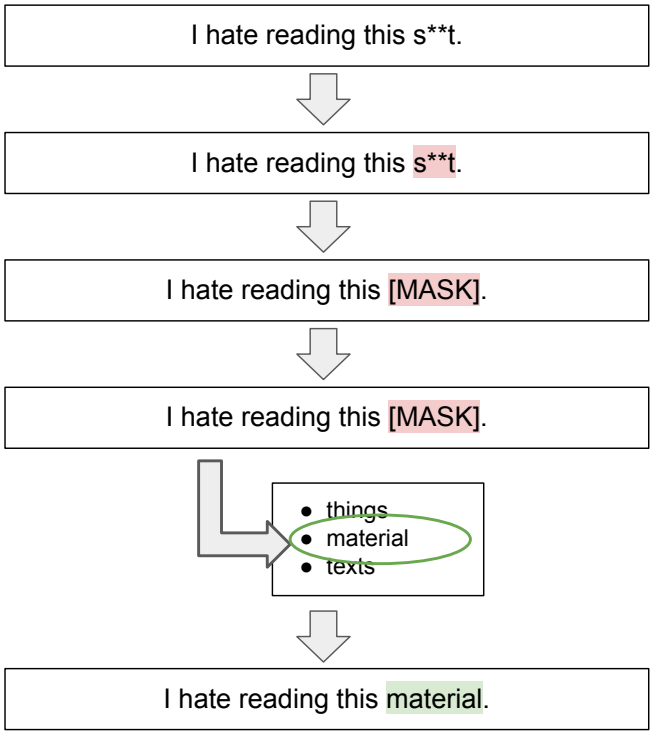


Figure 8-1: Visualization of the idea behind condBERT for unsupervised TST.

For each word in a sentence, we compute the toxicity score and then define toxic words as the words with a score above a threshold

$$t = max(t_{min}, max(s_1, s_2, ..., s_n)/2), \tag{8.1}$$

where $s_1, s_2, ..., s_n$ are scores of all words in a sentence and $t_{min} = 0.2$ is a minimum toxicity score. This adaptive threshold allows balancing the percentage

of toxic words in a sentence so that we avoid cases when too many or no words are marked as toxic.

To preserve the meaning of the replaced word, we employ the content preservation heuristics suggested by [Arefyev et al., 2020]:

1. Preserve the original tokens instead of masking them before the replacement;
2. Rerank the replacement words suggested by BERT by the similarity of their embedding with the embedding of the original word.

Despite using class-specific sentence embeddings, conditional BERT often predicts toxic words, apparently paying more attention to the context than to the embeddings of the desired class. To force the model to generate non-toxic words we calculate the toxicity of each token in BERT vocabulary and penalize the predicted probabilities of tokens with positive toxicities.

Finally, we enable BERT to replace a single `[MASK]` token with multiple tokens. We generate each next token progressively by beam search and score each multi-token sequence by the harmonic mean of the probabilities of its tokens.

8.2 Evaluation of Text Style Transfer

Here we describe in details two strategies of the evaluation of the text style transfer models – automatic and manual evaluation.

8.2.1 Automatic Evaluation

The goals of a style transfer model are to (i) change the text style, (ii) preserve the content, and (iii) yield a grammatical sentence. Thus, to evaluate its performance, we need to take into account all three parameters. The majority of works on style transfer evaluate each of these three parameters with an individual metric. However, Pang and Gimpel [2019] points out that these three parameters are usually inversely correlated, so they need to be combined to find the balance. Our evaluation setup (individual metrics and the joint metric that combines them) follows this principle.

Corresponding to the definition of the Text Style Transfer (TST) task given in Section 6.2.2, we evaluated all detoxification models by three main parameters:

- *Style transfer accuracy* ($\mathbf{STA}_a$): percentage of non-toxic outputs identified by a style classifier. In our case, we train for each language corresponding toxicity classifier.

- *Content preservation* ($\mathbf{SIM}_a$): measurement of the extent to which the content of the original text is preserved. For both languages, we used cosine similarity between corresponding text embeddings of original text and the model's output.

- *Fluency* ($\mathbf{FL}_a$): percentage of fluent sentences in the output. Although fluency is usually evaluated as perplexity, we follow [Krishna et al., 2020] and use a language acceptability classifier.

The aforementioned metrics must be properly combined to get one *Joint* metric to evaluate Textual Style Transfer and rank models. We follow [Krishna et al., 2020] and calculate **J** as an average of products of sentence-level *style transfer accuracy*, and *content preservation*, and *fluency*:

$$\mathbf{J} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{STA}(x_i) \cdot \mathbf{SIM}(x_i) \cdot \mathbf{FL}(x_i) \tag{8.2}$$

where the scores $\mathbf{STA}(x_i)$, $\mathbf{SIM}(x_i)$, $\mathbf{FL}(x_i) \in \{0, 1\}$ meaning the belonging to the corresponding class. Thus, if the sentence belongs to the incorrect class by one of the parameters – either toxic, or dissimilar by content, or non-fluent – we count this sentence as totally inappropriate and the $\mathbf{J}(x_i)$ score for it will be 0. The overall **J** score shows the percentage of sentences which are appropriate by all three parameters simultaneously.

In addition, as we have human references of non-toxic sentences in the corpus, we evaluate the similarity of the model's outputs with them. We used either BLEU or ChrF measure for the corresponding language.

8.2.2 Manual Evaluation

As there is still no best practice for automatic evaluation of the Natural Language Generation (NLG) model [van der Lee et al., 2019], moreover, there is no common practice in how text style transfer models should be evaluated, we provide an additional manual evaluation of proposed methods. Annotators, as in automatic setup, evaluate models by the same three parameters.

Toxicity ($\mathbf{STA}_m$) The toxicity level is defined as:

- **non-toxic** (1) — the sentence does not contain any aggression or offence. However, we allow covert aggression and sarcasm. Note also that toxicity should not be mixed with the lack of formality. Even if a sentence is extremely informal, it is non-toxic unless it attacks someone.
- **toxic** (0) — the sentence contains open aggression and/or swear words (this also applies to senseless sentences).

Content ($\mathbf{SIM}_m$) In terms of content, sentences should be classified as:

- **matching** (1) — the output sentence fully preserves the content of the input sentence. Here, we allow some change of sense which is inevitable during detoxification (e.g. replacement with overly general synonyms: *idiot* becomes *person* or *individual*). It should also be noted that content and toxicity dimensions are independent, so if the output sentence is toxic, it can still be good in terms of content.
- **different** (0) — the sense of the transferred sentence is different from the input. Here, the sense should not be confused with the word overlap. The sentence is different from its original version if its main intent has changed, (cf. *I want to go out* and *I want to sleep*). The partial loss or change of sense is also considered a mismatch (cf. *I want to eat and sleep* and *I want to eat*). Finally, when the transferred sentence is senseless, it should also be considered *different*.

Fluency ($\mathbf{FL}_m$) The fluency evaluation is different from the other metrics. We evaluate it along a ternary scale with the following values:

- **fluent** (1) — sentences with no mistakes, except punctuation and capitalisation errors.
- **partially fluent** (0.5) — sentences which have orthographic and grammatical mistakes, non-standard spellings. However, the sentence should be fully intelligible.
- **non-fluent** (0) — sentences which are difficult or impossible to understand.

However, since all the input sentences are user-generated, they are not guaranteed to be fluent in terms of this scale. People often make mistakes, and typos and use non-standard spelling variants. We cannot require that a detoxification model fixes them. Therefore, we consider the output of a model fluent if the model did not make it less fluent than the original sentence. Thus, we evaluate both the input and the output sentences and define the final fluency score as **fluent** (1) if the fluency score of the output is greater or equal to that of the input, and **non-fluent** (0) otherwise.

Joint Score ($\mathbf{J}_m$) Finally, we aggregate the three metrics in the same Joint score as it was done for automatic evaluation.

8.3 EN-Detox

We use English version of `ParaDetox` to train English `seq2seq` detoxification model.

8.3.1 Supervised Method

We fine-tune a Transformer-based generation model BART [Lewis et al., 2020][1] on our data. We test BART trained on the following datasets:

- **ParaDetox** – our full crowdsourced dataset.

[1] We use model https://huggingface.co/facebook/bart-base

- **ParaDetox-unique** – a subset of ParaDetox where each toxic sentence has only one paraphrase (selected randomly).
- **ParaDetox-1000** – 1,000 samples from the crowdsourced dataset (distributed evenly across data sources, each toxic sample has multiple non-toxic variants).

We train BART for 10,000 epochs with the learning rate of $3e^{-5}$ and the number of gradient accumulation steps set to 1. The other parameters are set to their default values.

8.3.2 Baselines

From baseline models described in Section 6.3 we compare proposed solution with following methods:

Duplicate (baseline) – This is a trivial baseline which consists in leaving the input text intact. It provides a lower threshold for models.

Delete (baseline) – Delete is an unsupervised method that eliminates toxic words based on a predefined toxic words vocabulary. The idea is often used on television and other media: rude words are bleeped out or hidden with special characters (usually an asterisk).

BART-zero-shot (baseline) – BART model with no additional training.

Mask&Infill [Wu et al., 2019b] – BERT-based pointwise editing model.

Delete-Retrieve-Generate Models [Li et al., 2018b]: **DRG-Template** (replacement of toxic words with similar neutral words) and **DRG-Retrieve** (retrieval of non-toxic sentences with the similar sense) varieties.

DLSM [He et al., 2020] encoder-decoder model that uses amortised variational inference.

SST [Lee, 2020] – encoder-decoder model with the cross-entropy of a pretrained style classifier as an additional discriminative loss.

ParaGeDi [Dale et al., 2021] – a model which enhances a paraphraser with style-informed LMs which re-weigh its output.

CondBERT – proposed BERT-based model with extra style and content control. For English language, we used BERT base model.[2]

8.3.3 Evaluation Setup

Train/Test split We separate the English ParaDetox dataset into training and test parts (11,939 and 671 sentence pairs, respectively). The test sentences have one reference per sentence. We manually validate the test set to exclude the appearance of non-detoxifiable sentences or sentences which stayed toxic after rewriting (we need to verify that since the corpus was generated via crowdsourcing only). We use the test set neither for training nor for parameter selection of the models.

Automatic Evaluation For automatic evaluation we used the following models:

- *Style transfer accuracy* (**STA**$_a$) is calculated with a style classifier - RoBERTa-based Liu et al. [2019a] model trained on the union of three Jigsaw datasets Jigsaw [2018]. The sentence is considered toxic when the classifier confidence is above 0.8. The classifier reaches the AUC-ROC of 0.98 and F_1-score of 0.76.
- *Content preservation* (**SIM**$_a$) – cosine similarity between the embeddings of the original text and the output computed with the model of [Wieting et al., 2019]. This model is trained on paraphrase pairs extracted from ParaNMT [Wieting and Gimpel, 2018] corpus. The model's training objective is to yield embeddings such that the similarity of embeddings of paraphrases is higher than the similarity between sentences that are not paraphrases.

[2] https://huggingface.co/bert-base-uncased

- *Fluency* (**FL**$_a$) – percentage of fluent sentences identified by a RoBERTa-based classifier of linguistic acceptability trained on the CoLA dataset [Warstadt et al., 2019].

The comparison of models' outputs with human references is done by **BLEU** metric.

Manual Evaluation For manual evaluation, we randomly select 200 sentences from the test set and ask assessors to evaluate them along the same three parameters: style accuracy (**STA**$_m$), content preservation (**SIM**$_m$), and fluency (**FL**$_m$). All parameters can take values of 1 (good) and 0 (bad). We also report the joint metric **J**$_m$ which is the percentage of sentences whose **STA**$_m$, **SIM**$_m$, and **FL**$_m$ are 1.

The evaluation was conducted by 6 NLP researchers with a good command of English. Each sample was evaluated by 3 assessors. The inter-annotator agreement (Krippendorff's α) reaches 0.64 (**STA**$_m$), 0.67 (**SIM**$_m$), and 0.68 (**FL**$_m$).

8.3.4 Results

Automatic Evaluation Table 8.1 shows the automatic scores of all tested models. Our BART models trained on ParaDetox outperform other systems in terms of BLEU and J. While BART-zero-shot achieves the highest SIM$_a$ score by mostly just duplicating the input text, it totally fails because of this in STA$_a$. The much lower scores of BART-zero-shot confirm that this success is due to fine-tuning and not the innate ability of BART. The majority of unsupervised SOTA approaches are not only worse than BART but also perform below the "change nothing" baseline. The closest competitor of our models is the Delete model. This can be explained by the fact that crowd workers often only remove or replaced swear words which is what the Delete model does.

When comparing models trained on supervised data, we can see that BART does not benefit from multiple detoxifications per sentence, its performance is the same when trained on ParaDetox and ParaDetox-unique.

Table 8.3 shows examples of different models output. Delete performs deterministic operations which can return disfluent text. ParaGeDi generates sentences from scratch, which sometimes results in a distorted sense. Our proposed condBERT

	BLEU	STA_a	SIM_a	FL_a	J
Human reference	100.0	0.96	0.77	0.88	0.66
Baselines and SOTA (unsupervised)					
Delete	61.24	0.81	0.93	0.64	0.46
Duplicate	53.86	0.02	1.0	0.91	0.02
DRG-Template	53.86	0.90	0.82	0.69	0.51
BART-zero-shot	53.64	0.01	**0.99**	0.92	0.01
Mask&Infill	52.47	0.91	0.82	0.63	0.48
CondBERT	42.45	**0.98**	0.77	0.82	0.62
SST	30.20	0.86	0.57	0.19	0.10
ParaGeDi	25.39	**0.99**	0.71	0.88	0.62
DLSM	21.13	0.76	0.76	0.52	0.25
DRG-Retrieve	4.74	0.97	0.36	0.86	0.31
BART on parallel data (supervised) – *our models*					
ParaDetox	**64.53**	0.89	0.86	0.89	**0.68**
ParaDetox-unique	**64.58**	0.87	0.87	0.88	0.65
ParaDetox-1000	63.26	0.83	0.86	0.90	0.62

Table 8.1: Automatic evaluation of English detoxification models. Numbers **in bold** indicate the best results. Rows in gray indicate the baselines.

outperforms the majority of baselines achieving the highest J_a score. However, condBERT has to insert something instead of a toxic word, which is not always a good strategy. BART trained on parallel data is usually free of these drawbacks. More examples of outputs are available in Appendix B.5.1.

	STA_m	SIM_m	FL_m	J_m
Delete	0.785	0.445	0.365	0.21
CondBERT	**0.935**	0.250	0.615	0.15
ParaGeDi	**0.930**	0.415	0.870	0.37
BART-ParaDetox	0.830	**0.925**	**0.960**	**0.76**

Table 8.2: Manual evaluation of English detoxification models. Numbers **in bold** indicate the best results (with the statistical significance $\alpha = 0.01$).

Manual Evaluation Manual evaluation (Table 8.2) confirms the usefulness of parallel data. condBERT shows quite good results in terms of STA_m, but loses both Delete and ParaGeDi by J_m. As a model not always generates suitable substitution

for the masked token, it can result in a disfluent or non-relative to the content text. BARTs trained on parallel data outperform other competitors, even if the size of this data is small. However, manual and automatic evaluations do not always match. Here, the well-performing Delete model gets the lowest score.

Overall, assessors agree with automatic metrics only in terms of fluency, their Spearman correlation r is 0.89. The manual style accuracy and content preservation are only moderately correlated with their automatic counterparts leaving space for further improvements. J_a and J_m almost do not correlate. Besides that, BLEU correlates only with content preservation score and is moderately inversely correlated with the style accuracy. Thus, BLEU measures only the degree of content preservation and cannot replace other metrics.

Original	economies of venezuela, iraq, etc still s**t.
Delete	economies of venezuela , iraq, etc still .
CondBERT	economies of venezuela , iraq , etc still exist today.
ParaGeDi	economies of venezuela, iraq, etc still intact.
BART-ParaDetox	**economies of venezuela, iraq etc are still bad.**
Original	f**k you, i wont do what you tell me.
Delete	you, i wont do what you tell me.
CondBERT	unless i tell you, i wont do what you tell me.
ParaGeDi	Fick, I'll do what you say.
BART-ParaDetox	**I won't do what you tell me.**
Original	your types of examples are *diotic.
Delete	your types of examples are.
CondBERT	your types of examples are very interesting.
ParaGeDi	Your types of examples are weird.
BART-ParaDetox	**Your types of examples are not good**.

Table 8.3: Examples of English detoxifications by different models. Bad answers are shown in red, the best answers **in bold**.

8.4 RU-Detox

We use the Russian version of `ParaDetox` to develop the Russian detoxification model. In comparison to English, there was no research conducted before in the

field of detoxification for the Russian language. The study presented in this work is the first of its kind in general.

8.4.1 Supervised Method

As now we have a parallel corpus available for the Russian detoxification task, we can address the problem as a seq2seq generation task. We experimented with several available Russian language Transformer-based models:

1. **ruGPT-3** the Russian version of GPT-2 [Radford et al., 2019], we test **small**, **medium**, and **large** versions o it;
2. **ruT5** the Russian version of T5 [Raffel et al., 2020], we test **base** and **large** versions of it.

8.4.2 Baselines

Duplicate (baseline) – This is a trivial baseline which consists in leaving the input text intact. It provides a lower threshold for models.

Delete (baseline) – Delete is an unsupervised method that eliminates toxic words based on a predefined toxic words vocabulary. The idea is often used on television and other media: rude words are bleeped out or hidden with special characters (usually an asterisk).

ruGPT-zero-shot (baseline) – ruGPT3 model with no additional training.

RuPrompts [Konodyuk and Tikhonova, 2021] – This baseline is based on the ruPrompts library[3] for fast language model tuning via automatic prompt search. The Continuous Prompt Tuning method consists in training embeddings corresponding to the prompts. Such approach is cheaper than classic fine-tuning of big language models. We tune the prompts for the ruGPT3-large model.[4]

[3] https://github.com/ai-forever/ru-prompts
[4] https://github.com/ai-forever/ru-gpts

condBERT – proposed BERT-based model with extra style and content control. For Russian language, we fine-tuned Conversational RuBERT[5] from DeepPavlov [Kuratov and Arkhipov, 2019].

8.4.3 Evaluation Setup

Train/Test split We separate the Russian ParaDetox dataset into training and test parts (7 798 and 875 sentence pairs, respectively). The test sentences have one reference per sentence. We manually validate the test set to exclude the appearance of non-detoxifiable sentences or sentences which stayed toxic after rewriting (we need to verify that since the corpus was generated via crowdsourcing only). We use the test set neither for training nor for parameter selection of the models.

Automatic Evaluation For automatic evaluation we used following models:

- *Style transfer accuracy* ($\mathbf{STA}_a$) is evaluated with a BERT-based [Devlin et al., 2019] toxicity classifier[6] fine-tuned from RuBERT Conversational. This classifier was additionally trained on Russian Language Toxic Comments dataset collected from `2ch.hk` and Toxic Russian Comments dataset collected from `ok.ru`.

- *Content preservation* ($\mathbf{SIM}_a$) is evaluated as a cosine similarity of LaBSE [Feng et al., 2020] sentence embeddings. The model is slightly different from the original one, only English and Russian embeddings are left.

- *Fluency* ($\mathbf{FL}_a$) is measured with a BERT-based classifier [Devlin et al., 2019] trained to distinguish real texts from corrupted ones. The model was trained on Russian texts and their corrupted (random word replacement, word deletion and insertion, word shuffling etc.) versions.

For the comparison of models' outputs with human references, we choose **ChrF** metric. We choose ChrF [Popović, 2015] over BLEU because it compares character n-grams and is more suitable for languages with rich morphology, such as Russian.

[5] https://huggingface.co/DeepPavlov/rubert-base-cased-conversational
[6] `https://huggingface.co/SkolkovoInstitute/russian_toxicity_classifier`

Manual Evaluation Manual evaluation for the Russian language was done by the described above design but with an automated pipeline via crowdsourcing. More details on how it was done are described in the next Chapter 9.

8.4.4 Results

Automatic Evaluation Table 8.4 presents the results of automatic evaluation of described models. Among the baselines, ruGPT-zero-shot performs the most poorly. It generates just random text which is non-toxic (that explains quite a high STA_a score) but absolutely does not correlate with the input. The delete method achieves the highest SIM_a score as edits the input text locally. The ruPrompts method has the best J_a score among baselines of 0.53. While condBERT has the same level of STA_a, it performs worse in terms of SIM_a and FL_a than ruPrompts.

