REVIEW OF RECENT TRENDS IN THE DETECTION OF HATE SPEECH AND OFFENSIVE LANGUAGE ON SOCIAL MEDIA

Zuzana SOKOLOVÁ*, Ján STAŠ**, Jozef JUHÁR***
Department of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, Tel. 055/602 3307
E-mails: *zuzana.sokolova@tuke.sk, **jan.stas@tuke.sk, ***jozef.juhar@tuke.sk

ABSTRACT

In the article, we describe recent trends in the detection of hate speech and offensive language on social media. We accord from the latest studies and scientific contributions. The article describes current trends and the most used methods in connection with the detection of hate speech and offensive language. At the same time, we focus on the importance of emoticons, hashtags, and swearing in the field of social networks. We point out the topicality of the selected topic, describe the next direction of our work, and suggest possible solutions to current problems in this field of research.

Keywords: *BERT, DistilBERT, hate speech, natural language processing, offensive language, social media*

1. INTRODUCTION

Currently, the issue of NLP is very popular and demanded by society. People are increasingly on the Internet and use several social networks to discuss many topics (politics, Covid-19 issues, climate crisis, opinions, and others). In recent times, social media has become a hotbed for hate speech which can have harmful effects on society. At the same time, there is an increasing rivalry, arrogance, unkindness, enmity, and envy among people who have different views on the same topics. Hate speech or offensive language can damage relationships between different groups of people. In the comments, people argue, insult, and prove who is right. Detection of hate speech is important because it can help prevent these harmful effects. In our opinion, dealing with this issue is important and necessary mainly because it would at least help to relieve the virtual world of negative thinking. Because when you look at a social network, in most cases you will come across slanderous comments. People are full of hate and anger lately. Sometimes they express themselves in a very ugly and derogatory way. Negative thinking leads, for example, to suicide, self-harm, psychological disorders, imbalance, diseases from stress, drugs or alcohol addiction, and others [1]. Therefore, in our opinion, it is important to deal with this issue and at least partially limit the expression of these people (bullies). That is why we need to focus on hateful and offensive language on the Internet and social networks.

Recent results show that it is best to use the BERT, DistilBERT, and mBERT language models to detect hateful and offensive speech. The Hugging Face research group has also significantly helped to simplify work with BERT models. Therefore, in this article, we will analyze how these models work, discuss some interesting other studies, and outline where we are heading in our research.

2. THE CURRENT STATE OF THE ISSUE

In 2018, an article by Devlin et al. [2] was published and they introduced the BERT language model. BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. It is based on the original implementation described in Vaswani et al. [3]. The transformer includes two separate mechanisms – an encoder that reads the text input and a decoder that produces a prediction for the task. Since BERT's goal is to generate a language model, only the encoder mechanism is necessary. The transformer encoder reads the entire sequence of words at once. Therefore, it is considered bidirectional (or rather non-directional). This characteristic allows the model to learn the context of a word based on all its surroundings (left and right of the word). It is well known that a directional approach essentially limits contextual learning. BERT generates embeddings by taking information from both the left and the right side of a token's context using the transformer model and hence called Bidirectional Encoder Representations from Transformers. Google has published the entire source code on GitHub [4]. These models cover 103 languages and have been released as open-source.

Hugging Face is the company that created a chat application for bored teenagers. The New York-based startup created a fun and emotional bot. Hugging Face can generate a digital friend so you can text back and forth and trade selfies [5]. The company has professional experience in the field of natural language processing (NLP). At the moment the company provides open-source NLP technologies, and a few years ago released an open-source library for natural language processing applications. From its chat app to this day, Hugging Face has been able to swiftly develop language processing expertise. The company aims to advance NLP and democratize it for use by everyone [6]. More and more companies are adding NLP technology to improve interaction. Therefore, it is essential to have quality libraries in which it is possible to train language models. This significantly saves costs and time. The BERT model variants developed by Hugging Face are considered highly effective. You will meet them everywhere. Hugging Face provides its pre-trained language models, and it becomes easier for businesses to extract easily decodable information on how well their product is functioning instead of deciphering graphs and reports. The main idea of NLP is to create a technology that understands every language and input on which the human world works.

