ISSN 2007-9737

Offensive Language and Hate Speech Detection Using Transformers and Ensemble Learning Approaches

Billel Aklouche[1,2,*], Yousra Bazine[2], Zoumrouda Ghalia-Bououchma[2]

[1] University of Constantine 2, Abdelhamid Mehri, LIRE Laboratory, Algeria

[2] University of Constantine 2, Abdelhamid Mehri, TLSI Department, Algeria

{billel.aklouche, yousra.bazine, zoumrouda.bououchma}@univ-constantine2.dz

Abstract. Social networks play a vital role in facilitating communication and information sharing. However, these platforms are also witnessing a growing prevalence of hate content, which can pose a major threat to individuals and entire communities. In this paper, we propose a new method that addresses the problem of offensive language and hate speech detection using seven transformer models, including BERT, and six ensemble learning strategies (Majority Voting, Averaging, Highest-sum, Stacking, Boosting and Bagging). Specifically, a fine-tuning process is run for each pre-trained model on hate speech detection downstream task. Subsequently, various ensemble learning techniques are applied by combining the predictions of individual models in order to improve overall performance. Extensive experiments have been conducted on the publicly available Davidson-dataset to assess the performance of our proposed method. Evaluation demonstrates promising results in terms of various evaluation metrics, outperforming competitive state-of-the-art baselines.

Keywords. Hate speech detection, offensive language, transformers, fine-tuning, ensemble learning, social media.

1 Introduction

Hate speech is a form of expression that aims to offend, hurt or discriminate against a person or a particular group. Indeed, it can have negative impacts on both the mental and physical health of victims [4]. According to the United Nations, it even affects the democratic values, social stability and peace. Therefore, detecting and preventing online hate speech is a crucial task for ensuring a safe and respectful environment for everyone. Social media platforms such as Twitter and Facebook are among the most common sources of hate speech, as they allow users to express their opinions and emotions freely and anonymously.

In fact, several suspects in recent hate-driven terrorist attacks had an extensive history of posting hateful content [11]. This emphasizes the crucial importance of detecting this kind of speech, particularly on social media. However, manually monitoring and moderating such a huge volume of user generated content is impractical, costly and time-consuming.

Therefore, there is a significant need for automated methods that can accurately and efficiently identify and classify hate speech within textual content on social media platforms [9]. In this paper, we propose a new method for automatic hate speech detection that leverages the performance of transformer-based models and ensembling techniques.

Indeed, this choice is motivated by the outstanding ability of transformer models to understand natural language, enabling them to accurately capture semantic meaning, context and interrelationships within textual data. Such

an ability proves crucial in accurately identifying instances of hate speech [20].

In addition, the versatility of transformer models enables them to be integrated with ensemble learning techniques, thus improving the method's performance and robustness [12]. Ensemble Learning, in particular, is used to alleviate the problems of variance and over-fitting associated with individual transformer models by leveraging various ensembling strategies [16].

To this end, we have fine-tuned seven state-of-the-art pre-trained language models (BERT, RoBERTa, Electra, DistilBERT, ALBERT, Large-BERT, XLMRoBERTa) on hate speech detection downstream task. We have applied various ensemble learning strategies (Majority Voting, Averaging, Highest-sum, Stacking, Boosting and Bagging) to combine the predictions of the seven individual models and improve the overall performance.

Extensive experiments have been conducted to assess the performance of the proposed method involving different parameters and strategies. Evaluation demonstrates that our method achieves promising results and outperforms competitive baselines in terms of various standard evaluation metrics.

The remainder of this paper is organized as follows. Section 2 discusses some related work. Section 3 presents the proposed hate speech detection method. Experiments and results are outlined and discussed in Section 4. The last section concludes the paper and provides insights for future work.

2 Related Work

Online hate speech is a pervasive and complex phenomenon that poses serious challenges for researchers, policymakers, and social media platforms. The widespread use of social media, especially during the COVID-19 pandemic, has contributed to the spread and escalation of hate speech across different languages, cultures and topics [2]. Consequently, researchers have shown significant interest in this problem [8]. Here, we briefly review some recent studies on hate speech and offensive language detection.