	ChrF	STA_a	SIM_a	FL_a	J
Human reference	0.77	0.85	0.72	0.78	0.49
Baselines and SOTA (unsupervised)					
Delete	0.53	0.56	**0.89**	0.85	0.41
Duplicate	0.56	0.24	1.0	1.00	0.24
ruGPT-zero-shot	0.05	0.92	0.20	0.11	0.00
CondBERT	0.54	0.81	0.77	0.74	0.47
ruPrompts	0.55	0.81	0.79	0.80	0.53
Models on parallel data (supervised) – *our models*					
ruGPT3-small	0.52	0.72	0.78	0.77	0.43
ruGPT3-medium	0.50	0.78	0.75	0.74	0.43
ruGPT3-large	0.55	0.73	0.75	0.74	0.41
ruT5-base	**0.57**	0.80	0.83	0.84	0.56
ruT5-large	0.55	**0.95**	0.86	**0.97**	**0.78**

Table 8.4: Automatic evaluation of Russian detoxification models. Numbers **in bold** indicate the best results. Rows in gray indicate the baselines.

RuGPT3 models trained on parallel data perform better than simple baselines and slightly worse than more advanced baselines. While SIM_a and FL_a metrics are almost the same, ruGPT3 models show lower STA_a. ruGPT3-large achieves the same level of ChrF measure as ruPrompts models. Finally, our ruT5 models trained on parallel data significantly outperform all baselines by J_a and rich the

best ChrF. The ruT5-large model shows the best J_a score of 0.78 outperforming the best baseline by 0.25. It also achieves the highest STA_a and FL_a among all models. At the same time, ruT5-base has a lower J_a score, but the highest ChrF of 0.57.

Table 8.5 shows the examples of models outputs. We can see that sometimes just the elimination of a rude word can be enough to achieve successful detoxification. However, it can work poorly for other cases. CondBERT for the Russian language fails to find a correct and fluent substitution of a rude word. ruGPT3 models can generate adequate detoxifications but also can add some auxiliary information that was not present in the original sentences. ruT5 model both base and large generate absolutely suitable fluent detoxified paraphrases. For more examples, you can refer to Appendix B.5.2, Table B.11.

Original	твари е***ие, с**а где статья ваша?
Delete	**где статья ваша?**
CondBERT	т е , су где статья ва ?
ruGPT3-small	Где статья вашего?
ruGPT3-medium	Где статья вашей статьи
ruGPT3-large	Люди, где статья ваша
ruT5-base	**Где статья ваша?**
ruT5-large	**Где статья Ваша?**
Original	Тебя это е***ь не должно, п*****га
Delete	Тебя это не должно,
CondBERT	Тебя это е не должно , потому что
ruGPT3-small	Тебя это обижать не должно
ruGPT3-medium	Тебя это должно не волновать
ruGPT3-large	Тебя это должно не беспокоить
ruT5-base	**Тебя это волновать не должно.**
ruT5-large	**Тебя это волновать не должно!**

Table 8.5: Examples of Russian detoxifications by different models. Bad answers are shown in red, the best answers **in bold**.

Manual Evaluation In addition, we evaluate the best baselines and the best seq2seq models manually. Table 8.6 presents the results. Our models trained on a parallel dataset in the manual evaluation as well significantly outperform the baselines. In comparison to automatic evaluation, the best ruT5-base gets the highest J_m score of 0.61.

	STA_m	SIM_m	FL_m	J_m
Delete	0.39	0.71	0.73	0.16
CondBERT	0.43	0.62	0.79	0.17
ruPrompts	**0.80**	0.70	0.87	0.49
ruT5-base	0.79	0.82	**0.92**	**0.61**
ruT5-large	0.73	**0.87**	**0.92**	0.60

Table 8.6: Manual evaluation of Russian detoxification models. Numbers **in bold** indicate the best results (with the statistical significance $\alpha = 0.01$).

As for English evaluation setup, we can observe the difference between automatic and manual evaluations. While assessors almost agree on ranking of systems by each parameter with automatic evaluation, the scale of scores if different. Moreover, the ranking of ruT5 models by J_a differ from automatic one.

8.5 Multilingual and Cross-lingual Setups

After confirmation of hypothesis, that the presence of parallel dataset improves significantly monolingual detoxification, we want to check if monolingual and cross-lingual setups for detoxification are possible.

Multilingual setup In this setup we train models on data containing both English and Russian texts and then compare their performance with baselines trained on these languages solely.

Cross-lingual setup In cross-lingual setup we test the hypothesis that models are able to perform detoxification without explicit fine-tuning on exact language. We fine-tune models on English and Russian separately and then test their performance.

8.5.1 Experimental Setup

Scaling language models to many languages has become an emerging topic of interest recently [Devlin et al., 2019, Tan et al., 2019, Conneau and Lample, 2019, Conneau et al., 2020]. We adopt several multilingual models to textual style transfer in our work.

Baselines For the baselines we use methods that have similar concept and implementation for both languages: i) **Delete**; ii) **CondBERT**.

mT5 mT5 [Xue et al., 2021] is a multilingual version of T5 [Raffel et al., 2020] - a text-to-text transformer model, which was trained on many downstream tasks. mT5 replicates T5 training but now it is trained on more than 100 languages.

mBART mBART [Liu et al., 2020] is a multilingual variation of BART [Lewis et al., 2020] - denoising autoencoder built with a sequence-to-sequence model. mBART is trained on monolingual corpora across many languages. We adopt mBART in sequence-to-sequence detoxification task via fine-tuning on parallel detoxification dataset.

For evaluation metrics we used all described above metrics for English and Russian detoxification in Sections 8.3.3 and 8.4.3 respectively.

8.5.2 Training

There is a variety of versions of large multilingual models available. In this work we use small and base versions of mT5[7][8] and large version of mBART[9].

Multilingual training In multilingual training setup we fine-tune models using both English and Russian data. We use Adam [Kingma and Ba, 2015] optimizer for fine-tuning with different learning rates ranging from $1 \cdot 10^{-3}$ to $5 \cdot 10^{-5}$ with linear learning rate scheduling. We also test different number of warmup steps from 0 to 1000. We equalize Russian and English data for training and use 10000 toxic sentences and their polite paraphrases for multilingual training in total. We train mT5 models for 40 thousand iterations[10] with a batch size of 8. We fine-tune mBART [Liu et al., 2020] for 1000, 3000, 5000 and 10000 iterations with batch size of 8.

[7] https://huggingface.co/google/mt5-base
[8] https://huggingface.co/google/mt5-large
[9] https://huggingface.co/facebook/mbart-large-50-many-to-many-mmt
[10] According to [Xue et al., 2021] mT5 was not fine-tuned on downstream tasks as the original T5 model. Therefore, model requires more fine-tuning iterations for Textual Style Transfer.

Cross-lingual training In cross-lingual training setup we fine-tune models using only one dataset, e.g.: we fine-tune model on English data and check performance on both English and Russian data. Fine-tuning procedure was left the same: 40000 iterations for mT5 models and 1000, 3000, 5000 and 10000 iterations for the mBART.
Back-translation approach to cross-lingual style transfer proved to work substantially better than the zero-shot setup discussed above. Nevertheless, both Google and FSMT did not yield scores comparable to monolingual setup. Besides, surprisingly Google yielded worse results than FSMT.

8.5.3 Results

Table 8.7 shows the best scores of both multilingual and cross-lingual experiments. In a multilingual setup, mBART performs better than baselines and mT5 for both English and Russian. Note that the table shows only the best results of the models. It is also notable that for mT5 increased training size for English data provides better metrics for English while keeping metrics for Russian almost the same. We also depict some of the generated detoxified sentences in the Table B.12 in the part B.5.3 of Appendix.

As for cross-lingual style transfer, the results are negative. None of the models have coped with the task of cross-lingual Textual Style Transfer. That means that models produce the same or almost the same sentences for the language on which they were not fine-tuned so that toxicity is not eliminated. We provide only some scores here in the Table 8.7 for reference.

8.6 Demonstration Systems

Following task motivation in Section 6.1, proposed detoxification models can be quite useful to real-world scenarios. Previous work did not achieve such high performance that they can be deployed for online usage. As our detoxification models for both English and Russian languages achieve the highest scores for automatic and, most importantly, for manual evaluation showing adequate results on test text samples, we implement system demonstration for detoxification and make a showcase for

	STA_a	SIM_a	FL_a	J_a	STA_a	SIM_a	FL_a	J_a
	Russian				**English**			
	Baselines							
Delete	0.532	0.875	0.834	0.364	0.810	0.930	0.640	0.460
condBERT	0.819	0.778	0.744	0.422	**0.980**	0.770	0.820	0.620
	Multilingual Setup							
mT5 base	0.772	0.676	0.795	0.430	0.833	0.826	0.830	0.556
mT5 small	0.745	0.705	0.794	0.428	0.826	0.841	0.763	0.513
mT5 base*	**0.773**	0.676	0.795	0.430	**0.893**	0.787	**0.942**	**0.657**
mBART 1000	0.599	**0.843**	0.867	0.431	0.763	**0.879**	0.879	0.563
mBART 3000	0.686	0.800	**0.872**	**0.484**	0.869	0.848	0.886	0.634
mBART 5000	0.705	0.772	0.857	0.475	0.887	0.836	0.896	0.651
mBART 10000	0.727	0.746	0.835	0.463	0.873	0.829	0.876	0.627
	Cross-lingual Setup							
mT5 base ENG	0.838	0.276	0.506	0.115	0.860	0.834	0.833	0.587
mT5 base RUS	0.676	0.794	0.846	0.454	0.906	0.365	0.696	0.171
mT5 small ENG	0.805	0.225	0.430	0.077	0.844	0.858	0.826	0.591
mT5 small RUS	0.559	0.822	0.817	0.363	0.776	0.521	0.535	0.169
mBART 1000 ENG	0.241	0.965	0.951	0.208	0.777	0.874	0.881	0.571
mBART 1000 RUS	0.610	0.827	0.865	0.435	0.352	0.872	0.911	0.215
mBART 3000 ENG	0.352	0.915	0.910	0.276	0.842	0.856	0.876	0.617
mBART 3000 RUS	0.699	0.778	0.858	0.475	0.547	0.778	0.888	0.299
mBART 5000 ENG	0.900	0.299	0.591	0.160	0.857	0.840	0.873	0.616
mBART 5000 RUS	0.724	0.746	0.827	0.457	0.806	0.484	0.864	0.242
mBART 10000 ENG	0.349	0.892	0.897	0.260	0.857	0.835	0.867	0.605
mBART 10000 RUS	0.718	0.735	0.827	0.448	0.517	0.840	0.903	0.342
	Backtranslation Setup							
mBART 5000 (Google)	0.675	0.669	0.634	0.284	0.678	0.762	0.568	0.284
mBART 5000 (FSMT)	0.737	0.633	0.731	0.348	0.744	0.746	0.893	0.415

Table 8.7: Evaluation of TST models. Numbers in **bold** indicate the best results. ↑ describes the higher the better metric. Results of unsuccessful TST depicted as gray. ENG and RUS depict the data model have been trained on. mT5 base* was trained on all English and Russian data available (datasets were not equalized). The last row depicts the backtranslation workaround for cross-lingual detoxification. We include only the best result for brevity.

real-life industry application.

8.6.1 Online Demonstrations

Firstly, we implement our models as demonstration system in a website[11] (Figure 8-2). The user can select one of the models from the list. There present several baselines as well as the current best models for each language. The user can then send his or her text requests and get the detoxified version of it. The system will highlight, how the text has been changed and how the score of toxicity has changed

[11] https://detoxifier.nlp.zhores.net

as well. Additionally, in some models, the user can adjust the strength of style change parameter with a scroller.

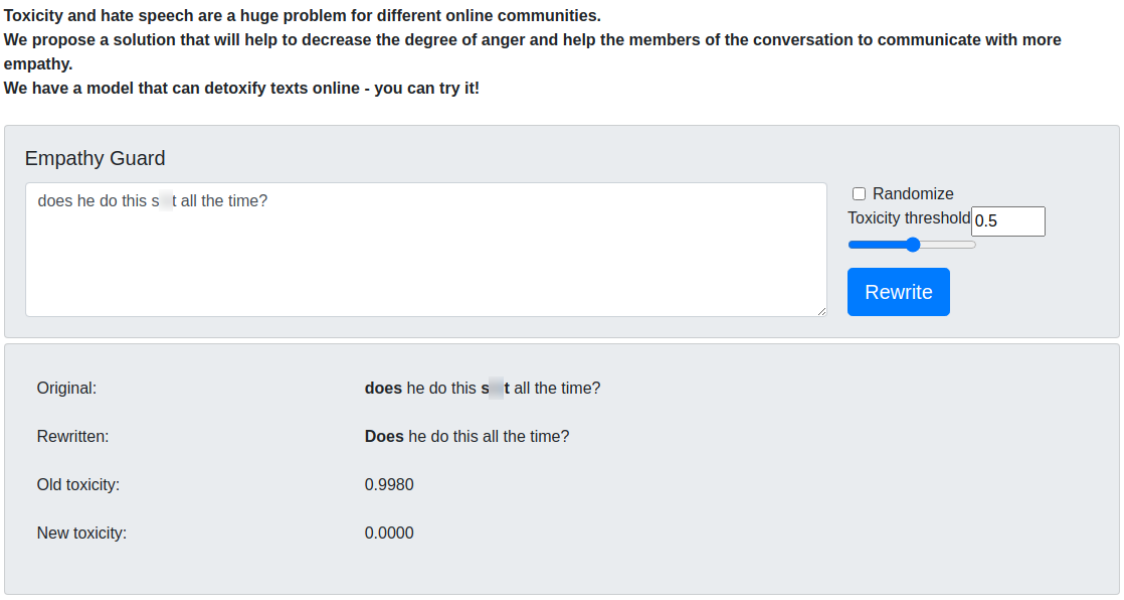


Figure 8-2: Demonstration system in the form of a website of detoxification models. The user can choose a model from the list – both baselines and proposed new models are presented – and then write text request in the corresponding language.

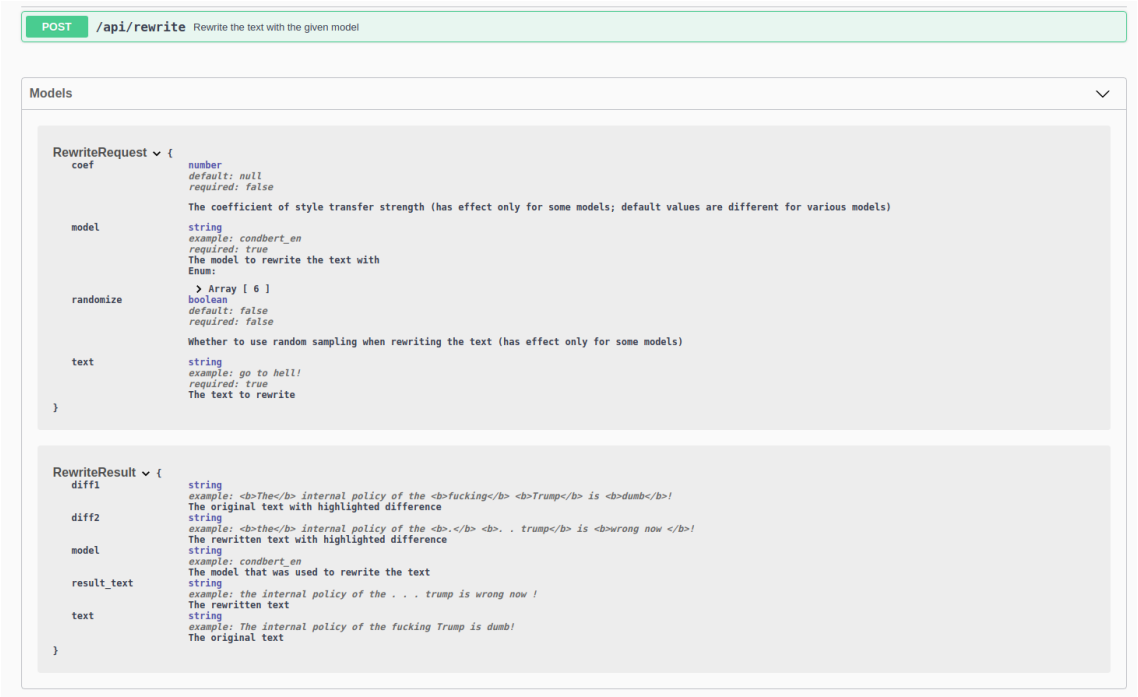


Figure 8-3: For a models we provide API that is available for further integration in various NLP applications.

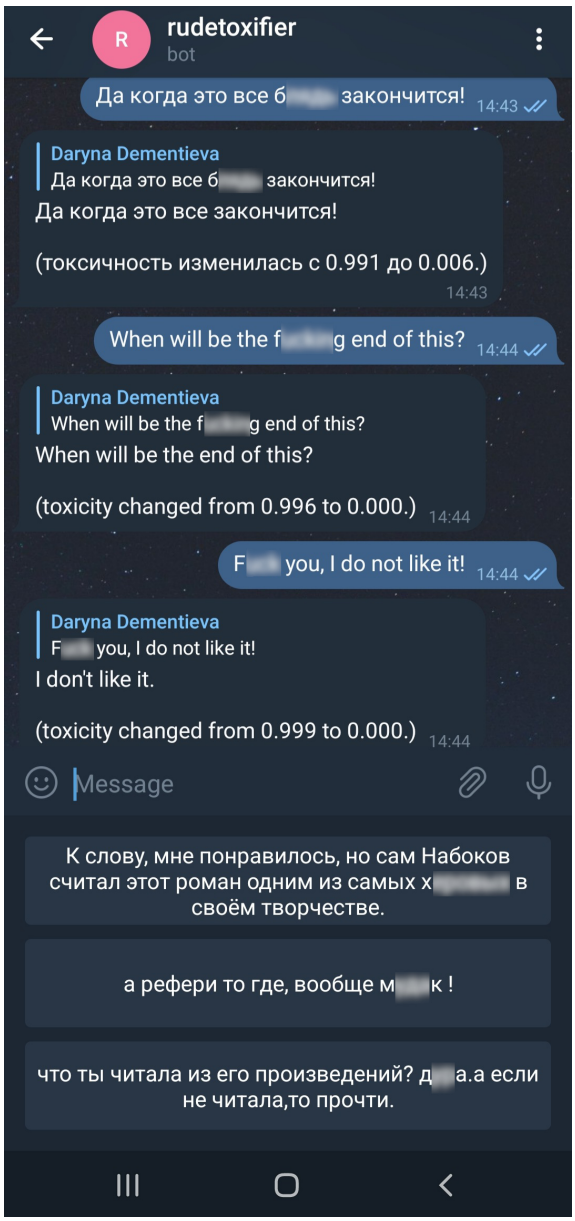


Figure 8-4: Demonstration system in the form of a Telegram bot of detoxification models. The user can write a text just in a language that he/she wants. The system can detect a language and perform detoxification with corresponding SOTA model.

Secondly, we implement the demonstration system with the interface more suitable for mobile devices – via Telegram-bot[12] (Figure 8-4). The users can write their text requests in the language that they prefer without explicit identification of the language. We implement in preprocessing step language detection with the model based on fasttext library [Joulin et al., 2016]. The user can get a detoxified version of the text request and the change text's toxicity score.

8.6.2 Game Industry Showcase

Despite the quite popular opinion that toxic communication is part of the game community, the majority of game players would like toxicity to be reduced according to recent studies. Finnish Refugee Council and SuperCell company launched a challenge to create a way to address toxicity in gaming dialogues.[13] We demonstrate how our proposed technology can be helpful to decrease toxicity in the chats between gamers.

Figure 8-5: A show case how NLP techniques can help to increase empathy in the players' chat.

One of the cases that can occur, is the players can be quite upset about the

[12] https://t.me/rudetoxifierbot
[13] https://www.junction2021.com/challenges/supercell

not successful game. They can start to behave self-destructive writing demotivating messages. To lift the spirits and prevent participants from becoming toxic or aggressive, an integrated NLP system can detect such behavior and suggest to the participants more proactive messages as shown in Figure 8-5.