Transfer learning is becoming the most widespread method of data training approach in natural language processing. However, it is very difficult to work with such large-scale pre-trained models and operate them. V. Sanh et al. introduced DistilBERT [7], a general-purpose pre-trained version of BERT, 40% smaller, 60% faster, that retains 97% of the language understanding capabilities. DistilBERT follows the same architecture as that of BERT [2] while reducing the number of layers by a factor of 2. DistilBERT follows a triple loss language modeling, which combines cosine distance loss with knowledge distillation for it (the student) to learn from the larger pre-trained language model (teacher) during training.

Sentiment analysis is very useful for finding out emotions, popularity, feelings, opinions about a product or a person, etc. In marketing, these analyzes are commonly used to determine interest in a product that is sold on the market. For example, a database of public reviews will be run through such an analysis, and the store will find out where they made a mistake and how they can improve the given product. In the field of NLP, sentiment analysis is mainly used to analyze text for statistical purposes. It is also often associated with the detection of hate speech. I did such an analysis in my master's thesis and it is described in detail in the article [8]. We worked with comments under political posts and created a hate speech detection tool.

In a study by B. Wei et al. [9] they tested the bidirectional LSTM (Bi-LSTM) model, which can better understand the context, and called it the AlBERT model. They worked on sentiment analysis, where they used the sentiment function of TextBlob to analyze the sentiment of tweet data. TextBlob returns two properties, polarity, and subjectivity when we run it on the text data. Consequently, they prepared the dataset and divided it into 3 groups. Since they had an unbalanced set, where specifically the "hate-speech" group had the size 6% from the total size of the dataset (100%), the "offensive-language" group had the size 77%, and group "neither" had the size 17%. Consequently, they had to ensure that no data, no duplicate data, and no discrepancies were missing from the tweet data. They used data augmentation to improve the even representation of classes in the model. Data augmentation is a data oversampling technique used to increase the size of the data by adding new samples that have a similar distribution to the original data or marginally altering the original data. They tried word embedding based on the replacement BERT model and synonym augmenter "Wordnet" as their main synonym augmenter.

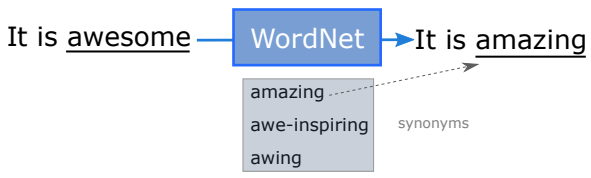


Fig. 1 Example of Word Embedding [9]

In basic terms, these two models tried to find a synonym for words in the tweet and replace the original with the synonym. The selection of synonyms is based on pre-trained embedding (See Figures 1 and 2).

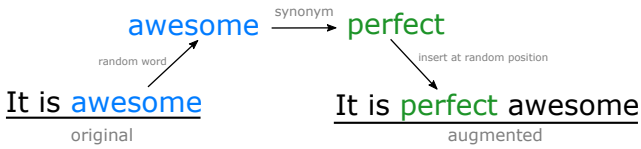


Fig. 2 Example of selection of synonym [9]

They used transfer learning and the pre-trained BERT model. At the same time, they froze all layers of the pre-trained model during tuning and added a dense layer and a Softmax layer to the architecture. They tested three different trained models for this task, by importing BERT Base (110 million parameters), DistilBERT (66 million parameters), and GPT-2 (1.5 billion parameters). They used the Hugging Face tokenizer for DistilBERT and BERT and the GPT-2 Hugging Face tokenizer for GPT-2 (in Figure 3 you can see the exact architecture). All layers of the pre-trained DistilBERT, BERT, and GPT-2 models were frozen. For the conversion of non-textual elements like emojis into text, they were using the model GloVE. For instance, smile face emojis were converted into the word "Smile". Prepared model GloVE with 100 dimensions and the activation function ReLu. The first layer had 128 neurons and the second layer had 64 neurons. To prevent overfitting, the dropout regularization technique is used on both layers. For the model, they added a softmax layer to classify the output and used the categorical cross-entropy loss function. They chose the "Adam" optimizer function for the model. The batch size was set to 128 and the number of epochs was set to 20 to keep the training time short. They used a Keras tokenizer to vectorize the text and convert it to a sequence of integers. Padding was used to create word vectors of the same size and the sequences were set to uniform length (MaxLength). According to the results achieved for the hate class, the AlBERT model had the best accuracy (91%) and DistilBERT (85%) was in second place. The results for the offensive and neither classes were similar to those for the hate class.