In the study proposed by Mozafari et al. [18], authors introduced a method for hate speech detection in social media posts. The proposed method leveraged BERT, a pre-trained language model, that has been fine-tuned with different strategies examining the effect of combining different layers, such as Bi-LSTM and CNN.

Experiments demonstrated that the proposed method overcame some of the challenges in the existing data. The best results were achieved by combining all pre-trained BERT layers with a CNN layer. Malik et al. [14] reviewed 14 classifiers based on shallow or Deep Learning, using different word representation methods.

They found that neural network-based classifiers combined with BERT, ELECTRA or ALBERT performed better than other methods. Their best models were BERT+CNN and ELECTRA+MLP, which achieved a weighted F1-score of 91% on Davidson dataset [5]. In the study proposed by Kovács et al. [9], authors proposed a model that combined RoBERTa and FastText with CNNs and RNNs and obtained weighted F1-score of 72%.

The closest recent work to our strategy is the one proposed by Magnossão et al. [12], in which authors tested six transformer models and two basic ensemble methods but only on Arabic language. Their best results were achieved by using the ensemble learning majority vote, and reported an F1-score of 0.60 and Accuracy of 0.86 on the test set. More recently, the method of Mazari et al. [15] used FastText and GloVe word embeddings with Bi-LSTM and Bi-GRU to build Deep Learning models.

They combined the obtained models with BERT to create ensemble learning architectures and reported a ROC-AUC score of 98.63% on social media data provided on Kaggle. Our proposal introduces a new strategy for hate speech detection that uses base transformers for classification and ensemble learning.

We examined the recent studies on this topic and found that most of them did not use only base transformers for classification. We found that most of the existing work integrated machine learning classifiers or transformers with other complex deep learning models.

Table 1. Used transformer models

Transformer	Developer
BERT	Google Research
Electra	
ALBERT	
Large-BERT	
RoBERTa	Facebook AI Research
XLM-RoBERTa	
DistilBERT	Hugging Face

3 Proposed Method

In this section, we describe our hate speech detection method, presenting the transformer models and ensemble learning strategies we applied. Figure 1 depicts the general architecture of our proposal. The proposed method can be divided into three main steps : (1) Data Pre-processing, (2) Fine-tuning Transformers, and (3) Ensemble Learning.

3.1 Data Pre-Processing

Data pre-processing is an essential and necessary phase before dealing with text data. its aim is to make raw data more suitable and understandable for machine learning or data analysis algorithms. This way, the model algorithm can work effectively and extract meaningful features and patterns from data [13].

In our method, this step essentially involves data cleaning, lowercase conversion and tokenization. Given our dataset, the result of this step consists of two distinct sets of data: a train set that forms the input of our primer models, and a separate test set for evaluation purposes.

3.2 Fine-tuning Transformers

Transformer models [22] have revolutionized the field of natural language processing (NLP). However, training these models from scratch requires a vast amount of data and substantial computational resources. Fortunately, there are various state-of-the-art pre-trained transformer models which are publicly available and can be fine-tuned for specific tasks. These models have different sizes: (i) Base, (ii) Medium, and (iii) Large, based on the number of parameters they can learn.

For our task, we chose, as shown in Table 1, seven of the publicly available transformer models, which have demonstrated significant results on various NLP tasks. The transformer models were imported from the open-source HuggingFace transformer library[1] and used for fine-tuning. Indeed, the fine-tuning process involves updating weights and adjusting the appropriate hyper-parameters for each model separately to optimize their performance.

The models were trained on the train set to learn task specific patterns and features. Once the training process is complete, we obtain a set of fine-tuned models, which are capable of classifying new tweets from the test set and producing level 1 predicted labels.

3.3 Ensemble Learning

Ensemble learning workflow is to train a set of individual base learners first and then combining them to improve results via some ensembling strategies [24]. In our method, we ensembled all of the fine-tuned transformer models from Table 1 using six different strategies of ensemble learning: Majority Voting, Averaging, Highest-sum, Stacking, Boosting and Bagging. We outline each technique employed in the following subsections.