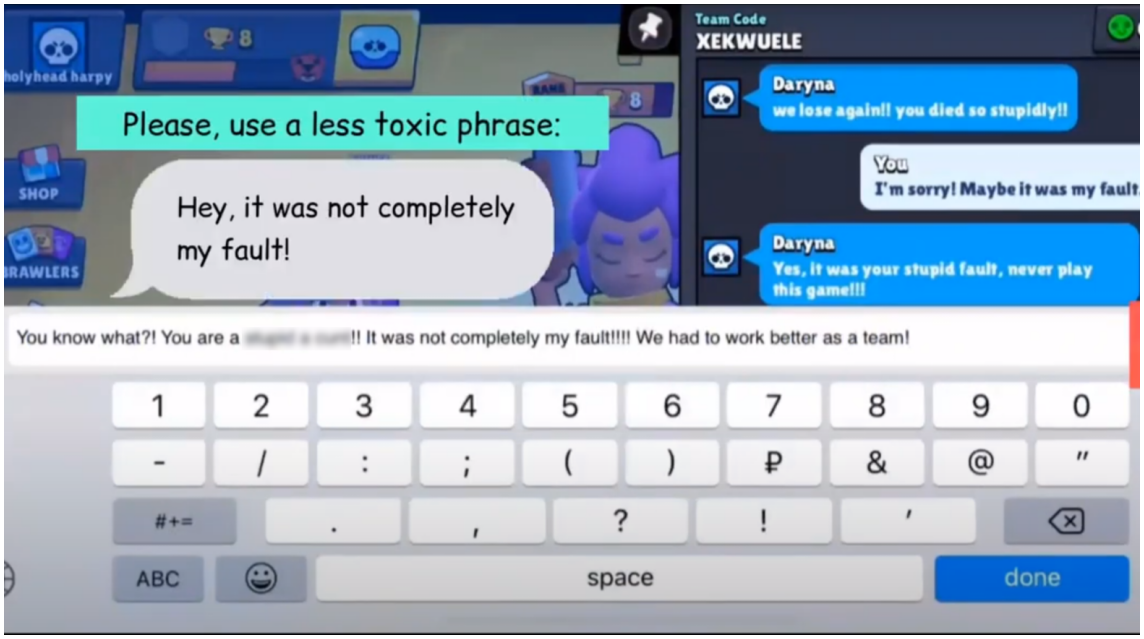


Figure 8-6: A show case how proposed detoxification system can provide recommendation for a user that uses toxic speech.

If the participants of the gaming group start to behave toxic or rude, then the system can detect such toxic speech, warns a user that his or her behavior is toxic, and suggest a less toxic version of the message (Figure 8-6). That can make a user pause, put his or her anger in a text, but then send detoxified version of his or her thoughts.

Not to violate the freedom of speech, detoxification of the user's message is only a recommendation. The user can skip it and send toxic message anyway. However, if the gaming platform has a politics to decrease the amount of toxic speech in conversations, then it can notify users that the too often usage of toxic speech and not returning to neutral tone can lead to some kind of fines (Figure 8-7). In addition, the users tat do not use toxic speech can be awarded with bonuses.

All proposed step can help users to be more aware of their toxic behaviour and motivate them to increase more empathic conversations in the future.

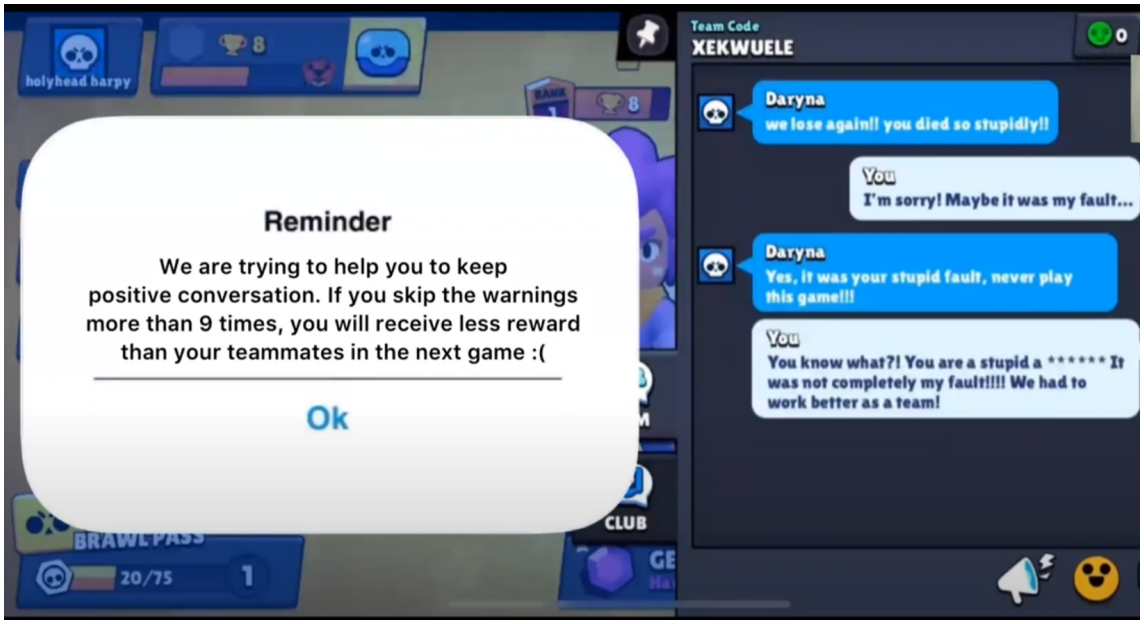


Figure 8-7: A show case how a platform can manage users that refer to toxic behaviour too often.

8.6.3 Speech Detoxification

All described in this Chapter detoxification technologies take as input only text. However, the next step of such technology development can be detoxification of speech. For instance, for discussed above use case for game industry it is even more important to detoxify speech. Commonly, the players talk with each other with voice during the game. Moreover, it is even more usual place for toxic behaviour than text chat.

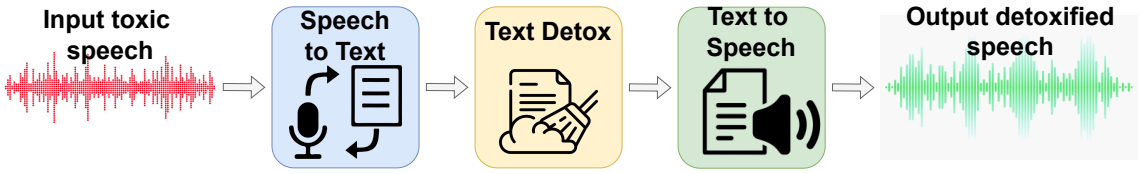


Figure 8-8: The pipeline of speech detoxification based on the already implemented text detoxification technologies.

The possible pipeline for speech detoxification is illustrated in Figure 8-8. Each step in this pipeline is already possible. There are already solutions that restore punctuation and generate text based on input speech track. Such models After that, the task is converted to the text detoxification task. The quite accurate text

detoxification approaches are proposed in this work. The last step is to convert the detoxified text back into speech.

8.7 Summary

Stated in Chapter 7 Hypothesis 4 that the usage of parallel corpus can improve the performance of text detoxification task is confirmed. Firstly, we proposed new unsupervised TST method condBERT that do not require any parallel data for training and do point-wise editing of the input sentence. It showed quite good results in both automatic and manual evaluation outperforming all previous proposed unsupervised TST models and was used as a strong baseline for further comparisons. The model can be significantly improved using the more advanced methods to generate mask substitutions.

Then we trained our seq2seq models on proposed ParaDetox dataset. Out models significantly outperform the baselines in monolingual English and Russian detoxification. Moreover, for Russian language the presented Russian texts detoxification study is the first of its kind. We release online the best models by J_a for both languages presenting EN-Detox[14] and RU-Detox[15].

Finally, we explore the task of multilingual and cross-lingual text detoxification with the help of our parallel data for English and Russian and large multilingual language models. According to the results of the experiments, supervised multilingual training yields decent detoxification models that are able to solve a task of multilingual text detoxification. However, the results of cross-lingual detoxification experiments has a room for improvement. One of the future directions to improve cross-lingual style transfer can be prompt engineering. As LLMs are trained on quite many languages, we can find a trainable with prompts way to inform the model about the idea of detoxification task and how it should be propagated to other languages.

In addition, we show the examples how the proposed systems can be used in real-life applications. We release online system demonstrations in the form of the

[14] https://huggingface.co/SkolkovoInstitute/bart-base-detox
[15] https://huggingface.co/SkolkovoInstitute/ruT5-base-detox

website and Telegram-bot. Also, we suggested a show case of the usage of the detoxification system for game industry. One of the further steps of detoxification technology development is synchronous detoxification of speech. We believe, that proposed detoxification techniques can help to increase empathic user behaviour.

8.8 Ethical Considerations

The research on toxicity raises some ethical issues. In terms of our work, the parallel corpus we created can indeed be used in the reverse direction, i.e. to "toxify" sentences. However, although we did not thoroughly evaluate the quality of such toxification, our intuition is that it would not be high enough to make the corrupted sentences look natural. The reason is that the toxic part of our corpus consists of real toxic sentences fetched on the Internet, whereas their non-toxic counterparts are "translations" performed by crowd workers. We suggest that they obey the common regularities observed for *translationese* (texts manually translated from their original language into a different one): they differ from regular texts in terms of vocabulary [Koppel and Ordan, 2011] and syntax [Lembersky et al., 2011]. The manually detoxified texts are different from the original non-toxic texts written by Internet users from scratch. While they are still recognized by human assessors as plausible sentences, we suggest that a sequence-to-sequence model trained to get translationese as input would not be as successful in transforming real texts (as it was shown for machine translation models [Freitag et al., 2019]).

Thus, although our corpus can be used in the reverse direction, it is not symmetric, which makes it less efficient as training datasets for "toxifiers". However, we should emphasize that these statements are our hypotheses and should be further investigated. Finally, we argue that the risk of using our corpus for toxification is perhaps not game-changing, as simpler approaches based on patterns (e.g. including a set of predefined obscene fragments into neutral texts) can serve the same purpose relatively well.

9

A Study of Human vs Automatic Evaluation

As it can be observed in the previous Chapter 8 for monolingual detoxification, the results of text style transfer can differ between manual and automatic evaluations. In this chapter, we explore this effect more deeply:

- We present **the first of its kind pipeline** for the automated collection of manual assessments for text style transfer tasks.
- We conduct **the first study** of exploration of the connection between the human and automatic evaluation of the detoxification task.

The results of the presented study are quite important for future exploration, evaluation, and ranking of text style transfer models.

The material of this section is based on the results of competition **RUSSE 2022**: *"Russian Text Detoxification Based on Parallel Corpora"* [**Daryna Dementieva** et al., 2022]. The competition, as well as the presented study in this chapter, are dedicated to the Russian language.

9.1 Automatic Evaluation of Style Transfer

In earlier works, reference-based evaluation metrics were considered a holistic evaluation technique [Li et al., 2018b], by analogy with Machine Translation. Even some

recent works [Sudhakar et al., 2019, Zhu et al., 2021] use BLEU or other metrics such as GLEU as the only means of evaluation. Unfortunately, they often cannot control style. Thus, it became obvious that both content and style have to be directly evaluated.

Some works settle for mere evaluation of style and content [Malmi et al., 2020, Zhang et al., 2020b]. However, more often these two metrics are combined by computing their geometric or harmonic mean, as first suggested by [Xu et al., 2018]. This technique is often used to get the joint quality score [Riley et al., 2021, Huang et al., 2021, Lai et al., 2021a,b]. Many (although not all) works also evaluate the fluency of the generated text. This is almost exclusively done via computing the perplexity of text in terms of a language model (e.g. GPT-2 [Radford et al., 2019]). The only alternative used in style transfer works is the use of a classifier of linguistic acceptability [Krishna et al., 2020] trained on the CoLA dataset [Warstadt et al., 2019]. Fluency is sometimes also included in the joint score together with the style and content preservation. Pang and Gimpel [2019] compute it as a document-level geometric mean, and [Krishna et al., 2020] multiply the sentence-level scores. In our work, we use the latter approach.

9.2 Manual Evaluation of Style Transfer

The researchers have come to the conclusion that these automatic metrics cannot provide an objective evaluation. It has become a de-facto standard to enhance automatic evaluation with human evaluation experiments.

There are two main human evaluation scenarios. Outputs of two models can be evaluated side by side, in this case, the authors report the number of wins of each of the models (i.e. the number of cases where a particular model generated a better text) and the number of ties [Zhu et al., 2021, Li et al., 2019, Cheng et al., 2020]. Alternatively, the outputs of different models are evaluated independently. In this case, the assessors evaluate the outputs along three parameters: style, content preservation, and fluency. The parameters are often evaluated in terms of a 1-to-5 scale [Zhou et al., 2020a, Madaan et al., 2020, John et al., 2019, Lee et al., 2021, Ma

et al., 2021]. Sometimes the style is evaluated in terms of a 7-value scale (from -3 to 3), content preservation takes values from 1 to 6 [Chawla and Yang, 2020, Briakou et al., 2021b]. Other scales are also possible. Besides that, the three individual metrics can be evaluated using the side-by-side scenario [Sudhakar et al., 2019, Lin et al., 2020].

9.3 Detoxification Models

In this section we provide the description of all models that are evaluated and considered in the research – both provided baselines and submitted models from participants.

9.3.1 Baselines

We provide four baselines for detoxification task: a trivial Duplicate baseline, a rule-based Delete approach, fine-tuning on the ruT5 model and the continuous prompt tuning approach for ruGPT3 model. Several baselines repeat the baselines for monolingual Russian detoxification study from Section 8.4. We remind here the description of the baselines:

Duplicate – copy of the input.

Delete – deletion of swear words.

RuPrompts The baseline is based on the library ruPrompts. Pre-trained prompts for the baseline is available in huggingface[1].

RuT5 Baseline We used the proposed in this work Ru-Detox model as a baseline for competition.

[1] https://huggingface.co/konodyuk/prompt_rugpt3large_detox_russe

9.3.2 Participants

We briefly describe the models developed by participants. More details about the participating systems can be found in [**Daryna Dementieva** et al., 2022].

Team 1 (ruT5-finetune) Authors approach is based on the ruT5 model[2]. It was fine-tuned on the part of competition train data with a learning rate 1e-5 on 15 epochs. Only the samples with fluency, similarity, and accuracy higher than 0.5 were selected from the train set. The best output is selected from 32 generated samples using beam search. It was decided not to use sampling.

Team 2 (ruGPT3-filter) This team's solution uses a model based on ruGPT3. The authors filtered the dataset released by the organizers with the following heuristics: (i) cosine similarity between the original and transformed sentences ranges from 0.6 to 0.99; (ii) ROUGE-L between the sentences ranges from 0.1 to 0.8; (iii) the transformed sentence length is less or equal to the original sentence length. This dataset was used to fine-tune ruGPT3.

Team 3 (lewis) solution is based on the LEWIS framework [Reid and Zhong, 2021], a coarse-to-fine editor for style transfer that transforms text using Levenshtein edit operation. First, the sequence of coarse-grain Levenshtein edit types (keep, replace, delete or insert) was predicted for each sentence pair. Next, the resulting tags were used to train the conversational RuBERT[3] for the sequence tagging task. The ruT5-base model was trained to fill in the tokens for coarse-grain edit type *replace*.

Team 4 (ruGPT3-XL) trained RuGPT3 XL[4] to generate a non-toxic text on the competition train data. The input is the concatenation of the toxic and non-toxic sentences.

[2] https://huggingface.co/sberbank-ai/ruT5-base
[3] https://huggingface.co/DeepPavlov/rubert-base-cased-conversational
[4] https://huggingface.co/sberbank-ai/rugpt3xl

Team 5 (RoBERTa-replace) solution is based on the RoBERTa-large[5]. The logistic regression model on the FastText vectors trained on the competition data was used as a toxic words classifier. Toxic tokens were substituted by RoBERTa-large model, where the best candidates were chosen by the cosine similarity between the candidate and the toxic token. In case it was not possible to find an acceptable candidate, the toxic word was removed from the sentence.

Team 6 (ruT5-clean) used the ruT5-large model[6] improved by data cleaning. The preprocessing stage consitsts of emoticons and smiley filtering and removing duplicate characters. The Levenshtein Transformer [Susanto et al., 2020] was used as an extra step in preprocessing to clean the ruT5-large model output.

Team 7 (ruT5-large) modified the t5 baseline. RuT5-base was replaced by ruT5-large with beam search used as inference algorithm. 20 candidates were generated for each toxic sentence, the best candidate was selected by the largest J-score metric.

Team 8 (ruT5-preproc) This solution is based on ruT5-base model with additional pre- and postprocessing of the texts. Team finetuned the ruT5-base model on the provided data and used heuristics for text pre/processing.

Team 9 (adversarial) This team devised an adversarial training setup where the training data was enriched with the artificially generated sentences which attained the highest scores of the automatic metrics.

Team 10 (ruPrompts-plus) This team advanced over the ruPrompts baseline. The solution is based on RuGPT3-XL (Generative Pretrained Transformer-3 for Russian) [7] adapted to the task via prompt tuning. Using RuGPT3-XL as a frozen backbone, team trains only a sequence of continuous embeddings inserted before and after an input text.

[5]https://huggingface.co/sberbank-ai/ruRoberta-large
[6]https://huggingface.co/sberbank-ai/ruT5-large
[7]https://huggingface.co/sberbank-ai/rugpt3xl

9.4 Automatic Evaluation

We use the same evaluation setup as for Russian monolingual detoxification described in Section 8.4.3.

Style ($\mathbf{STA}_a$) ruBERT-based classifier trained on Russian Language Toxic Comments dataset collected from `2ch.hk` and Toxic Russian Comments dataset collected from `ok.ru`.

Content ($\mathbf{SIM}_a$) The cosine similarity of embeddings from LaBSE model [Feng et al., 2020] of the source and the transformed sentences.

Fluency ($\mathbf{FL}_a$) The percentage of fluent sentences identified by BERT-based classifier [Devlin et al., 2019] trained to distinguish real texts from corrupted ones.

Joint ($\mathbf{J}_a$) We combine the three metrics at the sentence level by multiplying them. The document-level score is computed as the average of scores for all sentences.

ChrF We provide an additional reference-based metric which follows the Machine Translation evaluation setup.

9.5 Manual Evaluation via Crowdsourcing

For the definition of manual metrics, we use the same setup described for monolingual detoxification in Section 8.2. However, the novelty of this study is that the manual evaluation is not made by only several annotators and then manually aggregated but via a crowdsourcing setup. Such an automated pipeline allows saving time searching for annotators and aggregating results and allows for more accurate markup due to overlap. Each of the three parameters is evaluated in a separate crowdsourcing project. For all the projects, the evaluation was made by only native Russian speakers.

We use the same tasks for toxicity detection (Figure 7-3) and content similarity (Figure 7-2) as for parallel dataset collection. But, additionally, we apply the fluency

evaluation task (see Figure 9-1) to both the source and the target and compute the final fluency score from the source and target scores. The original Russian interfaces can be found in Appendix B.1.

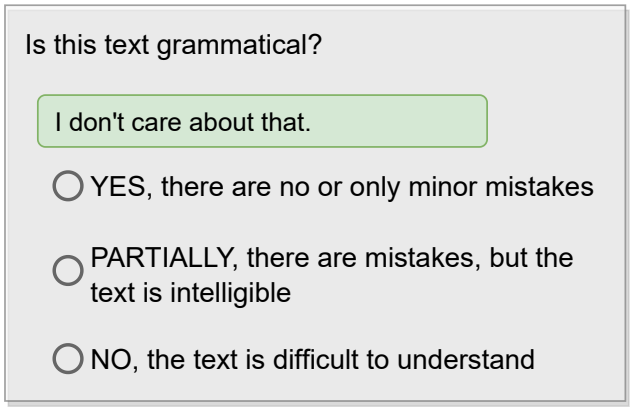


Figure 9-1: Interface of the fluency evaluation task.