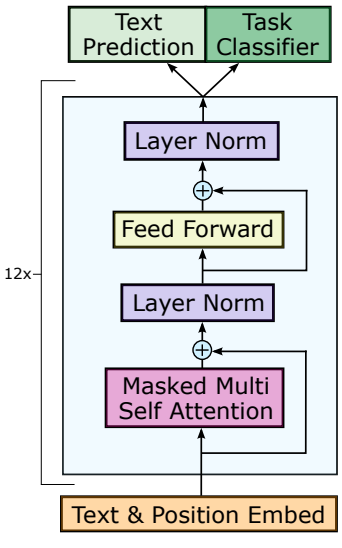


Fig. 3 GPT-2 Architecture [9]

DistilBERT and BERT achieved the best results in a study by A. Hande et al. [10] in offensive language research and sentiment analysis in 2021. Accuracy ranged between 78-98%, where they worked with comments from YouTube from 3 personalities. They tried to fine-tune the multilingual pre-trained language model. The performance of these models highlights the advantages of using a Multi-task learning model (MTL) to attend to two tasks at a time, mainly reducing the time required to train the models while additionally reducing the space complexities required to train them separately. MTL is a practical approach to utilizing shared characteristics of tasks to improve system performances. In MTL, the objective is to utilize learning multiple tasks simultaneously to improve the performance of the system [11]. Figure 4 shows the sharing of parameters between tasks.

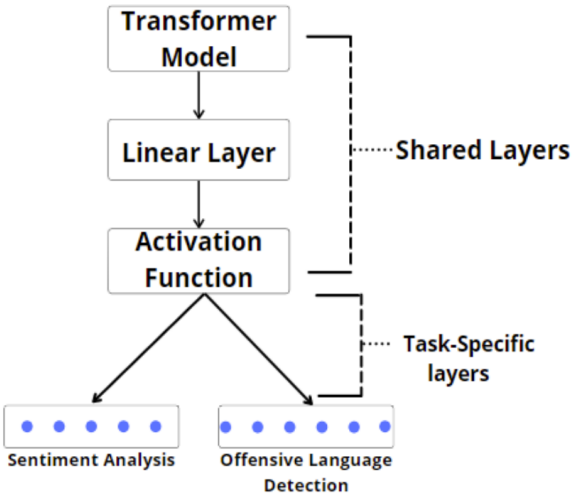


Fig. 4 Example of hard parameters sharing for sentiment analysis and offensive language identification [10]

Yasaswini et al. [12] achieved relatively high F1 scores by the DistilBERT model. They mainly use a case pre-trained multilingual DistilBERT model due to the presence of code-mixed data in their corpus. These tend to be case-sensitive language in the corpus. The training dataset consists of comments in six different classes:

- Not-Offensive;
- Offensive-Targeted-Insult-Individual;
- Offensive-Targeted-Insult-Group;
- Offensive-Targeted-Insult-Other;
- Offensive-Untargeted;
- Not-in-intended-language.

They also tried mBERT – Multilingual models of BERT [13]. This model is from a colossal part based on the architecture of BERT [2]. The model was pre-trained using the same pretraining strategy that was applied for BERT, i.e, Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). Also, the model was pre-trained on 104 languages from Wikipedia. To account for the data imbalance due to the size of Wikipedia for a given language, exponentially smoothed weighting of data was performed during data creation and word piece vocabulary creation.