3.3.1 Majority Voting

Majority voting can be approached in different ways, depending on the level of agreement required among the base transformer models: unanimous voting, simple majority, max voting [17]. We chose max voting, which takes the prediction that has maximum votes of the set of classifiers, according to the following formula:

$$\sum_{t=1}^{T} d_{t,c*} = \max_{c} \sum_{t=1}^{T} d_{t,c}, \tag{1}$$

[1] huggingface.co/docs/transformers/

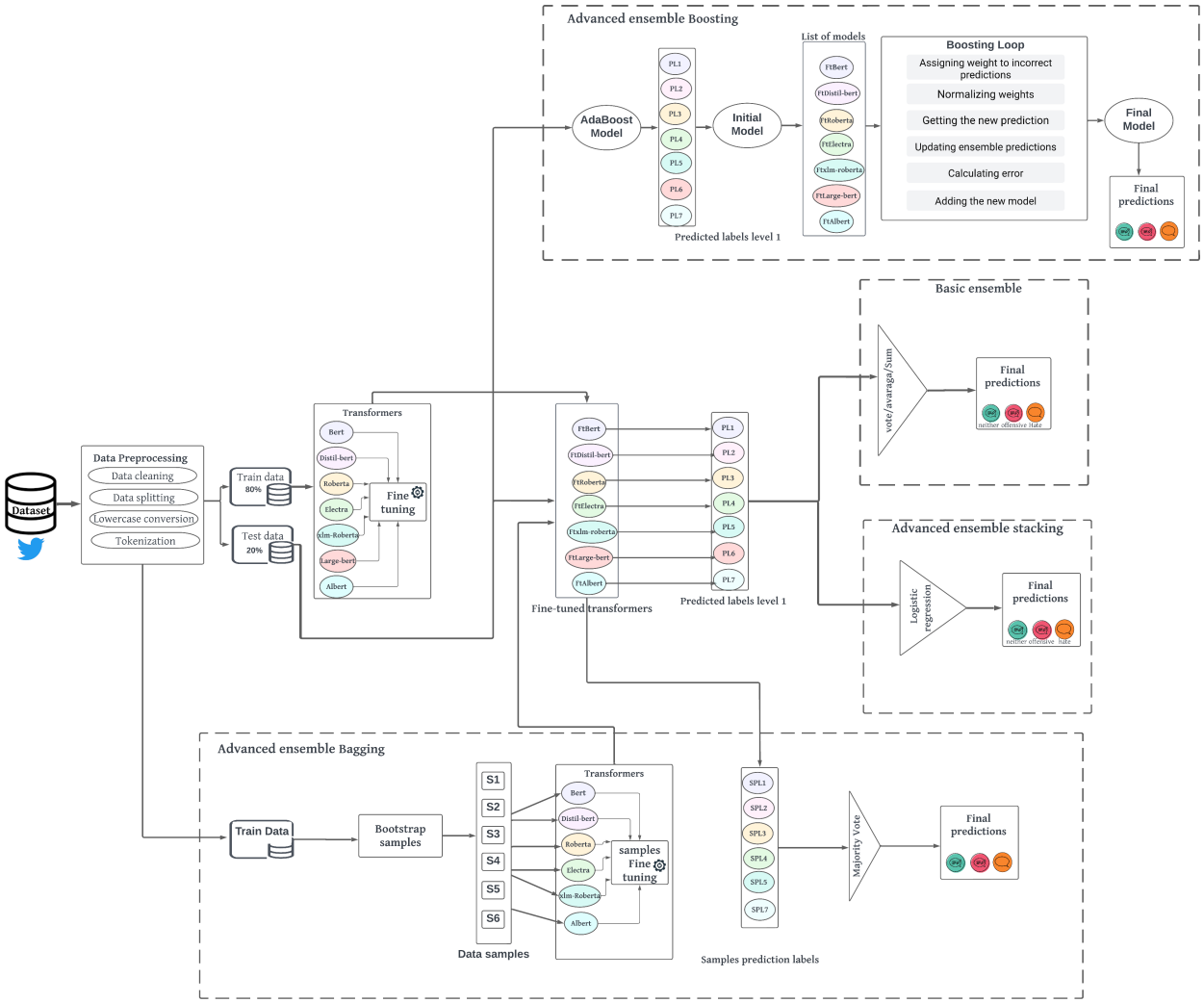

Fig. 1. Overall architecture of the proposed method

where $d_{t,c}$ represents the prediction of model t for the class c. The resulting class will be determined based on the majority of votes from the models.