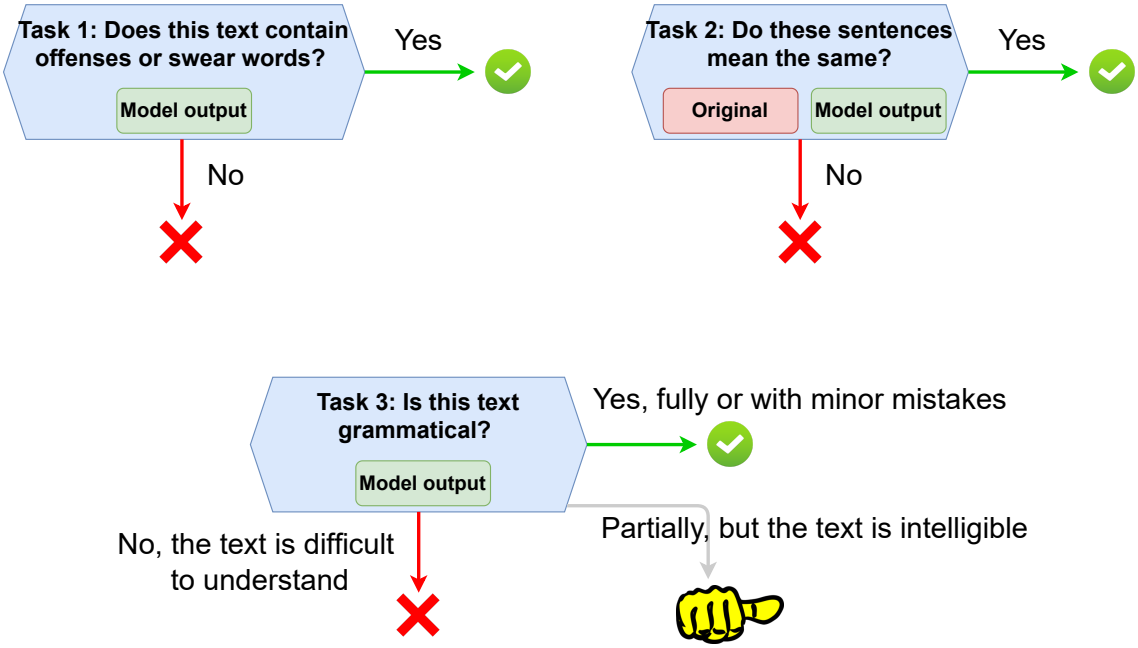


Figure 9-2: All tasks from the pipeline used for human evaluation via crowdsourcing of detoxification systems.

Each sentence in each of the projects is labeled by 10 to 12 workers. We aggregate their result using the Dawid-Skene aggregation method [Dawid and Skene, 1979]. It takes into account the dynamically defined reliability of workers. For each example, with multiple labels, Dawid-Skene method returns the label and its confidence. We

use only labels whose confidence is above 90%. The other labels (around 3% of all examples) are later filled by experts.

The overall schema of crowdsourcing evaluation is presented in Figure 9-2. The interpretation of answers is used the same as described in Section 8.2.2. For quality control, we use the same setup as described in Section 7.4.2.

	STA_a	SIM_a	FL_a	J_a	ChrF
adversarial	0.97	**0.94**	0.96	**0.87**	0.53
ruT5-finetune	**0.98**	0.86	**0.97**	0.82	0.55
ruT5-large	0.95	0.86	**0.97**	0.78	0.57
ruT5-clean	0.95	0.82	0.91	0.71	0.57
lewis	0.93	0.80	0.88	0.66	0.56
ruGPT3-XL	0.94	0.73	0.89	0.61	0.50
RuT5 Baseline	0.80	0.83	0.84	0.56	0.57
ruPrompts-plus	0.80	0.80	0.83	0.54	0.56
ruPrompts	0.81	0.79	0.80	0.53	0.55
ruT5-preproc	0.85	0.76	0.78	0.52	0.53
human references	0.85	0.72	0.78	0.49	**0.77**
ruGPT3-filter	0.83	0.76	0.76	0.48	0.51
RoBERTa-replace	0.57	0.89	0.91	0.44	0.54
Delete	0.56	0.89	0.85	0.41	0.53
Duplicate	0.24	1.00	1.00	0.24	0.56

Table 9.1: The performance of the participating models in terms of automatic metrics, sorted by J_a metric.

9.6 Results

In this section, first, we present the data, namely the outcome of the shared task on detoxification evaluation. Second, we perform an analysis of the correspondence of human and automatic metrics. Finally, we conclude with a discussion of the assessors's performance and overall difficulty of the task.

9.6.1 Models Performance

Table 9.1 shows the performance of the participating models and our baselines in terms of the automatic metrics. The adversarial example generation turns out to be very effective — it attains the highest scores of all metrics, thus yielding the highest

	STA_m	SIM_m	FL_m	J_m
human references	**0.89**	0.82	**0.89**	**0.65**
ruT5-clean	0.79	**0.87**	**0.90**	0.63
RuT5 Baseline	0.79	0.82	**0.92**	0.61
ruT5-large	0.73	**0.87**	**0.92**	0.60
lewis	0.82	0.79	0.85	0.58
ruPrompts-plus	0.78	0.81	**0.90**	0.57
ruT5-finetune	0.80	0.78	0.87	0.56
ruT5-preproc	0.79	0.72	0.78	0.51
ruGPT3-XL	0.81	0.70	0.90	0.50
ruPrompts	0.80	0.70	**0.87**	0.49
ruGPT3-filter	0.77	0.72	0.83	0.45
RoBERTa-replace	0.43	0.62	0.79	0.17
Delete	0.39	0.71	0.73	0.16
Duplicate	0.11	1.00	1.00	0.11
adversarial	0.25	0.13	0.24	0.02

Table 9.2: Manual evaluation of the participating models, the models are sorted by the J_m metric. The figures **in bold** show the highest value of the metric with the significance level of $\alpha = 0.05$.

J_a score. The next three places on the leaderboard are taken by the models based on our baseline ruT5 system. Notice that the human references are below the majority of models in terms of all metrics except ChrF whose score for the human references is the highest by a large margin.

The manual scores (see Table 9.2) provide a completely different result. There, the human references are significantly better than other models but closely followed by one of the ruT5-based systems. However, ruT5-clean (the best-performing participant) is not significantly better than the ruT5 baseline. Interestingly, the **adversarial** model whose automatic scores are the highest, in fact, produces sentences of a very low quality.

9.6.2 Automatic vs Manual Metrics

The automatic and manual metrics (Tables 9.1 and 9.2) provide very diverse results in terms of participants rankings. This suggests that they are weakly correlated.

We check this assumption by computing the Spearman ρ correlations at three different levels: sentence level, system level, and system ranking level. At the sen-

Metric	STA_a	SIM_a	FL_a	J_a	ChrF
STA_m	0.376	**-0.776**	-0.398	0.278	0.223
SIM_m	-0.046	0.031	0.190	0.000	**0.789**
FL_m	-0.083	-0.032	0.288	0.070	**0.619**
J_m	0.326	-0.495	-0.211	0.350	**0.735**

Table 9.3: Spearman's correlation coefficient between automatic VS manual metrics on system level. Bold numbers denote the statistically significant correlation (p-value ≤ 0.05).

Metric	STA_a	SIM_a	FL_a	J_a	ChrF
STA_m	**0.695**	**-0.888**	-0.398	0.305	0.264
SIM_m	-0.305	-0.153	-0.042	-0.431	0.276
FL_m	-0.237	-0.291	-0.116	-0.425	0.218
J_m	**0.595**	**-0.746**	-0.380	0.278	0.367

Table 9.4: Pearson's correlation coefficient between automatic VS manual metrics on system level. Bold numbers denote the statistically significant correlation (p-value ≤ 0.05).

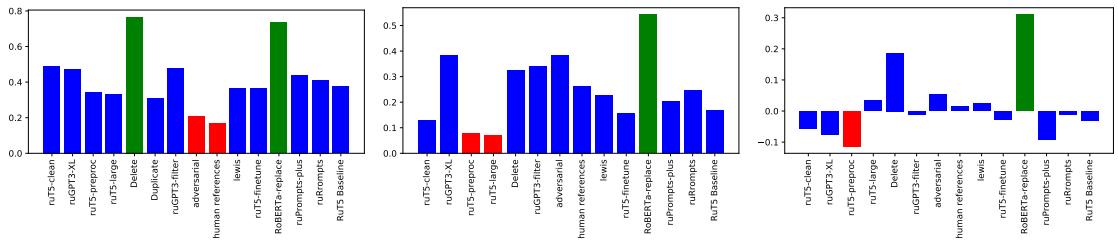


Figure 9-3: Correlations between automatic and manual metrics at the sentence level for different models.
(Right: **STA** metric; Center: **SIM** metric; Left: **FL** metric.)

tence level, we compare automatic metrics for each sentence and then compare them across their manual analogies. For the system level, we first compute average scores for each participant and each metric and then uses such vectors of scores to calculate correlations. As for the system ranking level, we use the rank of the system in the ranked system list instead of the scores, which allows to not take the difference in score distributions into account. The last metric is trying to assess the capability of a metric to predict the outcome of a competition.

System Level Correlations

At the system level, we compute correlation scores of all metrics. We highlight all high correlations (the absolute value above 0.6) in Table 9.4. We clearly see that

none of the automatic metrics correlate with their manually measured counterparts. On the other hand, there is a strong negative correlation between the manual style and automatic content preservation score.

Moreover, manual content and fluency metrics are correlated with the ChrF score. This suggests that ChrF can be used as an automatic evaluation score. On the other hand, ChrF is not sensitive to sentence style, which means that it can be deceived (for example, the trivial Duplicate baseline performs on par with strong T5-based models in terms of ChrF). However, the power of ChrF was also claimed by [Briakou et al., 2021a].

Metric	STA_a	SIM_a	FL_a	J_a
STA_m	-0.437	**0.679**	0.226	0.345
SIM_m	0.187	-0.126	0.099	0.022
FL_m	0.165	-0.314	0.037	-0.046
J_m	-0.041	0.020	0.275	0.178

Table 9.5: Spearman's correlation coefficient between automatic VS manual metrics based on system ranking. Bold numbers denote the statistically significant correlation (p-value ≤ 0.05).

Metric	BertScore	ROUGE-L	BLEU	ChrF
STA_m	**-0.710**	**-0.550**	**-0.600**	-0.296
SIM_m	**0.819**	**0.802**	**0.863**	0.495
Fl_m	**0.796**	**0.675**	**0.700**	0.464
J_m	**0.661**	**0.657**	**0.546**	0.325

Table 9.6: Spearman's correlation coefficient between automatic style transfer VS manual metrics based on system ranking. Bold numbers denote the statistically significant correlation (p-value ≤ 0.05).

System Ranking Level Correlations

We also compute the correlation of rankings of models produced by different metrics using Spearman's ρ correlation. According to Table 9.5, we mostly see weak or no correlation. The rankings by automatic metrics of style, content preservation, and fluency do not correlate with their counterparts produced by manual metrics, apart from the correlation between manual metric of style evaluation (STA_m) and

automatic metric of content preservation (SIM_a).

Despite that ChrF metric counted as more suitable text generation metric for the Russian Language, additionally we computed correlations for other text generation metrics as BLEU [Papineni et al., 2002], ROUGE-L [Sutherlin et al., 2011], and BertScore [Zhang et al., 2020a]. The results are presented in the Table 9.6. Unexpectedly, ChrF does not correlate at all with the manually computed manual metrics, according to the ranking evaluation. BertScore, ROUGE-L, and BLEU demonstrated quite strong correlations with the manual metrics, which are statistically significant in comparison to the ChrF scores. At the same time, from Table 9.6 we can conclude that even the highest correlation numbers (0.661) in our case cannot guarantee high-quality prediction of manual metrics, which still requires further manual evaluation steps.

Sentence-level Correlations

The sentence-level correlations show a slightly different picture. The highest correlation is seen for the style metric, the Spearman ρ score of automatic and manual judgments is 0.418 (moderate correlation). The manual and automatic sentence-level similarity, fluency, and joint scores show very weak or no correlation: 0.251, 0.015, and 0.141, respectively.

However, sentence-level correlations between corresponding manual and automatic metrics differ significantly across models (see Figure 9-3). We see that automatic and manual toxicity scores are much better correlated for the **Delete** and **RoBERTa-replace** models, which are the only models to explicitly remove or replace toxic words identified by a classifier or via a manually compiled list of toxic words. These models apparently produce texts which are easy to classify correctly. Conversely, **adversarial** model and **human references** are the most difficult to classify. The former deliberately "fools" the classifier with artificial examples, while the latter contains non-trivial phrases whose level of toxicity is difficult to grasp automatically.

Analogously, the similarity scores are also better correlated for **RoBERTa-replace** model which leaves the majority of words intact, so for it similarity boils

down to word matching. Instead, T5-based models produce non-trivial paraphrases. These T5 outputs are also difficult to correctly classify for fluency, unlike the models based on word replacements (**RoBERTa-replace** and **Delete**). Overall, we see that it is more difficult to correctly classify *better-performing models* and *models based on large pre-trained language models*. This suggests that the automatic evaluation might fail exactly where we need it most, i.e. in discriminating between the good models.

9.6.3 Assessors Performance

While in many works the human evaluation is considered undoubtedly reliable, we notice that this is not always true. Human evaluation can suffer from: (i) the low reliability of crowd workers and (ii) the difficulty and subjectivity of the tasks.

In crowdsourcing experiments, it is common to give each example for labeling to 3–5 people and aggregate the labels. In our case, 3 annotations per sample were not enough. They yielded labeling with around 10% mistakes. Thus, we collected 10 annotations per sample. Such labeling was more reliable: the error rate did not exceed 3% for style and content and 6% for fluency.

To measure the difficulty of the task, we compute the inter-annotator agreement coefficient Krippendorff's alpha [Krippendorff, 2011]. It turns out that the agreement is moderate: content: 0.522, 0.448, and 0.394 for style, content, and fluency, respectively. The expert Krippendorff's alpha scores are close: 0.584, 0.458, and 0.463. This confirms that in the experiment with 10 annotations per example the crowd workers are reliable enough, but the task itself is subjective.

Interestingly, the style evaluation gains the highest inter-annotator agreement, just as it had the highest correlation between manual and automatic labeling. This suggests that toxicity is more stable and better interpreted by both humans and models.

9.7 Summary

We conducted an evaluation of detoxification models for Russian using both automatic and manual metrics. This allowed us to analyse the relationship between the metrics and assess the suitability of automatic metrics for evaluation.

Our analysis shows that the metrics are overall weakly correlated with the human judgments both at the system and the sentence level. We found that the ChrF score has a strong correlation with the joint score of style, content, and fluency. Thus, ChrF could be used as a proxy for manual evaluation, but its lack of correlation with the style score makes this metric vulnerable to attacks. At the system ranking level, the BertScore metric yielded the best correlation with human judgments.

We also discovered that the correlation between manual and automatic scores varies for different models. This shows the necessity to consider diverse style transfer models for metrics analysis.

Overall, although the state-of-the-art evaluation setup for the detoxification task (three parameters and the joint score combined from them) is conceptually correct, the current performance of automatic metrics is insufficient to use as a replacement for manual evaluation. A worse thing is that the automatic metrics produce less reliability for better-performing models, thus blocking the advance of style transfer models.

Also, we presented the first-of-its-kind pipeline for automated manual evaluation of detoxification models. The adequate level of the inter-annotator agreement confirmed the usefulness of such markup. However, the toxicity classification task itself is subjective. Thus, a more narrow definition of toxicity can be a future improvement of the pipeline.

10

Conclusion

We have witnessed rapid advances in the field of Natural Language Processing (NLP) in recent years. Pre-trained Large Language Model (LLM) from the Transformer [Vaswani et al., 2017] "family" made it possible to solve most classical NLP tasks only fine-tuned on domain dataset. However, there is an ongoing exploration of how modern NLP technologies can be applied not only to atomic industrial tasks but also to socially important problems. In this dissertation, we proposed new approaches and new datasets to combat different types of harmful textual information in multiple languages. Our findings are already implemented in real-life applications and also can be used to open new horizons for future research in the field of NLP4SG.

10.1 Contributions

In **Part I**, we developed approaches to tackle the fight against fake news.

In **Chapter 3**, we provided the task formulation of the fake news classification task, as well as an overview of previously developed models and datasets for this task. We showed that previous work suffers from a lack of multilingual approaches.

For this reason, we presented **Multiverse** – a new feature for fake news detection based on multilingual evidence scraped from the web search (**Chapter 4**). The hypothesis about the different propagation of fake and legit news was confirmed, firstly, by manual annotation. In fact, legitimate news overcomes the natural barrier to cross-checking by journalists working in different languages. However, such

behavior does not repeat for fake news. After manual confirmation, the automated feature Multiverse was incorporated into several baseline fake news detection systems. The use of the proposed feature improves the performance of all baselines by at least $0.2 - 0.4$ for the F_1 score that achieves the highest score in concatenation with BERT-based embeddings.

Subsequently, in **Chapter 5** we explored new methods for measuring multilingual news similarities. We developed several approaches based on different types of multilingual word embeddings and information extraction from news text. For this task again, the usage of BERT-based embeddings in **TransformerEncoderCosSim** approach gives the highest performance. Finally, we showed how the proposed approach for the fake news classification can be used as a demonstration for real-life cases showing descriptive explanations for end users of the model decision.

Part II was dedicated to the development of detoxification technologies.

In **Chapter 6**, we provided motivation of detoxification task on a par with formal problem statement. Additionally, we provided an overview of previously developed approaches for text style transfer tasks in general and the detoxification task specifically. All previous methods fail to pass the human evaluation and never were explored in terms of multilinguality. One of the reasons of this for detoxification is simply the absence of data.

Firstly, we introduced **Paradetox** (**Chapter 7**) – new dataset with parallel pairs of `toxic` ↔ `non-toxic` pairs. We presented a new pipeline for such dataset collection and tested it for two languages – English and Russian. This shows that theoretically such a pipeline can be easily extended to any other language. We explored the ways how toxic sentences were written and which edits were made by annotators. It was found that while up to 30% swear words can be just removed, the other parts of the text with toxicity should be fully rewritten in at least 60% of cases. It proves at the data analysis level that to solve the detoxification task properly it is required not just to edit text point-wisely but to rewrite it, to generate new text (or, at least, some part of it) from scratch.

In Chapter 8, we confirmed the hypothesis that the usage of a parallel dataset improves the performance of NLP models on the detoxification task. We introduced

new unsupervised approach for text style transfer – **condBERT**. Then, we fine-tuned different versions of the T5 and BART models on Paradetox and compared them with previous state-of-the-art baselines. For both languages, **EN-Detox** and **RU-Detox** significantly outperformed baselines. For the English language, the improvement in the J metric reached 0.04 for automatic evaluation and 0.39 for manual evaluation compared to the strongest baseline. For the Russian language, a similar comparison brought 0.04 for automatic and 0.44 for manual evaluation. After monolingual detoxification, we investigated first-of-its-kind approaches for multilingual and cross-lingual detoxification. Although multilingual detoxification in the presence of parallel data is possible with the mBART and mT5 models, the cross-lingual setup is still quite difficult to handle. In the end, we showed how the proposed detoxification approaches can be used in real-life systems. We showed use cases with chatbots of how systems can be integrated into the game industry and provided an available online demonstration.

In addition to methods exploration, we investigated the metrics for text style transfer evaluation (**Chapter 9**). We calculated the correlation between automatic and human assessments for 15 detoxification systems. Unfortunately, a poor correlation was found for all parameters of detoxification task evaluation – style transfer accuracy, content similarity, and fluency. However, the human references-based metrics such as BLEU, ChrF, and BERT-Score reach high correlations (over 60%) with all three metrics' parameters and with the joint J score. This showed that these metrics can be used as a more realistic human assessment approximation for system ranking.

To sum up this section and this dissertation in general, we provide answers to the research questions stated in Section 1.1:

Q1: How can fake news detection benefit from multilingual evidence?

We showed that fake news detection performance can be significantly improved with the usage of Multiverse – multilingual evidence from external search. We showed how multilingual word embedding models can be used to develop new metrics for multilingual news similarity measures. The proposed approaches can be scaled to

new languages with little effort. In addition, we provided the system demonstration of how multilingual data can add explainability to the fake news classification system's decision from the user's perspective.

Q2: What NLP technologies (both monolingual and multilingual) can be used to detoxify texts?

Previously existing approaches did not achieve suitable performance according to human evaluation. So, we introduced a new method for parallel data collection for the detoxification task that can be scaled to any language. Additionally, we developed new monolingual methods. The unsupervised method condBERT did not show the best results, however, it has a good perspective in terms of multilingual and cross-lingual application for detoxification. Models based on seq2seq approach showed the best results achieving quite high scores from annotators. This showed the credibility of approaches to be used in real-world applications. Also, Large Language Model (LLM)s were proved to make it possible to scale detoxification for multilingual setup, but cross-lingual model development still should be investigated.

10.2 Future directions

While this dissertation took several steps forward to make the NLP method more applicable for social problems, there are still directions to investigate to bring the methods to the ideal and make them applicable in production.

10.2.1 More language coverage

While this dissertation proposes approaches that cover several languages, not all language variety is covered. For propaganda detection, there is still only one annotated corpus that covers only the English language. For fake news, only the most popular European languages are taken into account in this work. An additional exploration of fake and legit news spread in other European, Asian, and African languages is needed. Moreover, it was shown that modern approaches to multilingual news sim-

ilarities measure can lose performance while dealing with Asian languages. Thus, further research is needed to determine a more scalable and stable metric.