Table 1 Sample of weighted F1-scores of offensive language detection models on the datasets [12]

Model	Weighted F1-Score		
	Malayalam	Tamil	Kannada
CNN-BiLSTM	0.8367	0.6102	0.4857
mBERT-cased + BiLSTM	0.9282	0.7149	0.7029
mBERT-uncased	0.8338	0.6189	0.3936
mBERT-cased	0.8296	0.6078	0.3882
XLM-R-base	0.8645	0.6173	0.4748
DistilmBERT-cased	0.9432	0.7569	**0.7277**
albert-base-v2	0.8268	0.6112	0.3890
ULMFiT	**0.9603**	**0.7895**	0.7000

This results in high-resource languages being under-sampled, while low-resourced languages are over-sampled. With this model, they achieved a worse F1 score compared to the DistilBERT model. The achieved results can be seen in Table 1.

Another interesting study is from the Slovak researchers M. Pikuliak et al. [14], who created a language model called SlovakBERT. This model has RoBERTa architecture (Liu et al. [15]) and it is trained on a web-crawled corpus. They dealt with tasks such as part-of-speech tagging, semantic textual similarity, sentiment analysis, and document classification. The accuracy of POS tagging was 98.37%. Their dataset contained data from Wikipedia, Open Subtitles, OSCAR corpus, and the Slovak websites. The dataset contained a total of 19.35GB of clean text without HTML tags. They used a BPE tokenizer with a vocabulary size of 50,264. The model was trained for 300k training steps with a batch size of 512. Samples were limited to a maximum of 512 tokens so that as many whole sentences as possible fit into each sample. They also used the Adam optimization algorithm and Hugging Face Transformers. The F1 score for sentiment analysis was 0.672. They used 41,084 tweets with 11,160 negative samples, 6,668 neutral samples, and 23,256 positive samples. SlovakBERT achieves state-of-the-art results on these tasks. They also released fine-tuned models for the Slovak community. They noted that existing multilingual models can achieve comparable results on some tasks, however, they are less efficient memory-wise or compute-wise.

More and more researchers are trying to create a model that works regardless of the language in which the text is written. Mixed corpora of texts are created, on which the use of the transfer learning method together with language models has proved successful.

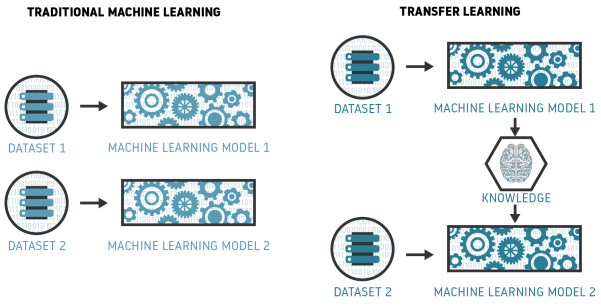


Fig. 5 Traditional machine learning vs Transfer learning [16]

Figure 5 describes the difference between traditional machine learning and transfer learning. A study by I. Bigoulaeva et al. [17] investigated the detection of hate speech on corpus texts in English and German. They collected 10,000 examples/comments of hate speech. They studied the behavior of the mBERT, BiLSTM, and CNN (Convolutional Neural Networks) models. They achieved the highest accuracy using CNN (78.11%), followed by the BiLSTM model (71.04%), and finally mBERT with an accuracy of 66.31%. They wanted to find out if transfer learning could alleviate the problem of lack of data. Because they worked with a class-unbalanced corpus of texts. Their results show that learning through transfer learning is effective in low-resource languages.

However, deep learning from class imbalanced data is still largely understudied, and statistical evidence which compares newly published methods across a variety of data sets and imbalance levels does not exist [18].