3.3.2 Averaging

Averaging takes the average value of all models' probabilities. It can be mathematically presented as follows:

$$P_i^j = \text{softmax}^j(O_i) = \frac{O_i^j}{\sum_{k=1}^{K} \exp(O_k^j)}. \qquad (2)$$

Using Softmax function, where P_i^j represents the probability outcome of the i^{th} unit on the j^{th} base model, O_i^j is the output of the i^{th} unite of the j^{th} base model and K is the number of classes [7].

3.3.3 Highest-sum

In our highest-sum strategy, final predictions were made by selecting the classes with the highest sum of predicted probabilities or confidence scores from each one of the transformer models.

Table 2. Fine-tuning results for seven transformers models. The highest values of each model are marked in bold

Transformers	Hyper-parameters	Accuracy	F1-score
BERT	Batch Size = 8	91.24%	90.03%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 16	91.56%	90.56%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 32	91.22%	90.14%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 32	91.82%	91,22%
	Learning Rate = 2e-5		
	Epochs = 5		
RoBERTa	Batch Size = 8	91,40%	90,70%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 16	90.67%	90.14%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 32	91,56%	90.74%
	Learning Rate = 2e-5		
	Epochs = 5		
DistilBERT	Batch Size = 16	91.93%	90.76%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 32	92.17%	91,53%
	Learning Rate = 2e-5		
	Epochs = 3		
XLM-RoBERTa	Batch Size = 16	90.23%	90.16%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 32	91,74%	90.49%
	Learning Rate = 2e-5		
	Epochs = 3		
Electra	Batch Size = 16	90.98%	88,62%
	Learning Rate = 5e-5		
	Epochs = 3		
	Batch Size = 32	91,56%	90,62%
	Learning Rate = 2e-5		
	Epochs = 3		
ALBERT	Batch Size = 16	90.21%	89.39%
	Learning Rate = 2e-5		
	Epochs = 3		
	Batch Size = 32	91,14%	90,55%
	Learning Rate = 2e-5		
	Epochs = 3		
Large-BERT	Batch Size = 16	91,95%	91,00%
	Learning Rate = 2e-5		
	Epochs = 3		

3.3.4 Stacking

Performing stacking involves combining different fine-tuned base models to reduce their errors. The predictions from each model are stacked together and used as input to a final meta-model, which is a logistic regression in our case.

The meta-model was trained using cross-validation on the level 1 predictions of the base models and learned how to best combine them and produced the final predictions. Stacking can be represented as shown in formula 3, where h_t represent our base models trained on data D resulting the predictions z that was fed to the meta-model:

$$z = \text{stack}\{h_t(D)\}. \tag{3}$$

3.3.5 Boosting

Boosting is another ensemble learning technique that aims to improve the performance of a single ”weak” learner by successively training multiple models, where each subsequent model focuses on misclassified samples or samples with high residual errors compared to previous models [23].

3.3.6 Bagging

Bagging, or Bootstrap Aggregating, is an ensemble learning technique that involves training multiple models independently on different subsets of the training data. After training all the base models, their predictions are then combined to make a final prediction.

4 Experiments and Results

In this section, we first describe our dataset and experimental settings. Then we discuss the obtained results.

4.1 Dataset

For dataset selection, we have carefully considered various factors such as the diversity of content, annotation quality, and the relevance of the dataset to real-world social media scenarios. We therefore chose to work with the Davidson dataset [5] for our experiments, which was specially created for the purpose of detecting hate speech and offensive language on social media.

It serves as a valuable resource in the field of harmful content detection, providing researchers with a diverse collection of annotated comments extracted from Twitter. The dataset is publicly available and it contains a significant number of tweets in English, ensuring a substantial amount of data for analysis. It consists of approximately 25,296 labeled instances classified into three classes: Hate speech (5.70%), Offensive Language (77.60%) or Neither (16.70%).