For toxicity neutralization, we covered two languages while there were toxicity classification datasets already existed. For other languages, the situation is not that optimistic. While there is a multilingual version of the Jigsaw dataset [kag, 2019], the dataset still covers mostly European languages. Out of European languages, for instance, there is a corpus of toxic Thai tweets [Sirihattasak et al., 2018] that can also be used to create a parallel detoxification corpus. Also, additional experiments are still required to address the problem of cross-lingual detoxification. As the problem of toxicity dataset available for a language is quite realistic, a solution to how to propagate the knowledge of toxic and non-toxic styles for a new language should be found. One of the approaches can be the usage of the Adapter layer [Pfeiffer et al., 2020]. Such an idea was already tested for formality style transfer in [Lai et al., 2022] for European languages.

10.2.2 More toxicity types variety

In this work, we covered obvious toxicity types that are expressed with rude words and direct insults. However, more hidden types of toxic language such as sarcasm or passive aggressiveness can be even more insulting. Also, such types of toxicity as racism or sexism can be addressed [Sánchez-Junquera et al., 2021, Frenda et al., 2019]. It is important to find a solution to neutralize such types of toxicity. One of the promising solutions can be an implementation of "positive frames" as presented in [Ziems et al., 2022]. The toxic text is rephrased in a more tolerant way with the addition of a positive way of thinking. For example: the phrase *i absolutely hate making decisions* can be formulated with more optimism as *I have a lot of decisions to make. It will become easier once I start to get used to it.* In addition, to neutralize toxic and hate speech, counterspeech can be generated. Examples and existing data sets for counter speech generation can be found in [Tekiroglu et al., 2020].

10.2.3 Human-in-the-loop

As was shown in Chapter 8, the random extension of the data set does not always bring an improvement in the performance of the detoxification model. It will be more efficient to add to datasets such samples in which the model is unsure, thus, they are new to the model. This goal can be achieved with Active Learning (AL) techniques. AL has already been explored for text classification tasks in [Shelmanov et al., 2021]. Also, for text generation Uncertainty Estimation techniques were introduced in [Malinin and Gales, 2021]. Toxicity, as language itself, can develop over years and generations. Both the toxicity classification and detoxification models can be improved with AL while tackling new samples with unknown toxicity.

In addition, the fake news, propaganda, and toxicity classification task can benefit greatly by adding more explanation. Thus, for the English language, there is a HateXplain dataset [Mathew et al., 2021] with an explanation of why a sample can be counted as hate, offensive, or normal. Such datasets can be used to generate fluent text with explanations and show them to the user. At the same time, the user can also correct the decision of the model by adjusting the explanation. For example, the model gives more weight to words that seem to be not toxic. Such adjustments than can be provided to the model and its weight can be recalculated. The overview of how human debugging can be used for NLP models is provided in [Lertvittayakumjorn and Toni, 2021]. Adding human interaction with NLP models can help to create more fair technology and ensure that it indeed helps to make a step for social good impact.

Bibliography

Jigsaw multilingual toxic comment classification. https://www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification, 2019. Accessed: 2021-01-13.

Rami Al-Rfou, Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. POLYGLOT-NER: massive multilingual named entity recognition. In Suresh Venkatasubramanian and Jieping Ye, editors, *Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, BC, Canada, April 30 - May 2, 2015*, pages 586–594. SIAM, 2015. doi:10.1137/1.9781611974010.66. URL https://doi.org/10.1137/1.9781611974010.66.

Firoj Alam, Fahim Dalvi, Shaden Shaar, Nadir Durrani, Hamdy Mubarak, Alex Nikolov, Giovanni Da San Martino, Ahmed Abdelali, Hassan Sajjad, Kareem Darwish, et al. Fighting the covid-19 infodemic in social media: A holistic perspective and a call to arms.

Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–36, 2017.

Nikolay Arefyev, Boris Sheludko, Alexander Podolskiy, and Alexander Panchenko. Always keep your target in mind: Studying semantics and improving performance of neural lexical substitution. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1242–1255, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi:10.18653/v1/2020.coling-main.107. URL https://aclanthology.org/2020.coling-main.107.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, 2020.

Alberto Barrón-Cedeno, Israa Jaradat, Giovanni Da San Martino, and Preslav Nakov. Proppy: Organizing the news based on their propagandistic content. *Information Processing & Management*, 2019.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic*

Evaluation, pages 54–63, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi:10.18653/v1/S19-2007. URL https://aclanthology.org/S19-2007.

Parminder Bhatia, Kristjan Arumae, Nima Pourdamghani, Suyog Deshpande, Ben Snively, Mona Mona, Colby Wise, George Price, Shyam Ramaswamy, and Taha A. Kass-Hout. AWS cord19-search: A scientific literature search engine for COVID-19. *CoRR*, abs/2007.09186, 2020. URL https://arxiv.org/abs/2007.09186.

BigScience. Bigscience language open-science open-access multilingual (BLOOM) language model, 2022. URL https://huggingface.co/bigscience/bloom.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguistics*, 5: 135–146, 2017. URL https://transacl.org/ojs/index.php/tacl/article/view/999.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics, 2015. doi:10.18653/v1/d15-1075. URL https://doi.org/10.18653/v1/d15-1075.

Luke Breitfeller, Emily Ahn, David Jurgens, and Yulia Tsvetkov. Finding microaggressions in the wild: A case for locating elusive phenomena in social media posts. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1664–1674, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1176. URL https://aclanthology.org/D19-1176.

Eleftheria Briakou, Sweta Agrawal, Joel Tetreault, and Marine Carpuat. Evaluating the evaluation metrics for style transfer: A case study in multilingual formality transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1321–1336, Online and Punta Cana, Dominican Republic, November 2021a. Association for Computational Linguistics. doi:10.18653/v1/2021.emnlp-main.100. URL https://aclanthology.org/2021.emnlp-main.100.

Eleftheria Briakou, Di Lu, Ke Zhang, and Joel Tetreault. Olá, bonjour, salve! XFORMAL: A benchmark for multilingual formality style transfer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3199–3216, Online, June 2021b. Association for Computational Linguistics. doi:10.18653/v1/2021.naacl-main.256. URL https://aclanthology.org/2021.naacl-main.256.

Keith Carlson, Allen Riddell, and Daniel Rockmore. Evaluating prose style transfer with the bible. *Royal Society Open Science*, 5, October 2018. URL https://arxiv.org/abs/1711.04731.

Samuel Carton, Qiaozhu Mei, and Paul Resnick. Feature-based explanations don't help people detect misclassifications of online toxicity. In Munmun De Choudhury, Rumi Chunara, Aron Culotta, and Brooke Foucault Welles, editors, *Proceedings of the Fourteenth International AAAI Conference on Web and Social Media, ICWSM 2020, Held Virtually, Original Venue: Atlanta, Georgia, USA, June 8-11, 2020*, pages 95–106. AAAI Press, 2020. URL https://ojs.aaai.org/index.php/ICWSM/article/view/7282.

Kunal Chawla and Diyi Yang. Semi-supervised formality style transfer using language model discriminator and mutual information maximization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2340–2354, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.findings-emnlp.212. URL https://aclanthology.org/2020.findings-emnlp.212.

Mia Xu Chen, Benjamin N. Lee, Gagan Bansal, Yuan Cao, Shuyuan Zhang, Justin Lu, Jackie Tsay, Yinan Wang, Andrew M. Dai, Zhifeng Chen, Timothy Sohn, and Yonghui Wu. Gmail smart compose: Real-time assisted writing. In Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis, editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2287–2295. ACM, 2019. doi:10.1145/3292500.3330723. URL https://doi.org/10.1145/3292500.3330723.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4232-2. doi:10.1145/2939672.2939785. URL http://doi.acm.org/10.1145/2939672.2939785.

Xi Chen, Ali Zeynali, Chico Camargo, Fabian Flöck, Devin Gaffney, Przemyslaw Grabowicz, Scott Hale, David Jurgens, and Mattia Samory. Semeval-2022 task 8: Multilingual news article similarity. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1094–1106, 2022.

Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 551–561. The Association for Computational Linguistics, 2016. doi:10.18653/v1/d16-1053. URL https://doi.org/10.18653/v1/d16-1053.

Yu Cheng, Zhe Gan, Yizhe Zhang, Oussama Elachqar, Dianqi Li, and Jingjing Liu. Contextual text style transfer. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2915–2924, Online, November 2020. Association

for Computational Linguistics. doi:10.18653/v1/2020.findings-emnlp.263. URL https://aclanthology.org/2020.findings-emnlp.263.

Anshika Choudhary and Anuja Arora. Linguistic feature based learning model for fake news detection and classification. *Expert Systems with Applications*, 169: 114171, 2021.

Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7057–7067, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/c04c19c2c2474dbf5f7ac4372c5b9af1-Abstract.html.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: evaluating cross-lingual sentence representations. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2475–2485. Association for Computational Linguistics, 2018. doi:10.18653/v1/d18-1269. URL https://doi.org/10.18653/v1/d18-1269.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451. Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.acl-main.747. URL https://doi.org/10.18653/v1/2020.acl-main.747.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. No language left behind: Scaling human-centered machine translation. *arXiv e-prints*, pages arXiv–2207, 2022.

Josh Cowls, Andreas Tsamados, Mariarosaria Taddeo, and Luciano Floridi. A definition, benchmark and database of AI for social good initiatives. *Nat. Mach. Intell.*, 3(2):111–115, 2021. doi:10.1038/s42256-021-00296-0. URL https://doi.org/10.1038/s42256-021-00296-0.

Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. SemEval-2020 task 11: Detection of propaganda techniques in news articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1377–1414, Barcelona (online), December 2020. International Committee for Computational Linguistics. doi:10.18653/v1/2020.semeval-1.186. URL https://aclanthology.org/2020.semeval-1.186.

David Dale, Anton Voronov, **Daryna Dementieva**, Varvara Logacheva, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. Text detoxification using large pre-trained neural models. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 7979–7996. Association for Computational Linguistics, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.629.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=H1edEyBKDS.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media (ICWSM-17)*, Montreal, Canada, May 2017.

Alexander P. Dawid and Allan Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of The Royal Statistical Society Series C-applied Statistics*, 28:20–28, 1979. URL https://www.jstor.org/stable/2346806.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics. doi:10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

Sumanth Doddapaneni, Gowtham Ramesh, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M. Khapra. A primer on pretrained multilingual language models. *CoRR*, abs/2107.00676, 2021. URL https://arxiv.org/abs/2107.00676.

Cícero Nogueira dos Santos, Igor Melnyk, and Inkit Padhi. Fighting offensive language on social media with unsupervised text style transfer. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 189–194. Association for Computational Linguistics, 2018. doi:10.18653/v1/P18-2031. URL https://aclanthology.org/P18-2031/.

Ashwin Geet D'Sa, Irina Illina, and Dominique Fohr. Towards non-toxic landscapes: Automatic toxic comment detection using DNN. In *Proceedings of the*

Second Workshop on Trolling, Aggression and Cyberbullying, pages 21–25, Marseille, France, May 2020. European Language Resources Association (ELRA). ISBN 979-10-95546-56-6.

Sue DW, Capodilupo CM, Torino GC, Bucceri JM, Holder AM, Nadal KL, and Esquilin M. Racial microaggressions in everyday life: implications for clinical practice. *Am Psychol.*, 62, May-Jun 2007.

Ullrich KH Ecker, Joshua L Hogan, and Stephan Lewandowsky. Reminders and repetition of misinformation: Helping or hindering its retraction? *Journal of Applied Research in Memory and Cognition*, 6(2):185–192, 2017.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. Language-agnostic BERT sentence embedding. *CoRR*, abs/2007.01852, 2020. URL https://arxiv.org/abs/2007.01852.

Luciano Floridi, Josh Cowls, Thomas C. King, and Mariarosaria Taddeo. How to design AI for social good: Seven essential factors. *Sci. Eng. Ethics*, 26(3):1771–1796, 2020. doi:10.1007/s11948-020-00213-5. URL https://doi.org/10.1007/s11948-020-00213-5.

Paula Fortuna and Sérgio Nunes. A survey on automatic detection of hate speech in text. *ACM Comput. Surv.*, 51(4), July 2018. ISSN 0360-0300. doi:10.1145/3232676. URL https://doi.org/10.1145/3232676.

Markus Freitag, Isaac Caswell, and Scott Roy. APE at scale and its implications on MT evaluation biases. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 34–44, Florence, Italy, August 2019. Association for Computational Linguistics. doi:10.18653/v1/W19-5204. URL https://aclanthology.org/W19-5204.

Simona Frenda, Bilal Ghanem, Manuel Montes-y-Gómez, and Paolo Rosso. Online hate speech against women: Automatic identification of misogyny and sexism on twitter. *J. Intell. Fuzzy Syst.*, 36(5):4743–4752, 2019. doi:10.3233/JIFS-179023. URL https://doi.org/10.3233/JIFS-179023.

Tanmay Garg, Sarah Masud, Tharun Suresh, and Tanmoy Chakraborty. Handling bias in toxic speech detection: A survey. *CoRR*, abs/2202.00126, 2022. URL https://arxiv.org/abs/2202.00126.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2414–2423. IEEE Computer Society, 2016. doi:10.1109/CVPR.2016.265.

Christine Geeng, Savanna Yee, and Franziska Roesner. Fake news on facebook and twitter: Investigating how people (don't) investigate. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

Bilal Ghanem, Manuel Montes-y Gòmez, Francisco Rangel, and Paolo Rosso. Upv-inaoe-autoritas-check that: An approach based on external sources to detect claims credibility. In *Proceedings of the Conference and Labs of the Evaluation Forum (CLEF'18)*, 2018.

Bilal Ghanem, Paolo Rosso, and Francisco Rangel. An emotional analysis of false information in social media and news articles. *ACM Transactions on Internet Technology (TOIT)*, 20(2):1–18, 2020.

Anna Glazkova, Maksim Glazkov, and Timofey Trifonov. g2tmn at constraint@ aaai2021: Exploiting ct-bert and ensembling learning for covid-19 fake news detection. *arXiv e-prints*, pages arXiv–2012, 2020.

Georgios Gravanis, Athena Vakali, Konstantinos Diamantaras, and Panagiotis Karadais. Behind the cues: A benchmarking study for fake news detection. *Expert Systems with Applications*, 128:201–213, 2019.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomás Mikolov. Learning word vectors for 157 languages. In Nicoletta Calzolari, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Kôiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA), 2018. URL http://www.lrec-conf.org/proceedings/lrec2018/summaries/627.html.

Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *CoRR*, abs/1410.5401, 2014. URL http://arxiv.org/abs/1410.5401.

Peter P Groumpos. A critical historical and scientific overview of all industrial revolutions. *IFAC-PapersOnLine*, 54(13):464–471, 2021.

Sunil Gundapu and Radhika Mamid. Transformer based automatic covid-19 fake news detection system. *arXiv e-prints*, pages arXiv–2101, 2021.

Xiaochuang Han and Yulia Tsvetkov. Fortifying toxic speech detectors against veiled toxicity. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7732–7739, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.622. URL https://aclanthology.org/2020.emnlp-main.622.

Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M. Meyer, and Iryna Gurevych. A retrospective analysis of the fake news challenge stance-detection task. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1859–1874, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics.

Maram Hasanain, Reem Suwaileh, Tamer Elsayed, Alberto Barrón-Cedeno, and Preslav Nakov. Overview of the clef-2019 checkthat! lab on automatic identification and verification of claims. task 2: Evidence and factuality. In *CLEF*, 2019.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. A probabilistic formulation of unsupervised text style transfer. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=HJlA0C4tPS.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spacy: Industrial-strength natural language processing in python. 2020. doi:10.5281/zenodo.1212303.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Toward controlled generation of text. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1587–1596. PMLR, 2017. URL http://proceedings.mlr.press/v70/hu17e.html.

Fei Huang, Zikai Chen, Chen Henry Wu, Qihan Guo, Xiaoyan Zhu, and Minlie Huang. NAST: A non-autoregressive generator with word alignment for unsupervised text style transfer. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1577–1590, Online, August 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.findings-acl.138. URL https://aclanthology.org/2021.findings-acl.138.

Jigsaw. Toxic comment classification challenge. https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge, 2018. Accessed: 2021-03-01.

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. Deep learning for text style transfer: A survey. *CoRR*, abs/2011.00416, 2020. URL https://arxiv.org/abs/2011.00416.

Zhijing Jin, Geeticka Chauhan, Brian Tse, Mrinmaya Sachan, and Rada Mihalcea. How good is nlp? A sober look at NLP tasks through the lens of social impact. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 3099–3113. Association for Computational Linguistics, 2021. doi:10.18653/v1/2021.findings-acl.273. URL https://doi.org/10.18653/v1/2021.findings-acl.273.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. Disentangled representation learning for non-parallel text style transfer. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence,*

Italy, July 28- August 2, 2019, Volume 1: Long Papers, pages 424–434. Association for Computational Linguistics, 2019. doi:10.18653/v1/p19-1041. URL https://doi.org/10.18653/v1/p19-1041.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Heejung Jwa, Dongsuk Oh, Kinam Park, Jang Mook Kang, and Heuiseok Lim. exbake: automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences*, 9(19):4062, 2019.

Kaggle. Russian language toxic comments. https://www.kaggle.com/blackmoon/russian-language-toxic-comments, 2019. Accessed: 2021-03-01.

Kaggle. Toxic russian comments. https://www.kaggle.com/alexandersemiletov/toxic-russian-comments, 2020. Accessed: 2021-03-01.

Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. Fndnet–a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61:32–44, 2020.

Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. Fakebert: Fake news detection in social media with a bert-based deep learning approach. *Multimedia Tools and Applications*, pages 1–24, 2021.

Cecilia Kang and Adam Goldman. In washington pizzeria attack, fake news brought real guns. *New York Times*, 5, 2016.

Georgi Karadzhov, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev. Fully automated fact checking using external sources. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 344–353, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Nikita Konodyuk and Maria Tikhonova. Continuous prompt tuning for russian: how to learn prompts efficiently with rugpt3? In *Proceedings of the International Conference on Analysis of Images, Social Networks and Texts*, 2021.

Moshe Koppel and Noam Ordan. Translationese and its dialects. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:*

Human Language Technologies, pages 1318–1326, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1132.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq R. Joty, Richard Socher, and Nazneen Fatema Rajani. Gedi: Generative discriminator guided sequence generation. *CoRR*, abs/2009.06367, 2020. URL https://arxiv.org/abs/2009.06367.

Klaus Krippendorff. Computing krippendorff's alpha-reliability. 2011.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. Reformulating unsupervised style transfer as paraphrase generation. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 737–762. Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.emnlp-main.55.

Mikhail Kuimov, **Daryna Dementieva**, and Alexander Panchenko. SkoltechNLP at semeval-2022 task 8: Multilingual news article similarity via exploration of news texts to vector representations. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1136–1144, 2022.

Yuri Kuratov and Mikhail Arkhipov. Adaptation of deep bidirectional multilingual transformers for russian language. *CoRR*, abs/1905.07213, 2019.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. Generic resources are what you need: Style transfer tasks without task-specific parallel training data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4241–4254, Online and Punta Cana, Dominican Republic, November 2021a. Association for Computational Linguistics. doi:10.18653/v1/2021.emnlp-main.349. URL https://aclanthology.org/2021.emnlp-main.349.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. Thank you BART! rewarding pre-trained models improves formality style transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 484–494, Online, August 2021b. Association for Computational Linguistics. doi:10.18653/v1/2021.acl-short.62. URL https://aclanthology.org/2021.acl-short.62.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. Multilingual pre-training with language and task adaptation for multilingual text style transfer. *CoRR*, abs/2203.08552, 2022. doi:10.48550/arXiv.2203.08552. URL https://doi.org/10.48550/arXiv.2203.08552.

Leo Laugier, John Pavlopoulos, Jeffrey Sorensen, and Lucas Dixon. Civil rephrases of toxic texts with self-supervised transformers. *CoRR*, abs/2102.05456, 2021. URL https://arxiv.org/abs/2102.05456.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.363. URL https://aclanthology.org/2020.emnlp-main.363.

Dongkyu Lee, Zhiliang Tian, Lanqing Xue, and Nevin L. Zhang. Enhancing content preservation in text style transfer using reverse attention and conditional layer normalization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 93–102, Online, August 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.acl-long.8. URL https://aclanthology.org/2021.acl-long.8.