To achieve good results with the help of the language model, we need to prepare a high-quality annotated database of texts. A study by I. Molles et al. [19] described the creation of a dataset that is smaller, but class balanced. They downloaded comments from social networks YouTube and Reddit and made the dataset named ETHOS. Also, they made sure that the set was as balanced as possible. They cleaned the comments and collected them into a CSV file. Each comment had 8 labels (labels can have a value of 0 or 1) assigned to it. Where value 0 or 1 means the occurrence for every category which is (or isn't) present in the given comment. Mentioned categories: violence, directed_vs_generalized, gender, race, national_origin, disability, religion and sexual_orientation. For example sentence from a hate speech comments on a social network: "Islam is the home of terrorism". In this sentence was mentioned word "Islam", for that they marked label 1 for the category religion. That means that string of labels for this comment should look like this: "0; 0; 0; 0; 0; 0; 1; 0", where:

- 0 (violence);
- 0 (directed_vs_generalized);
- 0 (gender);
- 0 (race);
- 0 (national_origin);
- 0 (disability);
- 1 (religion);
- 0 (sexual_orientation)

We can see that for category "religion" is marked with value 1. According to how people chose the annotation tool (view Figure 6). It has an even balance in the "isHate" category (55.61% comments without hate speech and 44.39% comments with hate speech content). It also has an almost perfect balance between the other 6 categories. Where was 19.41% for gender, 17.16% for a race, 16.70% for national origin, 11.96% for disability, 18.28% for religion, and 16.48% for sexual orientation. Additionally, their dataset keeps a fair ratio between the rest of the labels, 32.28% and 67.72% for violent and non-violent comments, respectively, and 31.83% and 68.71% for direct and generalized comments, respectively. They achieved the highest accuracy using the DistilBERT model (about 80%) in all categories.

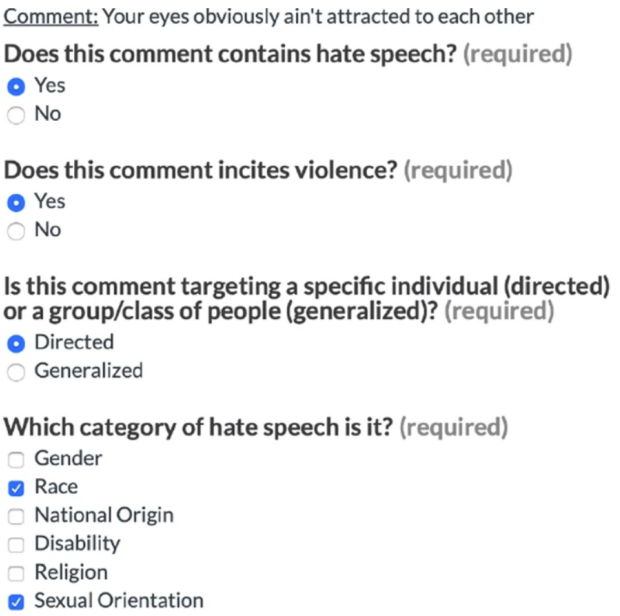


Fig. 6 Example of annotation tool [19]

3. NEXT DIRECTION OF THE WORK

Based on the studied materials, the process of analysis and detection of hate speech will look like this. Inspired by article [19] by I. Mollas et al., we plan to create the largest possible text database in the Slovak language, which would be class-balanced and clean. We would like to download the text (comments) from Facebook, Instagram, and Telegram. Because hateful, vulgar, offensive and racist comments in the Slovak language are most common here. In the annotation tool, we assign several labels to each comment. Above all, it will be a decision whether it is hate speech, offensive language, or neither. Furthermore, there could be a label about the category of the focus of the comment. We would use the created database in various tasks of hate speech analysis and offensive language. It is also very important to determine what is considered hate speech and what is offensive language. In the Slovak language, it is very difficult to distinguish whether it is hate speech or offensive language. It's even more complicated for the machine learning or language model. People who speak Slovak (but of course also in other languages) often consider these two terms as one. We plan to annotate the database manually, although it is laborious and time-consuming. However, at the moment this choice is the best solution for us because a person (human factor) knows best how to distinguish the context in the text. We will then check the annotated database to confirm the correctness of the annotations. However, we do not rule out automation over time, which would greatly facilitate the work for further research. We will collaborate with students from our department when annotating the database. Because of this problem, we were interested in the study by B. Wei et al. [9], where within the annotation of the corpus of texts, most of the comments were described as hate speech.