4.2 Experimental Setup

We present a comprehensive investigation into the application of fine-tuning techniques with transformer models to enhance the classification performance across the variety of models that we used. One of the benefits of transformers is that they do not require a lot of pre-processing in order to make the data understandable. For our method, we adopted the main following steps for pre-processing our dataset:

- Data Cleaning: includes removing useless columns, removing URLs, removing mentions and users, replacing numbers with $\langle \text{numbers} \rangle$, removing special characters, etc.
- Using Sklearn library to apply stratified splitting of the dataset into an 80:20 ratio, with 80% as Train set and 20% as Test set.
- Converting data into lower case format.
- Tokenizing data: each one of the transformer models has its own tokenizer from the pre-trained transformer library, which has its own characteristics and allows to do tokenization or reverse tokenization.

Afterword, the pre-processing step is followed by a fine-tuning process. For each model, we meticulously evaluated their Accuracy and F1-score on the Davidson dataset, by leveraging specific hyper-parameters, notably employing several batch sizes, as well as multiple learning rates. We employed a max length padding of 128, which is the nearest number to the maximum tokenized sequence length of 125. The dropout probability was set to 0.1. A range of 3 to 5 epochs was employed for training. Subsequently, to aggregate the outcomes of all fine-tuned models, we adopted some of the comprehensive ensemble learning techniques.

4.3 Results and Discussion

In Table 2, we present the results of a comparative study in which we assessed how various fine-tuning strategies impact the performance of our transformer models. We display the obtained results of the different applied parameters in terms of Accuracy and F1-score. The entire experimental process was conducted on Google Collaboratory, which provides free but limited daily usage of a Tesla 4 (T4) GPU and 12 GB RAM.

We began our experiments with BERT, one of the most well-known transformers for NLP tasks [6]. After examining multiple previous research and conducting personal experiments, we found that BERT achieves its best results when using a high number of batches. As shown in Table 2, we increased the batch-size from 8 to 32 and used a medium learning rate (2e-5), which yielded positive results.

We observed similar results for DistilBERT and ALBERT, which are smaller versions of the BERT model, designed to be more efficient, reduce model size and maintain similar performance with a lighter weight in terms of computational resources. As we can see, the best results were obtained with DistilBERT, reaching an accuracy of over 92%.

RoBERTa and XLM-RoBERTa models, developed by Facebook [1], are also variants of BERT which have been pre-trained longer over more data with bigger batches [10]. We chose to increase the batch size for these models as well, which had a positive impact on our results.

Table 3. Ensemble learning strategies test results compared to literature baselines

	Accuracy	F1-Score	Precision
Davidson et al.	–	90%	91%
Waseem et al.	–	89%	–
Mukherjee et Das	–	90,84%	–
Majority vote	92,35%	91,47%	91,40%
Highest-sum	92,25%	91,26%	91,26%
Averaging	92,15%	91,10%	91,22%
Boosting	92,33%	91,39%	91,37%
Bagging	91,80%	91,26%	91,02%
Stacking	**92,47%**	**91,80%**	**91,64%**

ELECTRA, designed to be more sample-efficient and quicker than other pre-trained models [3], also performed well when we adjusted the learning rate and used two different batch-sizes.

Large-BERT was the only model that restricted our experiments to a maximum batch-size of 16 due to its large size and the limitations of our experimental environment. Despite these limitations, we still obtained promising results.

The obtained results demonstrate the effectiveness of fine-tuning transformer models for the hate speech detection task. As we can see, all the seven tested models performed well in terms of test accuracy and F1-score.

After obtaining the results of the transformers fine-tuning phase, we show in Table 3 how different ensemble learning strategies impact the performance of our final models.

In order to evaluate and compare the effectiveness of the proposed method, we consider the following works as reference baselines, which were chosen according to the Davidson dataset:

– **Davidson et al. [5]**: the method which resulted in the creation of the actual dataset.

– **Waseem et al. [21]**: the proposed method trained a machine learning model using a multi-task learning (MTL).