Joosung Lee. Stable style transformer: Delete and generate approach with encoder-decoder for text style transfer. In Brian Davis, Yvette Graham, John D. Kelleher, and Yaji Sripada, editors, *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pages 195–204. Association for Computational Linguistics, 2020. URL https://aclanthology.org/2020.inlg-1.25/.

Alyssa Lees, Daniel Borkan, Ian Kivlichan, Jorge Nario, and Tesh Goyal. Capturing covertly toxic speech via crowdsourcing. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 14–20, Online, April 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.hcinlp-1.3.

Gennadi Lembersky, Noam Ordan, and Shuly Wintner. Language models for machine translation: Original vs. translated texts. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 363–374, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL https://aclanthology.org/D11-1034.

Piyawat Lertvittayakumjorn and Francesca Toni. Explanation-based human debugging of NLP models: A survey. *CoRR*, abs/2104.15135, 2021. URL https://arxiv.org/abs/2104.15135.

Stephan Lewandowsky, Ullrich KH Ecker, Colleen M Seifert, Norbert Schwarz, and John Cook. Misinformation and its correction: Continued influence and successful debiasing. *Psychological science in the public interest*, 13(3):106–131, 2012.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703.

Dianqi Li, Yizhe Zhang, Zhe Gan, Yu Cheng, Chris Brockett, Bill Dolan, and Ming-Ting Sun. Domain adaptive text style transfer. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3304–3313, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1325. URL https://aclanthology.org/D19-1325.

Juncen Li, Robin Jia, He He, and Percy Liang. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1865–1874, New Orleans, Louisiana, June 2018a. Association for Computational Linguistics. doi:10.18653/v1/N18-1169. URL https://aclanthology.org/N18-1169.

Juncen Li, Robin Jia, He He, and Percy Liang. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1865–1874, New Orleans, Louisiana, June 2018b. Association for Computational Linguistics. doi:10.18653/v1/N18-1169. URL https://aclanthology.org/N18-1169.

Liangda Li and Hongyuan Zha. Energy usage behavior modeling in energy disaggregation via marked hawkes process. In Blai Bonet and Sven Koenig, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 672–678. AAAI Press, 2015. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9623.

Qifei Li and Wangchunshu Zhou. Connecting the dots between fact verification and fake news detection. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1820–1825, 2020.

Kevin Lin, Ming-Yu Liu, Ming-Ting Sun, and Jan Kautz. Learning to generate multiple style transfer outputs for an input sentence. In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 10–23, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.ngt-1.2. URL https://aclanthology.org/2020.ngt-1.2.

Yang Liu and Yi-Fang Wu. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019a. URL http://arxiv.org/abs/1907.11692.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for

neural machine translation. *Trans. Assoc. Comput. Linguistics*, 8:726–742, 2020. URL https://transacl.org/ojs/index.php/tacl/article/view/2107.

Zhan Liu, Shaban Shabani, Nicole Glassey Balet, and Maria Sokhn. Detection of satiric news on social media: analysis of the phenomenon with a french dataset. In *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–6. IEEE, 2019b.

Varvara Logacheva*, **Daryna Dementieva***, Sergey Ustyantsev, Daniil Moskovskiy, David Dale, Irina Krotova, Nikita Semenov, and Alexander Panchenko. ParaDetox: Detoxification with parallel data. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6804–6818. Association for Computational Linguistics, 2022a. URL https://aclanthology.org/2022.acl-long.469.

Varvara Logacheva*, **Daryna Dementieva***, Irina Krotova, Alena Fenogenova, Irina Nikishina, Tatiana Shavrina, and Alexander Panchenko. A study on manual and automatic evaluation for text style transfer: The case of detoxification. In *Proceedings of the 2nd Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 90–101, Dublin, Ireland, May 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.humeval-1.8.

Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Xu Sun, and Zhifang Sui. A dual reinforcement learning framework for unsupervised text style transfer. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5116–5122. ijcai.org, 2019. doi:10.24963/ijcai.2019/711. URL https://doi.org/10.24963/ijcai.2019/711.

Yun Ma, Yangbin Chen, Xudong Mao, and Qing Li. Collaborative learning of bidirectional decoders for unsupervised text style transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9250–9266, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.emnlp-main.729. URL https://aclanthology.org/2021.emnlp-main.729.

Bill MacCartney and Christopher D Manning. *Natural language inference*. Citeseer, 2009.

Aman Madaan, Amrith Setlur, Tanmay Parekh, Barnabás Póczos, Graham Neubig, Yiming Yang, Ruslan Salakhutdinov, Alan W. Black, and Shrimai Prabhumoye. Politeness transfer: A tag and generate approach. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1869–1881. Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.acl-main.169.

Andrey Malinin and Mark J. F. Gales. Uncertainty estimation in autoregressive structured prediction. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=jN5y-zb5Q7m.

Eric Malmi, Aliaksei Severyn, and Sascha Rothe. Unsupervised text style transfer with padded masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8671–8680, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.699. URL https://aclanthology.org/2020.emnlp-main.699.

Alejandro Martín, Javier Huertas-Tato, Álvaro Huertas-García, Guillermo Villar-Rodríguez, and David Camacho. Facter-check: Semi-automated fact-checking through semantic similarity and natural language inference. *CoRR*, abs/2110.14532, 2021. URL https://arxiv.org/abs/2110.14532.

Neo D. Martinez, Perrine Tonnin, Barbara Bauer, Rosalyn C. Rael, Rahul Singh, Sanghyuk Yoon, Ilmi Yoon, and Jennifer A. Dunne. Sustaining economic exploitation of complex ecosystems in computational models of coupled human-natural networks. In Jörg Hoffmann and Bart Selman, editors, *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*. AAAI Press, 2012. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/5123.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 14867–14875. AAAI Press, 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17745.

David D. McDonald and James Pustejovsky. A computational theory of prose style for natural language generation. In Maghi King, editor, *EACL 1985, 2nd Conference of the European Chapter of the Association for Computational Linguistics, March 27-29, 1985, University of Geneva, Geneva, Switzerland*, pages 187–193. The Association for Computer Linguistics, 1985.

Igor Melnyk, Cícero Nogueira dos Santos, Kahini Wadhawan, Inkit Padhi, and Abhishek Kumar. Improved neural text attribute transfer with non-parallel data. *CoRR*, abs/1711.09395, 2017. URL http://arxiv.org/abs/1711.09395.

Meta. Increasing our efforts to fight false news. , 2018. Accessed: 2022-08-08.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In

Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

Daniil Moskovskiy, **Daryna Dementieva**, and Alexander Panchenko. Exploring cross-lingual text detoxification with large multilingual language models. In Samuel Louvan, Andrea Madotto, and Brielen Madureira, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 346–354. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.acl-srw.26.

Ryo Nagata, Masato Hagiwara, Hanawa Kazuaki, and Masato Mita. Genchal 2022: Feedback comment generation for writing learning. , 2022. Accessed: 2022-08-08.

Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. Fang: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1165–1174, 2020.

Yixin Nie, Haonan Chen, and Mohit Bansal. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6859–6866, 2019.

Cicero Nogueira dos Santos, Igor Melnyk, and Inkit Padhi. Fighting offensive language on social media with unsupervised text style transfer. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 189–194, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi:10.18653/v1/P18-2031.

Jeppe Nørregaard and Leon Derczynski. DanFEVER: claim verification dataset for Danish. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 422–428, Reykjavik, Iceland (Online), May 31–2 June 2021. Linköping University Electronic Press, Sweden. URL https://aclanthology.org/2021.nodalida-main.47.

Jeppe Nørregaard, Benjamin D Horne, and Sibel Adalı. Nela-gt-2018: A large multi-labelled news dataset for the study of misinformation in news articles. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 13, pages 630–638, 2019.

Endang Wahyu Pamungkas and Viviana Patti. Cross-domain and cross-lingual abusive language detection: A hybrid approach with deep learning and a multilingual lexicon. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 363–370, Florence, Italy, July 2019. Association for Computational Linguistics. doi:10.18653/v1/P19-2051.

Richard Yuanzhe Pang and Kevin Gimpel. Unsupervised evaluation metrics and learning criteria for non-parallel textual transfer. In Alexandra Birch, Andrew M. Finch, Hiroaki Hayashi, Ioannis Konstas, Thang Luong, Graham Neubig, Yusuke Oda, and Katsuhito Sudoh, editors, *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019, Hong Kong, November 4, 2019*, pages 138–147. Association for Computational Linguistics, 2019. doi:10.18653/v1/D19-5614.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL, 2002. doi:10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040/.

Parth Patwa, Shivam Sharma, Srinivas PYKL, Vineeth Guptha, Gitanjali Kumari, Md Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. Fighting an infodemic: Covid-19 fake news dataset. *arXiv e-prints*, pages arXiv–2011, 2020.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. Automatic detection of fake news. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3391–3401, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulic, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. Adapterhub: A framework for adapting transformers. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 46–54. Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.emnlp-demos.7. URL https://doi.org/10.18653/v1/2020.emnlp-demos.7.

Kay T Pham, Amir Nabizadeh, and Salih Selek. Artificial intelligence and chatbots in psychiatry. *Psychiatric Quarterly*, pages 1–5, 2022.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. Credibility assessment of textual claims on the web. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 2173–2178, 2016.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. Where the truth lies: Explaining the credibility of emerging claims on the web and social media. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1003–1012, 2017.

Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi:10.18653/v1/W15-3049. URL https://aclanthology.org/W15-3049.

Juan-Pablo Posadas-Durán, Helena Gomez-Adorno, Grigori Sidorov, and Jesús Jaime Moreno Escobar. Detection of fake news in a new corpus for the spanish language. *Journal of Intelligent & Fuzzy Systems*, 36(5):4869–4876, 2019.

Reid Pryzant, Richard Diehl Martinez, Nathan Dass, Sadao Kurohashi, Dan Jurafsky, and Diyi Yang. Automatically neutralizing subjective bias in text. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 480–489. AAAI Press, 2020.

John Alexander Quinn, Kevin Leyton-Brown, and Ernest Mwebaze. Modeling and monitoring crop disease in developing countries. In Wolfram Burgard and Dan Roth, editors, *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7-11, 2011*. AAAI Press, 2011. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3777.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1 (8):9, 2019. URL https://openai.com/blog/better-language-models/.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Sudha Rao and Joel Tetreault. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi:10.18653/v1/N18-1012. URL https://aclanthology.org/N18-1012.

Machel Reid and Victor Zhong. LEWIS: Levenshtein editing for unsupervised text style transfer. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3932–3944, Online, August 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.findings-acl.344. URL https://aclanthology.org/2021.findings-acl.344.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics, 2019. doi:10.18653/v1/D19-1410. URL https://doi.org/10.18653/v1/D19-1410.

Parker Riley, Noah Constant, Mandy Guo, Girish Kumar, David Uthus, and Zarana Parekh. TextSETTR: Few-shot text style extraction and tunable targeted restyling. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3786–3800, Online, August 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.acl-long.293. URL https://aclanthology.org/2021.acl-long.293.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

Fariba Sadeghi, Amir Jalaly Bidgoly, and Hossein Amirkhani. Fake news detection on social media using a natural language inference approach.

Javier Sánchez-Junquera, Berta Chulvi, Paolo Rosso, and Simone Paolo Ponzetto. How do you speak about immigrants? taxonomy and stereoimmigrants dataset for identifying stereotypes about immigrants. *Applied Sciences*, 11(8):3610, 2021.

Anna Schmidt and Michael Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain, April 2017. Association for Computational Linguistics. doi:10.18653/v1/W17-1101.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi:10.18653/v1/P16-1162. URL https://aclanthology.org/P16-1162.

Khaustov Sergei, Kabaev Andrey, Gorlova Nadezda, and Kalmykov Andrey. Bert for russian news clustering. In *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2021". Moscow, Russia (Online)*, 2021. doi:10.28995/2075-7182-2021-20-385-390.

Gautam Kishore Shahi and Durgesh Nandini. Fakecovid-a multilingual cross-domain fact check news dataset for covid-19.

Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3):1–42, 2019.

Artem Shelmanov, Evgenii Tsymbalov, Dmitri Puzyrev, Kirill Fedyanin, Alexander Panchenko, and Maxim Panov. How certain is your Transformer? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1833–1840, Online, April 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.eacl-main.157. URL https://aclanthology.org/2021.eacl-main.157.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi S. Jaakkola. Style transfer from non-parallel text by cross-alignment. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6830–6841, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/2d2c8394e31101a261abf1784302bf75-Abstract.html.

Zheyuan Ryan Shi, Claire Wang, and Fei Fang. Artificial intelligence for social good: A survey. *CoRR*, abs/2001.01818, 2020. URL http://arxiv.org/abs/2001.01818.

Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. Fakenewsnet: A data repository with news content, social context and spatialtemporal information for studying fake news on social media. *arXiv preprint ArXiv:1809.01286*, 2018.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 395–405, 2019a.

Kai Shu, Suhang Wang, and Huan Liu. Beyond news contents: The role of social context for fake news detection. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 312–320, 2019b.

Kai Shu, Xinyi Zhou, Suhang Wang, Reza Zafarani, and Huan Liu. The role of user profiles for fake news detection. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 436–439, 2019c.

Craig Silverman. Emergent: A real-time rumor tracker. *Online: http://www. emergent. info/. Accessed*, pages 12–13, 2017.

Maneet Singh, Rishemjit Kaur, and S. R. S. Iyengar. Multidimensional analysis of fake news spreaders on twitter. In Sriram Chellappan, Kim-Kwang Raymond Choo, and NhatHai Phan, editors, *Computational Data and Social Networks - 9th International Conference, CSoNet 2020, Dallas, TX, USA, December 11-13, 2020, Proceedings*, volume 12575 of *Lecture Notes in Computer Science*, pages 354–365. Springer, 2020. doi:10.1007/978-3-030-66046-8_29. URL https://doi.org/10.1007/978-3-030-66046-8_29.

Sugan Sirihattasak, Mamoru Komachi, and Hiroshi Ishikawa. Annotation and classification of toxicity for thai twitter. In *TA-COS 2018: 2nd Workshop on Text Analytics for Cybersecurity and Online Safety*, page 1, 2018.

Amir Soleimani, Christof Monz, and Marcel Worring. Bert for evidence retrieval and claim verification. In *European Conference on Information Retrieval*, pages 359–366. Springer, 2020.

Magda Stroińska. Toxic language of contempt. *Warsaw East European*, page 79, 2020.

Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. "transforming" delete, retrieve, generate approach for controlled text style transfer. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3267–3277. Association for Computational Linguistics, 2019. doi:10.18653/v1/D19-1322. URL https://doi.org/10.18653/v1/D19-1322.

Raymond Hendy Susanto, Shamil Chollampatt, and Liling Tan. Lexically constrained neural machine translation with Levenshtein transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3536–3543, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.325. URL https://aclanthology.org/2020.acl-main.325.

Daniel P Sutherlin, Linda Bao, Megan Berry, Georgette Castanedo, Irina Chuckowree, Jenna Dotson, Adrian Folks, Lori Friedman, Richard Goldsmith, Janet Gunzner, et al. Discovery of a potent, selective, and orally available class i phosphatidylinositol 3-kinase (pi3k)/mammalian target of rapamycin (mtor) kinase inhibitor (gdc-0980) for the treatment of cancer. *Journal of medicinal chemistry*, 54(21):7579–7587, 2011.

Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. Multilingual neural machine translation with knowledge distillation. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=S1gUsoR9YX.

Edson C Tandoc Jr, Richard Ling, Oscar Westlund, Andrew Duffy, Debbie Goh, and Lim Zheng Wei. Audiences' acts of authentication in the age of fake news: A conceptual framework. *New Media & Society*, 20(8):2745–2763, 2018.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. Multilingual translation with extensible multilingual pretraining and finetuning. 2020.

Serra Sinem Tekiroglu, Yi-Ling Chung, and Marco Guerini. Generating counter narratives against online hate speech: Data and strategies. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1177–1190. Association for Computational Linguistics, 2020. doi:10.18653/v1/2020.acl-main.110. URL https://doi.org/10.18653/v1/2020.acl-main.110.

Daryna Dementieva and Alexander Panchenko. Cross-lingual evidence improves monolingual fake news detection. In Jad Kabbara, Haitao Lin, Amandalynne

Paullada, and Jannis Vamvas, editors, *Proceedings of the ACL-IJCNLP 2021 Student Research Workshop, ACL 2021, Online, JUli 5-10, 2021*, pages 310–320. Association for Computational Linguistics, 2021. doi:10.18653/v1/2021.acl-srw.32. URL https://doi.org/10.18653/v1/2021.acl-srw.32.

Daryna Dementieva and Alexander Panchenko. Fake news detection using multilingual evidence. In Geoffrey I. Webb, Zhongfei Zhang, Vincent S. Tseng, Graham Williams, Michalis Vlachos, and Longbing Cao, editors, *7th IEEE International Conference on Data Science and Advanced Analytics, DSAA 2020, Sydney, Australia, October 6-9, 2020*, pages 775–776. IEEE, 2020. URL https://doi.org/10.1109/DSAA49011.2020.00111.

Daryna Dementieva, Igor Markov, and Alexander Panchenko. SkoltechNLP at SemEval-2020 task 11: Exploring unsupervised text augmentation for propaganda detection. In Aurélie Herbelot, Xiaodan Zhu, Alexis Palmer, Nathan Schneider, Jonathan May, and Ekaterina Shutova, editors, *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 1786–1792. International Committee for Computational Linguistics, 2020. URL https://doi.org/10.18653/v1/2020.semeval-1.234.

Daryna Dementieva, Daniil Moskovskiy, Varvara Logacheva, David Dale, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. Methods for detoxification of texts for the russian language. *Multimodal Technol. Interact.*, 5(9):54, 2021. URL https://doi.org/10.3390/mti5090054.

Daryna Dementieva, Varvara Logacheva, Irina Nikishina, Alena Fenogenova, David Dale, Irina Krotova, Nikita Semenov, Tatiana Shavrina, and Alexander Panchenko. RUSSE-2022: Findings of the first Russian detoxification task based on parallel corpora. In *Computational Linguistics and Intellectual Technologies*, 2022.

Daryna Dementieva, Daniil Moskovskiy, Varvara Logacheva, David Dale, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. Methods for detoxification of texts for the russian language. In *Computational Linguistics and Intellectual Technologies*, 2021. URL https://www.dialog-21.ru/media/5503/dementievadplusetal046.pdf.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi:10.18653/v1/N18-1074.

Alexey Tikhonov and Ivan P. Yamshchikov. What is wrong with style transfer for texts? *CoRR*, abs/1808.04365, 2018.

Nenad Tomašev, Julien Cornebise, Frank Hutter, Shakir Mohamed, Angela Picciariello, Bec Connelly, Danielle Belgrave, Daphne Ezer, Fanny Cachat van der Haert, Frank Mugisha, et al. Ai for social good: unlocking the opportunity for positive impact. *Nature Communications*, 11(1):1–6, 2020.

Minh Tran, Yipeng Zhang, and Mohammad Soleymani. Towards a friendly online community: An unsupervised style transfer framework for profanity redaction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2107–2114, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi:10.18653/v1/2020.coling-main.190.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan, October–November 2019. Association for Computational Linguistics. doi:10.18653/v1/W19-8643. URL https://aclanthology.org/W19-8643.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Inna Vogel and Peter Jiang. Fake news detection with the new german dataset "germanfakenc". In *International Conference on Theory and Practice of Digital Libraries*, pages 288–295. Springer, 2019.

Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, sep 2014. ISSN 0001-0782. doi:10.1145/2629489. URL https://doi.org/10.1145/2629489.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 3261–3275, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/4496bf24afe7fab6f046bf4923da8de6-Abstract.html.

William Yang Wang. " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*, 2017.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. Neural network acceptability judgments. *Trans. Assoc. Comput. Linguistics*, 7:625–641, 2019. URL https://transacl.org/ojs/index.php/tacl/article/view/1710.

Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June 2016. Association for Computational Linguistics. doi:10.18653/v1/N16-2013. URL https://aclanthology.org/N16-2013.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. Overview of the germeval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*, Vienna, Austria, September 2018.

John Wieting and Kevin Gimpel. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi:10.18653/v1/P18-1042. URL https://aclanthology.org/P18-1042.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. Beyond BLEU:training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355, Florence, Italy, July 2019. Association for Computational Linguistics. doi:10.18653/v1/P19-1427. URL https://aclanthology.org/P19-1427.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi 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 Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

Xing Wu, Shangwen Lv, Liangjun Zang, Jizhong Han, and Songlin Hu. Conditional BERT contextual augmentation. In João M. F. Rodrigues, Pedro J. S. Cardoso, Jânio M. Monteiro, Roberto Lam, Valeria V. Krzhizhanovskaya, Michael Harold Lees, Jack J. Dongarra, and Peter M. A. Sloot, editors, *Computational Science - ICCS 2019 - 19th International Conference, Faro, Portugal, June 12-14, 2019, Proceedings, Part IV*, volume 11539 of *Lecture Notes in Computer Science*, pages 84–95. Springer, 2019a. doi:10.1007/978-3-030-22747-0_7.