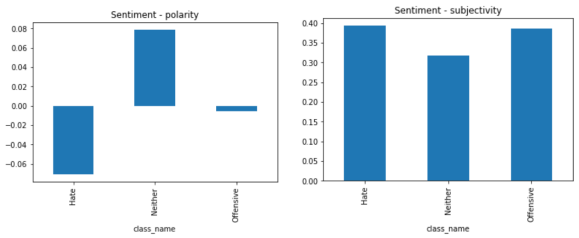


Fig. 7 Example of sentiment analysis [9]

They used TextBlob analysis which returns two properties, polarity, and subjectivity when we run it on text data. The polarity is variable, which lies in the range [-1,1], where 1 represents a positive statement and -1 a negative statement (results are shown in Figure 7). Subjective sentences generally refer to personal opinions, emotions, or judgment, while objective sentences refer to factual information. Subjectivity is also a float that lies in the range of [0,1]. We can see there is no significant difference between different classes for subjectivity. For polarity, they found that the hate class usually has more negative sentiment compared to neither and offensive. This could prove to be important information that helps to differentiate the hate class from the offensive and neither class.

An article by T. Davidson et al. [20] deals with exactly this problem. The best-performing model has an overall precision of 0.91, a recall of 0.90, and an F1 score of 0.90. Looking at Figure 8, however, we see that almost 40% of hate speech is misclassified: the precision and recall scores for the hated class are 0.44 and 0.61 respectively. Most of the misclassification occurs in the upper triangle of this matrix, suggesting that the model is biased towards classifying tweets as less hateful or offensive than the human coders. Far fewer tweets are classified as more offensive or hateful than their true category; approximately 5% of offensive and 2% of innocuous tweets have been erroneously classified as hate speech.

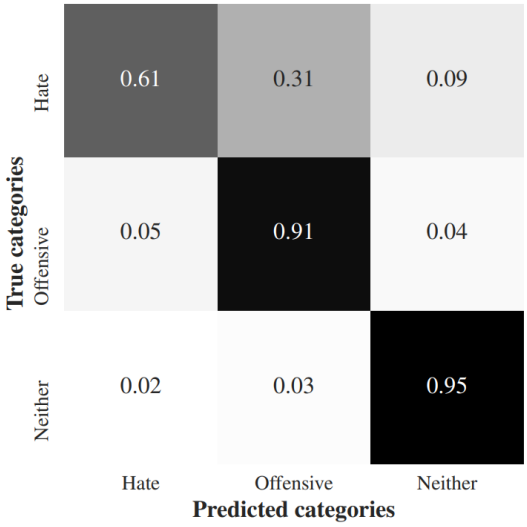


Fig. 8 True versus predicted categories [20]

They explored that tweets with the highest predicted probabilities of being hate speech tend to contain multiple racial or homophobic slurs. Other tweets tend to be correctly identified as hate when they contained strongly racist or homophobic terms like n*gger and f*ggot. Interestingly, they also found cases where people use hate speech to respond to other hate speakers. They also draw attention to the fact that it is necessary to study the people who use hate speech more closely. Focus on their characteristics, motivation, and the social groups in which they are included. Therefore, we would also like to focus on the demographic data of people commenting on social networks. Statistics would help us to uncover and outline more information about people, who often express themselves as derogatory, hateful, and offensive on social networks.

Emoticons play a very important role in detecting hate speech and offensive language. Studies [21] and [22] also looked at emoticons and their use in the detection of hate speech field. Nowadays, feelings and emotions on the Internet are expressed through emoticons. This means that preprocessing of text for emoticons is important to improve aggression detection [23]. Therefore, it would make sense to translate these emoticons into text so that we can continue working with them. In a study by F. Husain et al. [24] they worked with a similar problem, they converted emoticons to text. Beautifulsoup4 version 4.8.22 was used to extract the emoticon description in English from Unicode.org.