– **Mukherjee and Das [19]**: method in which authors used transformers and adopted RoBERTa model as their best result.

Results are reported on the test dataset in terms of three evaluation metrics: Accuracy, F1-score and Precision. As we can see, by applying ensemble learning techniques, we can observe an overall improvement in performance. We began by applying basic ensemble learning techniques, namely: Majority Voting, Highest-sum and Averaging.

From the results obtained, we note that the application of all three techniques led to a further improvement in performance compared to the results achieved by each model individually, with all three techniques exceeding 92% in terms of Accuracy. Furthermore, we can observe that our method outperforms the reference baselines in terms of all evaluation metrics.

In the next set of experiments, we turned to more advanced ensemble learning techniques, namely: Stacking, Boosting and Bagging. According to the results obtained, the Stacking method further enhanced performance, achieving an accuracy of 92.47% and an F1-score of 91.80%, in addition to a precision of 91.64%.

It is worth noting that applying the Bagging method required a large amount of memory, as all the models were trained simultaneously. Due to this memory constraint, it was not feasible to train all the models with this method.

Consequently, we had to exclude the ”Large-BERT” model due to its large size and keep the remaining six models. We believe that the results can be further improved with the Bagging method using all seven transformers.

5 Conclusion and Future Work

In this paper, we focused on the task of Offensive Language and Hate Speech detection using transformers and ensemble learning. We employed seven state-of-the-art pre-trained language models such as BERT and RoBERTa, as well as six ensembling strategies.

Experiments demonstrated the effectiveness of the proposed method in addressing the

challenges posed by hate content. The obtained results highlighted the performance of fine-tuned transformers and the significance of ensemble learning. These findings provide additional support for the efficacy of transformer-based techniques in NLP tasks, reinforcing their effectiveness and applicability. Results also showed that the use of ensembling strategies yielded better performance than using transformers alone.

This observation is exemplified by the successful application of different ensemble learning techniques, highlighting the power of combining multiple models to achieve better results. As future work, we plan to explore other transformer models and incorporate more diverse datasets as well as applying further hyper-parameters optimization to improve the detection performance. Another promising research direction is to study the use of the proposed method in other NLP tasks, such as Fake News Detection (FND).

References

1. **Chernyavskiy, A., Ilvovsky, D., Nakov, P. (2021).** Transformers: "The end of history" for natural language processing? Springer International Publishing, pp. 677–693. DOI: 10.1007/978-3-030-86523-8_41.
2. **Cinelli, M., Pelicon, A., Mozetič, I., Quattrociocchi, W., Novak, P. K., Zollo, F. (2021).** Dynamics of online hate and misinformation. Scientific Reports, Vol. 11, No. 1. DOI: 10.1038/s41598-021-01487-w.
3. **Clark, K., Luong, M. T., Le, Q. V., Manning, C. D. (2020).** ELECTRA: Pre-training text encoders as discriminators rather than generators. Proceedings of the 8th International Conference on Learning Representations, pp. 1–18. DOI: 10.48550/ARXIV.2003.10555.
4. **Cramer, R. J., Fording, R. C., Gerstenfeld, P., Kehn, A., Marsden, J., Deitle, C., King, A., Smart, S., Nobles, M. R. (2020).** Hate-motivated behavior: impacts, risk factors, and interventions. Health Affairs, Health Policy Brief, Vol. 9, pp. 1–6. DOI: 10.1377/hpb20200929.601434.
5. **Davidson, T., Warmsley, D., Macy, M., Weber, I. (2017).** Automated hate speech detection and the problem of offensive language. Vol. 11, No. 1, pp. 512–515. DOI: 10.1609/icwsm.v11i1.14955.
6. **Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2019).** BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.
7. **Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., Suganthan, P. N. (2022).** Ensemble deep learning: A review. Engineering Applications of Artificial Intelligence, Vol. 115, pp. 105151. DOI: 10.1016/j.engappai.2022.105151.
8. **Jahan, M. S., Oussalah, M. (2023).** A systematic review of hate speech automatic detection using natural language processing. Neurocomputing, Vol. 546, pp. 126232. DOI: 10.1016/j.neucom.2023.126232.
9. **Kovács, G., Alonso, P., Saini, R. (2021).** Challenges of hate speech detection in social media: Data scarcity, and leveraging external resources. SN Computer Science, Vol. 2, No. 2. DOI: 10.1007/s42979-021-00457-3.
10. **Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V. (2019).** RoBERTa: A robustly optimized BERT pretraining approach. Proceedings of the International Conference on Learning Representations, pp. 1–15. DOI: 10.48550/arXiv.1907.11692.
11. **MacAvaney, S., Yao, H. R., Yang, E., Russell, K., Goharian, N., Frieder, O. (2019).** Hate speech detection: Challenges and solutions. PLOS ONE, Vol. 14, No. 8, pp. e0221152. DOI: 10.1371/journal.pone.0221152.
12. **Magnossão-de-Paula, A. F., Bensalem, I., Rosso, P., Zaghouani, W. (2023).**