Xing Wu, Tao Zhang, Liangjun Zang, Jizhong Han, and Songlin Hu. Mask and infill: Applying masked language model for sentiment transfer. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5271–5277. ijcai.org, 2019b. doi:10.24963/ijcai.2019/732. URL https://doi.org/10.24963/ijcai.2019/732.

Jingjing Xu, Xu Sun, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 979–988, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi:10.18653/v1/P18-1090. URL https://aclanthology.org/P18-1090.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 483–498. Association for Computational Linguistics, 2021. doi:10.18653/v1/2021.naacl-main.41. URL https://doi.org/10.18653/v1/2021.naacl-main.41.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. Predicting the type and target of offensive posts in social media. 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 1415–1420, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi:10.18653/v1/N19-1144. URL https://aclanthology.org/N19-1144.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020a. URL https://openreview.net/forum?id=SkeHuCVFDr.

Yi Zhang, Tao Ge, and Xu Sun. Parallel data augmentation for formality style transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3221–3228, Online, July 2020b. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.294. URL https://aclanthology.org/2020.acl-main.294.

Wenqing Zhao. Misinformation correction across social media platforms. In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1371–1376. IEEE, 2019.

Zilong Zhao, Jichang Zhao, Yukie Sano, Orr Levy, Hideki Takayasu, Misako Takayasu, Daqing Li, Junjie Wu, and Shlomo Havlin. Fake news propagates differently from real news even at early stages of spreading. *EPJ Data Science*, 9 (1):7, 2020.

Chulun Zhou, Liangyu Chen, Jiachen Liu, Xinyan Xiao, Jinsong Su, Sheng Guo, and Hua Wu. Exploring contextual word-level style relevance for unsupervised

style transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7135–7144, Online, July 2020a. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.639. URL https://aclanthology.org/2020.acl-main.639.

Xinyi Zhou, Apurva Mulay, Emilio Ferrara, and Reza Zafarani. Recovery: A multimodal repository for covid-19 news credibility research. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3205–3212, 2020b.

Wanrong Zhu, Xin Wang, Tsu-Jui Fu, An Yan, Pradyumna Narayana, Kazoo Sone, Sugato Basu, and William Yang Wang. Multimodal text style transfer for outdoor vision-and-language navigation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1207–1221, Online, April 2021. Association for Computational Linguistics. doi:10.18653/v1/2021.eacl-main.103. URL https://aclanthology.org/2021.eacl-main.103.

Caleb Ziems, Minzhi Li, Anthony Zhang, and Diyi Yang. Inducing positive perspectives with text reframing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3682–3700, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi:10.18653/v1/2022.acl-long.257. URL https://aclanthology.org/2022.acl-long.257.

Appendix A

Fake News Supplementary

A.1 Feature Importance for Fake New Classification method

In this section, we provide the illustration of feature importance for fake news classification model for: i) FakeNewsAMT dataset (Figure A-1); ii) Celebrity dataset (Figure A-3); iii) ReCOVery dataset (Figure A-3). The notation for CE feature designation: <language of news>_<its position in search results>_<content similarity feature (sim)> or <source rank feature (rank)>. We can see that cross-lingual evidence features (both similarities and ranks) are at the top for all datasets.

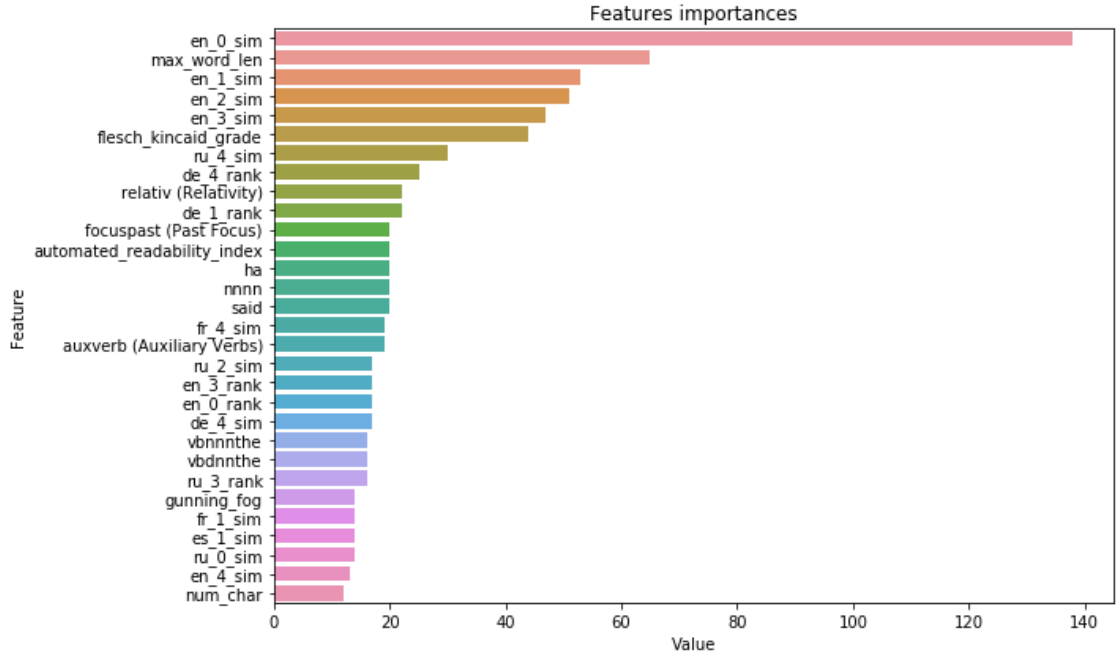


Figure A-1: Top 30 features importances of the best model for FakeNewsAMT dataset: LightGBM model based on All linguistic + CE Emb. + Rank feature set.

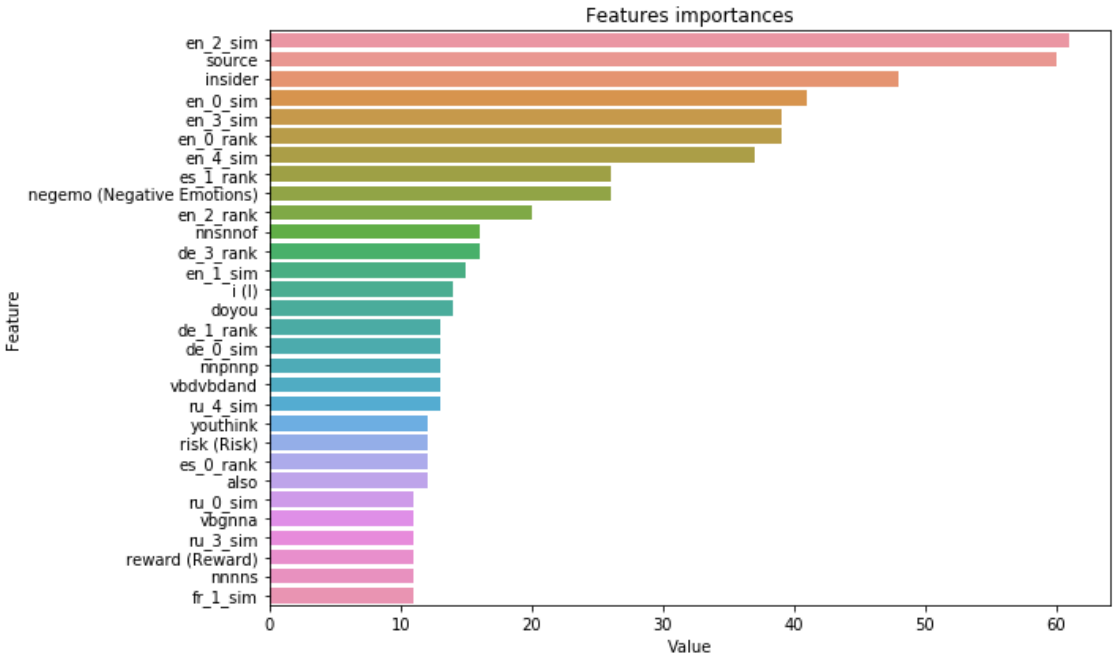


Figure A-2: Top 30 features importances of the best model for Celebrity dataset: LightGBM model based on All linguistic + CE Emb. + Rank feature set.

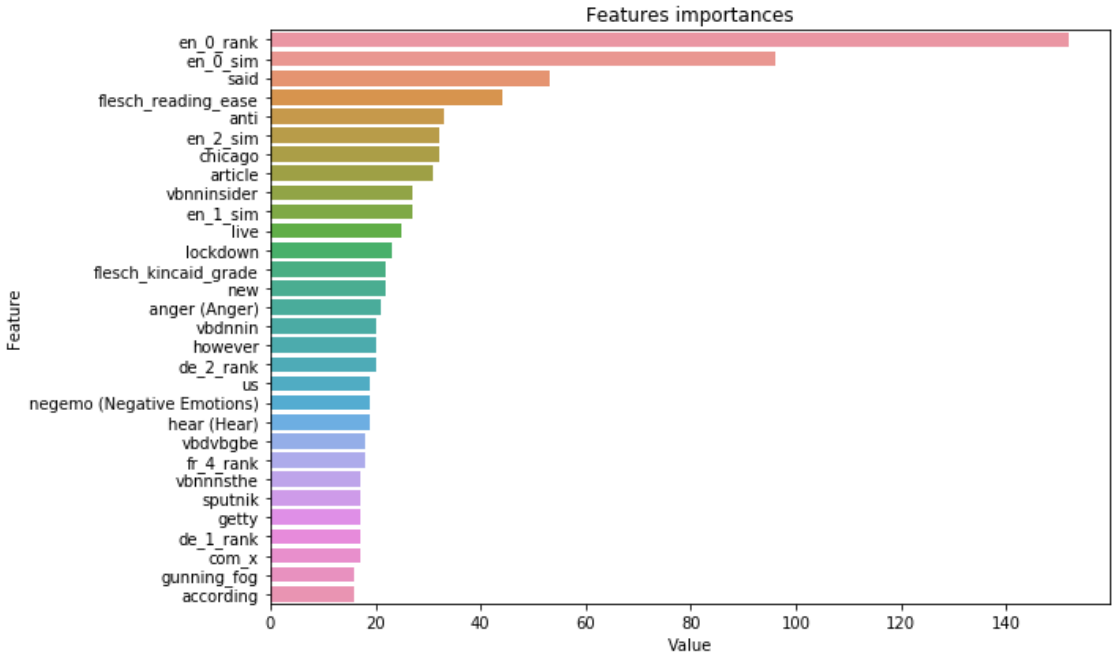


Figure A-3: Top 30 features importances of the best model for ReCOVery dataset: LightGBM model based on Ngrams + CE Emb. + Rank feature set.

A.2 Mutliverse usage: Real-Case Example

Here we provide examples of how the proposed Mutliverse approach for cross-lingual evidence feature extraction can be used for the explanation of fake news classification model decision explanation. In Table A.1, we provide an example which cross-lingual evidence is extracted for **fake** news. We can observe that there is no supportive information and even refutation. On the contrary, for **legit** news we can observe a lot of supportive information all across different media.

Title	English translation	Source rank ↓	Similarity score ↑
Original news (FAKE)			
Lottery winner arrested for dumping $200,000 of manure on ex-boss' lawn	–	–	–
English search results			
PolitiFact - Viral post that lottery winner was arrested for dumping manure on former boss' lawn reeks of falsity	–	15947	0.00
Was a Lottery Winner Arrested for Dumping $200,000 of Manure on the Lawn of His Former Boss?	–	5798	0.00
Lottery winner arrested for dumping $200,000 of manure on ex-boss' lawn	–	314849	0.89
French search results			
Un gagnant de loterie arrêté pour avoir déversé 200 000$ de fumier sur la pelouse de son ex-patron | Africa24.info	Lottery winner arrested for dumping $ 200,000 in manure on expatron's lawn Africa24info	2595725	0.78
Fertiliser le jardin	Fertilize the garden	193218	0.43
Histoire de Suresnes — Wikipedia	History of Suresnes – Wikipedia	13	0.31
German search results			
Mit "Scream"-Maske zum Millionen-Jackpot: Lottogewinner will anonym bleiben - aber er übersieht eine wichtige Sache	With a Scream mask for the millionaire jackpot lottery winner, he wants to remain anonymous but he overlooks an important thing	15294	0.55
Lotto-Gewinner holt Mega-Jackpot und lässt 291 Millionen Dollar sausen	Lottery winner takes MegaJackpot and drops $ 291 million	15294	0.58
Hesse knackt Sechs-Millionen-Jackpot: Noch hat sich der Gewinner nicht gemeldet	Hesse cracks six million jackpot The winner has not yet announced	44799	0.57
Spanish search results			
Ganador de 125 millones en la lotería arrestado por vaciar camiones de heces en casa de su jefe	125 million lottery winner arrested for dumping trucks of feces at his boss's home	922337	0.76
Le toca la lotería y compra 20.000 toneladas de estiércol para arrojar en el porche de su jefe	He wins the lottery and buys 20,000 tons of manure to dump on his boss's porch	149185	0.77
Estas son las 50 noticias falsas que tuvieron mayor éxito en Facebook en 2018	These are the 50 fake news that had the most success on Facebook in 2018	405	0.00
Russian search results			
ПОБЕДИТЕЛЬ ЛОТЕРЕИ АРЕСТОВАН ЗА ТО, ЧТО ПОТРАТИЛ $200.000, ЧТОБЫ СВАЛИТЬ ГОРУ НАВОЗА НА ГАЗОН / победитель :: смешные картинки (фото приколы) :: новости	LOTTERY WINNER ARRESTED FOR SPENDING $ 200,000 TO DUMP MOUNT OF MANURE ON THE LAWN / winner :: funny pictures (funny photos) :: news	15418	0.76
ПОБЕДИТЕЛЬ ЛОТЕРЕИ АРЕСТОВАН ЗА ТО, ЧТО ПОТРАТИЛ $200.000, ЧТОБЫ СВАЛИТЬ ГОРУ НАВОЗА НА ГАЗОН СВОЕГО БЫВШЕГО БОССА ПО НЕМУ ВИДНО, ЧТО ОНО ТОГО СТОИЛО...	LOTTERY WINNER ARRESTED FOR SPENDING $ 200,000 TO DUMP A MOUNTAIN OF MANURE ON THE LAW OF HIS FORMER BOSS ONE SEE THAT IT WAS WORTH ...	146662	0.70
Победитель лотереи потратил выигрыш, убойно отомстив бывшему боссу	Lottery Winner Wasted Winning In Hellful Revenge On Ex-Boss	146662	0.83

Table A.1: The example of work of the proposed approach for fake and legit news. For each target language (English, French, German, Spanish, Russian) search results are presented: titles of top 3 news. For every non-Enlgish title the English translation is provided. Each piece of scraped news is rated with the rank of its source and content similarity to the original news based on text embedding. The larger↑ (or lower↓) score, the better. For **fake news** the search results either come from unreliable sources or provide no relevant information to the original news.

Title	English translation	Source rank ↓	Similarity score ↑
Original news (LEGIT)			
В Монголии произошла вспышка бубонной чумы: https://hightech.fm/2020/07/02/plague-outbreak	Bubonic plague outbreak in Mongolia		
English search results			
Bubonic plague: Case found in China's Inner Mongolia - CNN	–	91	0.88
Teenager dies of Black Death in Mongolia	–	178	0.72
China bubonic plague: Inner Mongolia takes precautions after case	–	101	0.69
French search results			
Epidémie : des cas de peste détectés en Chine et en Mongolie	Epidemic: cases of plague detected in China and Mongolia	284	0.73
Craintes d'une épidémie de peste bubonique? Un adolescent de 15 ans est la première victime recensée en Mongolie	Fear of a bubonic plague epidemic? A 15-year-old is the first victim in Mongolia	496	0.70
Chine : Un cas de peste bubonique détecté en Mongolie intérieure	China: Bubonic plague case detected in Inner Mongolia	5003	0.84
German search results			
Mongolei: 15-Jähriger an Beulenpest gestorben - DER SPIEGEL	Mongolia: 15-year-old died of bubonic plague - DER SPIEGEL	928	0.78
Beulenpest - Was über die Pest-Fälle in China bekannt	Bubonic plague - what is known about the plague cases in China	6234	0.75
Bringen Murmeltiere die Pest zurück? Mongolei warnt vor Tier-Kontakt	Will marmots bring the plague back? Mongolia warns of animal contact	48864	0.61
Spanish search results			
BROTE DE PESTE BUBÓNICA EN MONGOLIA	BUBONIC PLAGUE OUTBREAK IN MONGOLIA	436	0.84
Brote de peste negra provoca cuarentena en Mongola	Black plague outbreak causes quarantine in Mongolia	4417	0.78
Brote de peste negra alarma en Mongolia y cierra frontera con Rusia	Black plague outbreak alarms Mongolia, closes border with Russia	453	0.63
Russian search results			
В Монголии произошла вспышка бубонной чумы ... - Гордон	There was an outbreak of bubonic plague in Mongolia ... - Gordon	21372	0.91
В Монголии произошла вспышка бубонной чумы - Урал56.Ру	Bubonic plague outbreak in Mongolia - Ural56.Ru	124712	0.92
Возвращение «Черной смерти»: главное о вспышке бубонной чумы в Монголии	Return of the "Black Death": the main thing about the outbreak of the bubonic plague in Mongolia	8425	0.87

Table A.2: The example of work of the proposed approach for fake and legit news. For each target language (English, French, German, Spanish, Russian) search results are presented: titles of top 3 news. For every non-Enlgish title the English translation is provided. Each piece of scraped news is rated with the rank of its source and content similarity to the original news based on text embedding. The larger↑ (or lower↓) score, the better. For **legit news** the search results across different languages are strongly related to the original news.

A.3 Multilingual News Similarity: NER-based approach Performance Example

Here we present the result of NER extraction for the approach for multilingual news similarly described in Section 5.5. We can observe that the method fails to correlate with manual annotations because of not precise named entities extraction.

pair_id	NER 1	NER 2	dist LOC	dist PER	dist ORG	Predict.	Ground true
1484012638 1483801741	**LOC**: Baku, Azerbaijan, Shamakhi, Ismayilli, Aghsu **PER**: Ilham Aliyev **ORG**: _	**LOC**: Azerbaijan, Baku **PER**: Ilham Aliyev **ORG**: _	0.148	0.000	0.500	2.959	2.500
1483806302 1483770632	**LOC**: Atlanta, GA, Washington, D. C., Capitol Hill, BarackO, America, Georgia, New Jersey **PER**: John Lewis, Lewis, RepJohnLewis, Barack Obama, God, Stacey Abrams, Cory Booker, Jim Crow, Mark Hamill **ORG**: Ku Klux Klan	**LOC**: America, Mississippi Delta, Edmund Pettus Bridge, Georgia **PER**: John Lewis, Peniel Joseph, Jim Crow, Barbara Jordan, Peniel Joseph, Lewis, Crow, Donald Trump, **ORG**: Center for the Study of Race and Democracy, CNN, LBJ School of Public Affairs, University of Texas at Austin	0.078	0.071	0.971	2.492	1.000
1546012672 1488866568	**LOC**: Dresden, Chemnitz **PER**: _ **ORG**: Staatsanwaltschaft	**LOC**: Dresden **PER**: Carolyn, Carolyn Anne Cavender **ORG**: Jackson Madison, General Hospital	0.471	0.500	0.998	3.360	4.000

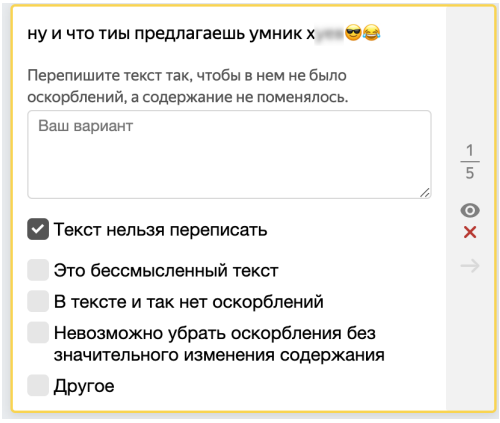
Table A.3: Example of performance of the best NER model. (BERT-based NER extractor, BERT embeddings, Gradient Boosting model).