Table 2 Example of emoticons database [24]

Emoji	Arabic Label	English Translation
🙂	وجه مبتسم قليلا	slightly smiling face
😃	وجه مبتسم بعيون كبيرة	grinning face with big eyes
🐵	وجه القرد	monkey face
🐒	قرد	monkey
🎂	كعكة عيد الميلاد	birthday cake

Subsequently, they translated these descriptions into their own language (Arabic language) using the Translate 1.0.7 Python package (view Table 2). In total, they created a database of 1,374 emoticons. Therefore, it is very important that we create a similar database with an emoticon and verbal description for the Slovak language. This step could significantly help us to detect hate speech and offensive language more accurately. And, to better distinguish between hate speech and offensive language. Not only emoticons but also hashtags are important in detecting hate speech or offensive language. People tend to use hashtags to highlight important words in the text. Hashtags also help you share content quickly. For these reasons, it is necessary to keep the words from the hashtags in the text and remove only the initial grid or other characters such as underscores and hyphens. For example, with the hashtag #hate_speech, we would remove # and _. We would like to include this task in the goals of our research.

4. CONCLUSION

Hate speech is a big issue in social networks due to the high volume of data, which are generated every day. Recent works show the advantages of machine learning in dealing

with tasks required to choose between hateful posts from sarcasm or offensive language. Because it is difficult for a machine to distinguish sarcasm from hate speech or offensive language.

In this article, we have described the current and latest results in the field of hate speech and offensive language detection partially. We have outlined the direction in which our research will go. We want to create a quality database in the Slovak language designed to solve tasks for this issue. We will make sure that we consistently include emoticons, hashtags, and swearing in our solution because they are important in detecting a person's emotions. We then select one or more language models to analyze the created database.

Preliminarily, we would like to work with the DistilBERT model and also with other approaches because of the comparison of results. We will examine the differences between hate speech and offensive language. We want to create various statistics on people who post such hate/offensive/racist comments on social media.

ACKNOWLEDGEMENT

The research in this paper was partially supported by the Ministry of Education, Science, Research and Sport of the Slovak Republic under the research projects VEGA 1/0753/20, VEGA 2/0165/21, and KEGA 009TUKE-4/2019, and by the Slovak Research and Development Agency under the project of bilateral cooperation APVV SK-TW-21-0002 and project of applied research APVV-15-0517.

REFERENCES

[1] WHITLOCK, J. – WYMAN, A. P. – MOORE, S. R.: (2014) Connectedness and suicide prevention in adolescents: Pathways and implications. *Suicide and Life-Threatening Behavior*, Vol. 44, No. 3, ISSN 1943-278X, doi:10.1111/sltb.12071.

[2] DEVLIN, J. et al.: (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, Minnesota, pp. 4171–4186, doi:10.18653/v1/N19-1423.

[3] VASWANI, A. et al.: (2017) Attention is all you need". Guyon, I., et al. (eds.): *Advances in Neural Information Processing Systems*, Vol. 30, ISBN 9781510860964.

[4] DEVLIN, J. et al.: (2022) BERT. Available Online: `https://github.com/google-research/bert`

[5] DILLET, R.: (2017) Hugging Face wants to become your artificial BFF. Available Online: `https://techcrunch.com/2017/03/09/hugging-face-wants-to-become-your-artificial-bff/`

[6] SYAL, A.: (2020) Hugging Face: A step towards democratizing NLP. Available Online: `https://towardsdatascience.com/hugging-face-a-step-towards-democratizing-nlp-2c79f258c951`

[7] SANH, V. et al.: (2019) DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In: Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing, NeurIPS 2019, arXiv:1910.01108.

[8] SOKOLOVÁ, Z. – STAŠ, J. – HLÁDEK, D.: (2022) An introduction to detection of hate speech and offensive language in Slovak. In: Proceeding of the 12th International Conference on Advanced Computer Information Technologies, ACIT 2022, Ružomberok, Slovakia, ISBN 978-1-6654-1049-6, doi:10.1109/ACIT54803.2022.9913104.

[9] WEI, B. et al.: (2021) Offensive language and hate speech detection with deep learning and transfer learning. CoRR, arXiv:2108.03305.

[10] HANDE, A. et al.: (2021) Benchmarking multi-task learning for sentiment analysis and offensive language identification in under-resourced Dravidian languages. CoRR, arXiv:2108.03867.

[11] ALONSO, H. M. – PLANK, B.: (2017) When is multitask learning effective? Semantic sequence prediction under varying data conditions. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia, Spain, pp. 44–53.