Transformers and ensemble methods: A solution for hate speech detection in Arabic languages. Proceedings of the Centre for Electronic Ubiquitous Research Workshop, pp. 1–7. DOI: 10.48550/arXiv.2303.09823.

13. **Maharana, K., Mondal, S., Nemade, B. (2022).** A review: Data pre-processing and data augmentation techniques. Proceedings of the Global Transitions, Vol. 3, No. 1, pp. 91–99. DOI: 10.1016/j.gltp.2022.04.020.
14. **Malik, J. S., Pang, G., Hengel, A. V. D. (2022).** Deep learning for hate speech detection: A comparative study. arXiv. DOI: 10.48550/arXiv.2202.09517.
15. **Mazari, A. C., Boudoukhani, N., Djeffal, A. (2023).** Bert-based ensemble learning for multi-aspect hate speech detection. Cluster Computing, Vol. 27, No. 1, pp. 325–339. DOI: 10.1007/s10586-022-03956-x.
16. **Mnassri, K., Rajapaksha, P., Farahbakhsh, R., Crespi, N. (2022).** Bert-based ensemble approaches for hate speech detection. IEEE Global Communications Conference, pp. 4649–4654. DOI: 10.1109/GLOBECOM48099.2022.10001325.
17. **Mohammed, A., Kora, R. (2023).** A comprehensive review on ensemble deep learning: Opportunities and challenges. Journal of King Saud University - Computer and Information Sciences, Vol. 35, No. 2, pp. 757–774. DOI: 10.1016/j.jksuci.2023.01.014.
18. **Mozafari, M., Farahbakhsh, R., Crespi, N. (2019).** A bert-based transfer learning approach for hate speech detection in online social media. Complex Networks and Their Applications VIII, pp. 928–940. DOI: 10.1007/978-3-030-36687-2_77.
19. **Mukherjee, S., Das, S. (2021).** Application of transformer-based language models to detect hate speech in social media. Journal of Computational and Cognitive Engineering, Vol. 2, No. 4, pp. 278–286. DOI: 10.47852/bonviewjcce2022010102.
20. **Roy, S. G., Narayan, U., Raha, T., Abid, Z., Varma, V. (2021).** Leveraging multilingual transformers for hate speech detection. Working Notes of Forum for Information Retrieval Evaluation, pp. 128–138. DOI: 10.48550/ARXIV.2101.03207.
21. **Talat, Z., Thorne, J., Bingel, J. (2018).** Bridging the gaps: Multi task learning for domain transfer of hate speech detection: Multi-task learning for domain transfer of hate speech detection. Springer International Publishing, pp. 29–55. DOI: 10.1007/978-3-319-78583-7_3.
22. **Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. (2017).** Attention is all you need. Advances in Neural Information Processing Systems, Vol. 30, pp. 5998–6008.
23. **Zhang, C., Ma, Y. (2012).** Ensemble machine learning: Methods and applications. Springer. DOI: 10.1007/978-1-4419-9326-7.
24. **Zhou, Z. H. (2009).** Ensemble Learning. Springer US, pp. 270–273. DOI: 10.1007/978-0-387-73003-5_293.

Article received on 17/10/2023; accepted on 06/05/2024.
**Corresponding author is Billel Aklouche.*