Appendix B

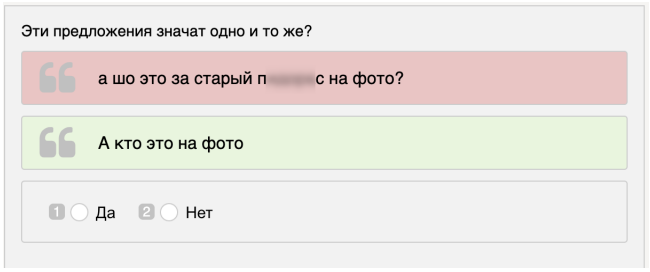
Detoxification Supplementary

B.1 ParaDetox: Labeling Pipeline Instructions for Russian

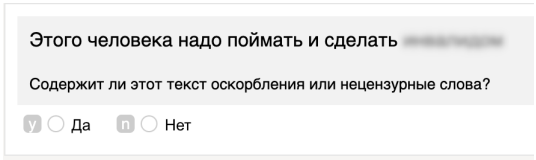
This appendix contains the illustration of all labeling tasks at Yandex Toloka platform in original Russian language: (i) detoxificated paraphrase generation (Figure B-1a); (ii) content preservation check (Figure B-1b); (iii) toxicity classification (Figure B-1c); (iv) fluency check (Figure B-1d).



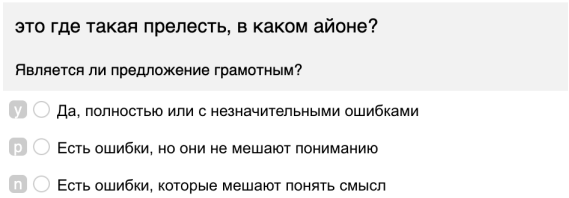
(**a**) Generation of paraphrases.



(**b**) Content preservation check.



(**c**) Toxicity classification.



(**d**) Fluency check.

Figure B-1: Original Russian interfaces in Yandex Toloka platform for labeling.

B.2 ParaDetox: Instructions and Training examples for Crowdsourcing Tasks (English)

In this section, we provide detailed instructions that are visible for crowd workers at the beginning of the tasks for the English ParaDetox collection as well as additional examples from training pools.

B.2.1 Task 1: Paraphrase Generation

Instruction You are given a text which **probably** contains offences or swear words. You should rewrite it to keep its meaning and get rid of offence. If the text is not offensive, tick the box "I can't rewrite the text" and choose the reason from the list.

The project uses deferred quality control of assignments. The completed tasks will be checked and paid for within ten days. We can reject the completed task if:

- You don't rewrite a text which can be rewritten;
- Your text does not match the meaning of the original text;
- Your text is a copy of the original text;
- Your text contains offence or swear words.

Control Examples During Task 1, we control that crowd workers do not skip the tasks with samples that we definitely know can be rewritten. The examples of such samples are provided in Table B.1.

B.2.2 Task 2: Content Preservation Check

Instruction You will be shown two sentences. Your task is to indicate whether they mean the same (or close) thing or not. The **sentences don't have to be identical** - one of them can be offensive and the other one say the same thing in a neutral tone.

Input Text	**Can be detoxified**
*good, i'm tired of seeing people abusing that s**t.*	✔
*they re both f**ked up.*	✔
*dont be a d**k and add st**id and clear vandalism to articles.*	✔
*either way you sound f**lish and ill informed.*	✔
*who the f**k cares whose fault it is?*	✔
*anchorage is becoming a c**p town.*	✔
*f**k off if you are going to do s**t like that.*	✔
*why is reddit so f**king biased?*	✔
*this sick f**k is just a s**iopath who doesn't want to face the consequences for his actions.*	✔

Table B.1: Task 1 (Paraphrase Generation, English) control tasks showing which texts can be definitely detoxified.

If one or both sentences contain nonsense (non-words, senseless chains of words, etc.), choose the option "No".

Training Examples For Task 2, we used a training pool to provide the crowd workers understanding of which content preservation in texts we expect. Afterward, the annotators took examinations to be accepted to do a markup of real samples. Also, we had several (2-3) on-page training samples that helped the annotators not to lose the sense of a task if, for instance, they paused a markup process. We provide examples of training samples that we showed the annotators in Table B.2.

Input Text 1	**Input Text 2**	**Match**	**Hint**
*you don t think countries with dicktators experience the same s**t?*	*you don't think countries with dictators experience the same?*	✔	The second sentence doesn't have the word "s**t", which doesn't change the sense.
*this is the d**best idea i have heard the conservatives and ndp come up with.*	*this is very bad idea that conservatives and ndp have come up with.*	✔	The phrase "the d**best" is replaced with "very bad", which doesn't change the sense.
*what a life a this l**er leads.*	*what a life a this man leading.*	✔	In the second sentence the word "l**er" is replaced with a neutral synonym "man".
*keep your f**thy paws off wikipedias articles unless you have something worthwhile to contribute.*	*I do not think these articles are serious enough to be shared.*	✘	The sentences have different meanings: the first sentence states that the edits of articles that the opponent makes are not worthy, the second one states that the opponent's articles are not worthy.
*i hate hearing about this s**t where they pr*secute minors as adults.*	*I dont like hearing about this.*	✘	The second sentence doesn't tell what exactly the speaker hates hearing about.
*a vexatious , willfully *gnorant pov pusher.*	*Living in dark on will.*	✘	The sentences have nothing in common.

Table B.2: Task 2 (Content Preservation Check, English) examples used to provide understanding of content preservation requirement to crowd workers.

B.2.3 Task 3: Toxicity Check

Instruction You should read the sentences and determine if it is offensive or not.

Offensive texts are the ones containing insults, threats, swear words.

Non-offensive texts can contain criticism and be negative (but not insulting) towards the conversation partner.

Training Examples We used the same design for control as for Task 2. Here we provide examples of training samples in Table B.3.

Input Text	**Is toxic**	**Hint**
*They are l**ers.*	✔	The text offends some group of people.
poison the water, so there is no hope of survival.	✔	The text incites to cause harm.
Sad pathetic and liable with law	✔	The text gives negative feedback in an unacceptable way.
*if you ask a st**id question, dont complain about the answer.*	✔	The message contains aggression and implies that the opponent is stupid.
The people running the city of eugene is not doing their work correctly.	✘	Non-offensive criticism of the authorities.
This is mine, you don't have to be worried.	✘	Neutral statement.
ive been at it as a value added farmer for years in this state.	✘	The message contains no offence or swear words.
Either way you sound uninformed.	✘	The message contains non-offensive criticism.

Table B.3: Task 3 (Toxicity Check, English) examples used to provide understanding of toxic style to crowd workers.

B.3 ParaDetox: Instructions and Training examples for Crowdsourcing Tasks (Russian)

In this section, we provide detailed instructions that are visible for crowd workers at the beginning of the tasks for the Russian ParaDetox collection as well as additional examples from training pools.

B.3.1 Task 1: Paraphrase Generation

Instruction Вам будет показан текст, который, возможно, содержит оскорбления или грубые выражения. Вам требуется переписать его так, чтобы сохранить содержание и избавиться от оскорблений. Если текст не оскорбительный и не грубый, выберите опцию "Текст нельзя переписать"и укажите причину.

В проекте используется отложенная приемка заданий. Выполненные задания будут проверены и оплачены не более, чем через 10 дней. Задание не будет оплачено, если:

- Вы не переписали текст, который можно переписать;
- Смысл переписанного текста слишком сильно отличается от оригинала;
- Переписанный текст - скопированный оригинальный текст;
- Переписанный текст все равно содержит оскорбления и грубые выражения.

Примеры, как можно переписать текст и какие способы переписывания считаются некорректными, приведены в Таблице B.4.

Control Examples As for English ParaDetox, during Russian Task 1, we did not specifically provide training examples, but we control that crowd workers do not skip the tasks with samples that we definitely know can be rewritten. The examples of such samples are provided in Table B.5.

Input Text	**Paraphrase**	**Hint**
*какой то ос*л написал.*	*Какой-то человек написал.*	Good paraphrase ✔
*из-за таких п***ров мы и страдаем*	*Из-за таких людей мы и страдаем*	Good paraphrase ✔
*порядок бы б***ь навёл!*	*Порядок бы навел*	Good paraphrase ✔
*этого х**ря на кол посадить надо.*	*я не согласен с его мнением.*	Bad paraphrase ✘: Сильное изменение смысла
*какая же ты м**зь, кузнецов, п***р и м**зь!*	*какой же вы нехороший человек, кузнецов, и м**зь!*	Bad paraphrase ✘: Сохранение оскорбления
*автор, пошел ты на**й!*	*ждцоуа длыоваа!!!!!!*	Bad paraphrase ✘: Бессмысленный текст

Table B.4: Task 1 (Paraphrase Generation, Russian) examples used to provide understanding of style change requirement to crowd workers.

Input Text	**Can be detoxified**
*б***ь! это кому-то же нравится! п****ц! три **ла п****противных!*	✔
*с**и б***и издеваются над животным*	✔
*что за бред, ***утые.*	✔
*ну и что ты предлагаешь умник х****	✔
*Д**ильный скот уже час не унимается со своими фейерверками.*	✔
*накосячил будь мужиком умей отвечать за свои поступки не будь п**орасом!*	✔

Table B.5: Task 1 (Paraphrase Generation, Russian) control tasks showing which texts can be definitely detoxified.

B.3.2 Task 2: Content Preservation Check

Instruction Вы увидите два предложения. Ваша задача состоит в том, чтобы определить, значат ли они одно и то же. Предложения не должны быть абсолютно идентичным по смыслу - одно из них может быть оскорбительным, а другое содержать ту же информацию в нейтральном тоне.

Если одно из предложений или оба предложения бессмысленны или содержат бессмысленные слова/фразы затрудняющие понимания, выберите ответ "Нет".

Training Examples For Task 2, we used a training pool to provide the crowd workers understanding of which content preservation in texts we expect. Afterward, the annotators took examinations to be accepted to do a markup of real samples. Also, we had several (2-3) on-page training samples that helped the annotators not to lose the sense of a task if, for instance, they paused a markup process. We provide the examples of training samples that we showed the annotators in Table B.6.

B.3.3 Task 3: Toxicity Check

Instruction Вам требуется прочесть предложения и определить, содержат ли они оскорбления или нецензурные и грубые слова. **Внимание!** Неоскорбительное предложение может содержать критику и быть негативно окрашенным.

Training Examples We used the same design for control as for Task 2. Here we provide examples of training samples in Table B.7.

Input Text 1	**Input Text 2**	**Match**	**Hint**
*ты и правда **анутая картинками только и общаешься*	*Ты картинками только и общаешься*	✔	Удалено только оскорбление ("ты и поавда **анутая"), смысл сохранён.
*это д**ил. не трудитесь....вы не достучитесь*	*не трудитесь....вы не достучитесь*	✔	Нецензурное "это д**ил."удалено, смысл сохранен
*в глаза хочу взглянуть я тем с**ам, голосующим за эту власть!!!*	*Хочу посмотреть в глаза тем, кто голосует за эту власть.*	✔	Удалено оскорбление ("с**ам"), смысл сохранён.
****ище ты зря веришь сми в р**ке*	*Ты зря веришь СМИ.*	✘	Во втором предложении нет упоминания, что речь именно о СМИ в определенной. Потеряна часть смысла.
это тебя нужно и таких как ты стрелять за подобные высказывание	*Мне не нравятся люди, которые выражают подобные идеи.*	✘	Смысл предложений не совпадает
расстреливать, без суда и следствия.	*;aliwjefm wekj*	✘	Второе предложение это просто набор символов

Table B.6: Task 2 (Content Preservation Check, Russian) examples used to provide understanding of content preservation requirement to crowd workers.

Input Text	**Is toxic**	**Hint**
*За я**а тебя надо подвесить за дочь.*	✔	Призыв к насилию ("за я**а тебя надо подвесить")
Сжечь живём этого человека	✔	Призыв к насилию ("сжечь живьем")
*Интернетная ш**на, не тебе меня поучать*	✔	Наличие оскорбления ("ш**на"
*Вообще не вижу смысла спорить по поводу дюймов, х***ый там ips или нет, машина не этим интересна.*	✔	Наличие матерной лексики ("х***ый"
Смотреть на них не приятно.	✘	Мнение выражено в корректной форме.
это святые птицы.нашел в кого стрелять.	✘	Не содержит оскорблений, мата или призывов к насилию.
Не понимаешь не говори про продовольствие	✘	Не содержит оскорблений, мата или призывов к насилию.
Из-за таких людей страдают невинные	✘	Не содержит оскорблений, мата или призывов к насилию.

Table B.7: Task 3 (Toxicity Check, Russian) examples used to provide understanding of toxic style to crowd workers.

B.4 ParaDetox Samples

Here we provide additional examples of manually detoxified parallel sentences from the English ParaDetox corpus (Table 7.5) and Russian ParaDetox corpus (Table 7.8).

B.4.1 English ParaDetox Samples

Original	my computer is broken and my phone too!! wt* is this devil s**t???
Paraphrases	My computer is broken and my phone too! So disappointed! My computer is broken and my phone too, what is this? Both my computer and phone are broken.
Original	s**t is crazy around here.
Paraphrases	It is crazy around here. Stuff is crazy around here. Something is crazy around here.
Original	delete the page and sh*t up
Paraphrases	Delete the page and stay silent. please delete the page delete the page
Original	massive and sustained public pressure is the only way to get these b**tards to act.
Paraphrases	Massive and sustained public pressure is the only way to get them to act. massive and sustained preasure is the only way to get these people to act
Original	f**k you taking credit for some s**t i wanted to do
Paraphrases	You are taking credit for something I wanted to do You're taking credit fro something i wanted to do.
Original	you gotta admit that was f**kin hilarious though!
Paraphrases	you got to admit that was very hilarious though! you gotta admit that was hilarious though!

Table B.8: Examples of detoxified sentences from the collected English ParaDetox.

B.4.2 Russian ParaDetox Samples

Original	из-за таких п***ров мы и страдаем *(we suffer because of such f**gots)*
Paraphrases	из-за таких плохих людей мы и страдаем *(we suffer because of such bad people)* Из-за таких людей мы и страдаем *(we suffer because of such people)* из за таких как он мы и страдаем *(we suffer because of people like him)*
Original	лучше б руководство выкинули бы на х**н!!!! *(it would be better if the leaders were thrown out to h**l!!!)*
Paraphrases	лучше б руководство выкинули бы *(it would be better if the leaders were thrown out)* Лучше бы руководство убрали *(it would be better if the leaders were removed)* Лучше б руководство поменяли бы!!! *(it would be better if the leaders were changed!!!)*
Original	да вообще по**й сколько этот д****еб получает *(I don't f**king care how much this motherf***er gets paid)*
Paraphrases	да вообще все равно сколько он получает *(I don't much care how much he gets paid)* Мне всё равно сколько он получает. *(I don't care how much he gets paid)*
Original	б***ь, развели срачь из ни***. *(d**n, they made a f**king quarrel out of nothing)*
Paraphrases	Развели ссору из ничего. *(they made a quarrel out of nothing)* Развели ругань из нечего *(they made a fight out of nothing)*

Table B.9: Examples of detoxified sentences from the collected Russian ParaDetox.

B.5 Outputs of Detoxification Models

This section provides additional examples of detoxification performance of considered baselines, proposed English and Russian detoxification models, and multilingual detoxification models.

B.5.1 English Detoxification Examples

Table B.10 contains the outputs of four well-performing detoxification models: Delete model which deterministically removes swear words, unsupervised ParaGeDi and CondBERT models, and BART model fine-tuned on our parallel ParaDetox dataset.

Original	at least one party gives a c**p about humans after birth.
Delete	at least one party gives a about humans after birth.
CondBERT	at least one party gives a whole story about humans after birth.
ParaGeDi	At the same time, at least one party doesn't believe in human life after birth
BART-ParaDetox	**at least one party cares about humans after birth.**
Original	gosh, discrimination is such a b***h!
Delete	gosh, discrimination is such a!
CondBERT	gosh, discrimination is such a thing!
ParaGeDi	Gosh, discrimination is a bit of a bad thing!
BART-ParaDetox	**Discrimination is bad!**
Original	he ll f**k that up too.
Delete	he ll that up too.
CondBERT	he ll ' ve picked that up too.
ParaGeDi	He'll get that up.
BART-ParaDetox	**He will mess that up too.**
Original	whats the title of this report, no s**t?
Delete	whats the title of this report, no?
CondBERT	whats the title of this report , no stuff?
ParaGeDi	What's the name of the report, no?
BART-ParaDetox	**whats the title of this report?**

Table B.10: Examples of English detoxifications by different models. Bad answers are shown in red, the best answers **in bold**.

B.5.2 Russian Detoxification Examples

Table B.11 contains the outputs of the baselines and our fine-tuned seq2seq models: Delete model which deterministically removes swear words, unsupervised CondBERT models, and ruGPT and ruT5 models fine-tuned on our parallel ParaDetox dataset.

Original	иди на**р, верните иваныча, черти!
Delete	иди , верните иваныча, !
CondBERT	иди на , верните иваныча , че !
ruGPT3-small	Верните ивановича
ruGPT3-medium	Верните ивана Васильевича
ruGPT3-large	Верните Ивана Ивановича
ruT5-base	**Отстаньте, верните Иваныча!**
ruT5-large	**Отстаньте, верните Иваныча.**
Original	Ух с**а, меня аж прижало, еду удержался
Delete	Ух , меня аж прижало, еду удержался
CondBERT	Ух су , меня аж прижало , еду удержался
ruGPT3-small	Меня даже прижало, еду удержался
ruGPT3-medium	Меня прям прижало, еду удержался
ruGPT3-large	Меня аж прижало, еду не удержался
ruT5-base	Меня аж прижало, еду удержался
ruT5-large	**Меня аж прижало, еле удержался.**

Table B.11: Examples of Russian detoxifications by different models. Bad answers are shown in red, the best answers **in bold**.

B.5.3 Multilingual Detoxification Examples

Table B.12 contains the outputs of our fine-tuned multilingual models for the English and Russian languages.

Original	After all its hard to get a job if you're st*pid.
mBART 5000 ENG	After all its hard to get a job if you don't understand.
mBART 3000 ENG	After all its hard to get a job if you are not smart.
mT5 base (multilingual)	It's hard to get a job if you don't know.
Original	F**k minimum security, put him in real prison.
mBART 5000 ENG	Put him in a real prison with minimum security.
mBART 3000 ENG	I don't care about minimum security, put him in real prison.
mT5 base	I don't care about minimum security, put him in real prison.
Original	Sure he should get punished, but three f**king years?
mBART 5000 ENG	Sure he should get punished, but three years?
mBART 3000 ENG	He should get punished, but three years?
mT5 base	He'd get punished, but three years?
Original	Что у этих людей в башке!? походу нас**то!
mBART 5000 ENG	Что у этих людей в голове!? Походу бесчеловечно.
mBART 3000 ENG	Что у этих людей в голове? Походу ненормально!
mT5 base	походу этих людей!? походу!

Table B.12: Detoxified examples produced by our fine-tuned multilingual models.

B.6 Non-detoxifiable Samples

Table B.13 gives examples of sentences that could not be detoxified by any of the workers who tried to do that. We group the samples by the reasons for the inability to detoxify that the workers indicated.

Unclear content
she is! f**k's s*ke stross! *fights against you*
yourwhy bring in the red herring of extramarital s*x, getting desperate aj?
wikibofh f**k u u f**kin suck argument is fine u nerd now unblock me.
you have to be a complete *diot to be supporting the gop ticket this go around.
wikipedia review and tell the world why you not a grand d**khead
trump will make a f*ol out of little potato
dont d*mb as in dead d*mmy.
Toxic content
f*ck you you mother f**king a*s cr**ker d**do licking piece of f*cking a*s sh*t!
if by christian you mean h*pocrites.
lying anti american m*slim mouthpiece.
they simply are too st**id to do otherwise.
xman is either an *diot or a ps**hopath or maybe both.
youre the insignificant f**k mother f**ker!
hello, you s*xy little loveh*le!
No toxicity
this is all coming out of our darned pockets, and i am not confident!
you people complete lack morals.
this idea seems to be a rather *gnorant one.
youre implying, therefore, that women ought to stay away from all black men.
blaming everyone else for the hole that you dug is pathetic.
k*lling the innocent nearly born should be the very last choice.
*gnorant to me means without knowledge.

Table B.13: Examples of sentences which could not be detoxified for different reasons.