[12] YASASWINI, K. et al.: (2021) IIITT DravidianLangTech-EACL2021: Transfer learning for offensive language detection in Dravidian languages. In: Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, Kyiv, Ukraine, pp. 187–194 .

[13] PIRES, T. – SCHLINGER, E. – GARETTE, D.: (2019) How multilingual is multilingual BERT? In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, pp. 4996–5001.

[14] PIKULIAK, M. et al.: (2021) SlovakBERT: Slovak masked language model. CoRR, arXiv:2109.15254.

[15] LIU, Y. et al.: (2019) RoBERTa: A robustly optimized BERT pretraining approach. CoRR, arXiv:1907.11692.

[16] MARTINEZ, D.: (2020) Is transfer learning the final step for enabling AI in aviation? Available Online: `https://datascience.aero/transfer-learning-aviation/`

[17] BIGOULAEVA, I. et al.: (2022) Addressing the challenges of cross-lingual hate speech detection. CoRR, arXiv:2201.05922.

[18] JOHNSON, J. M. – KHOSHGOFTAAR, T. M.: (2019) Survey on deep learning with class imbalance. *Journal of Big Data*, Vol. 6, No. 27, ISSN 2196-1115

[19] MOLLAS, I. et al.: (2022) ETHOS: A multi-label hate speech detection dataset. *Complex & Intelligent Systems*, doi:10.1007/s40747-021-00608-2.

[20] DAVIDSON, T. et al.: (2017) Automated hate speech detection and the problem of offensive language. In:

Proceedings of the International AAAI Conference on Web and Social Media, Montreal, Canada, pp. 512–515.

[21] DEL VIGNA, F. et al.: (2017) Hate me, hate me not: Hate speech detection on Facebook. In: Proceedings of the First Italian Conference on Cybersecurity, ITASEC17, Venice, Italy, pp. 86–95.

[22] KOVÁCS, G. – ALONSO, P. – RAJKUMAR, S.: (2021) Challenges of hate speech detection in social media. *SN Computer Science*, Vol. 2, No. 95, doi:10.1007/s42979-021-00457-3.

[23] ORĂSAN, C.: (2018) Aggressive language identification using word embeddings and sentiment features. In: Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying, TRAC-2018, Santa Fe, New Mexico, USA, pp. 113–119.

[24] HUSAIN, F. – UZUNER, O.: (2022) Investigating the effect of preprocessing Arabic text on offensive language and hate speech detection. *ACM Transactions on Asian and Low-Resource Language Information Processing*, Vol. 21, No. 4, doi:10.1145/3501398.

Received June 30, 2022, accepted December 6, 2022

BIOGRAPHIES

Zuzana Sokolová was born in 1996. In 2021 she graduated (M.S.) from the Department of Electronics and Multimedia Communications (DEMC) of the Faculty of Electrical Engineering and Informatics (FEEI) at Technical University in Košice. In July 2021, she received a certificate for outstanding university work in conjunction with Artificial Intelligence from Huawei. In September 2021 she began studying Ph.D. study at the DEMC. Her scientific research is focusing on natural language processing, especially the detection of hate speech and offensive language.

Ján Staš was born in 1984. In 2007 he graduated (M.S.) at the DEMC FEEI at Technical University in Košice. He defended his Ph.D. in the field of Telecommunications in 2011. Since 2015 he is working as an Assistant Professor at the Department of Electronics and Multimedia Communications. His scientific research is focusing on natural language processing, statistical language modeling, and speech recognition. He also investigates questions related to the detection of hate speech and offensive language.

Jozef Juhár was born in 1956. He graduated (M.S.) from the Technical University of Košice in 1980. He received Ph.D. degree in Radioelectronics at the Technical University of Košice in 1991. Currently, he is the Full Professor and Head of the Department of Electronics and Multimedia Communications. He is the founder of the Laboratory of speech and mobile technologies at DEMC FEEI TUKE. His research interests include speech and audio processing, automatic speech recognition, speaker recognition, speech analysis and synthesis, voice acoustics and research and development of spoken dialogue systems.