International Journal of Electrical and Computer Engineering (IJECE)
Vol. 14, No. 3, June 2024, pp. 2927~2936
ISSN: 2088-8708, DOI: 10.11591/ijece.v14i3.pp2927-2936

Fine-grained hate speech detection in Arabic using transformer-based models

Rajae Bensoltane, Taher Zaki
Laboratory of Innovation in Mathematics and Intelligent Systems, Faculty of Applied Sciences, Ibn Zohr University, Agadir, Morocco

Article Info

Article history:

Received Jun 28, 2023
Revised Feb 13, 2024
Accepted Feb 20, 2024

Keywords:

Arabic
Bidirectional gated recurrent unit
Fine-grained hate speech
Natural language processing
Transformer

ABSTRACT

With the proliferation of social media platforms, characterized by features such as anonymity, user-friendly access, and the facilitation of online community building and discourse, the matter of detecting and monitoring hate speech has emerged as an increasingly formidable challenge for society, individuals, and researchers. Despite the crucial importance of hate speech detection task, the majority of work in this field has been conducted in English, with insufficient focus on other languages, particularly Arabic. Furthermore, most existing studies on Arabic hate speech detection have addressed this task as a binary classification problem, which is unreliable. Therefore, the aim of this study is to provide an enhanced model for detecting fine-grained hate speech in Arabic. To this end, three transformer-based models were evaluated to generate contextualized word embeddings from input sequence. Additionally, these models were combined with a bidirectional gated recurrent unit (BiGRU) layer to further improve the extracted semantic and context features. The experiments were conducted on an Arabic reference dataset provided by the open-source Arabic corpora and processing tools (OSACT-5) shared task. A comparative analysis indicates the efficiency of the proposed model over the baseline and related work models by achieving a macro F1-score of 61.68%.

Corresponding Author:

Rajae Bensoltane
Laboratory of Innovation in Mathematics and Intelligent Systems, Faculty of Applied Sciences, Ibn Zohr University
Agadir, Morocco
Email: r.bensoltane@uiz.ac.ma

1. INTRODUCTION

The advent of online social networks has granted billions of online users a means to freely articulate their thoughts and ideas on the internet. This widespread accessibility has yielded substantial advantages in fostering cultural advancements. Nevertheless, it has also become a tool for spreading offensive and hate speech content by malicious people. The dangers of hate speech are becoming more evident, as research shows its connection to hate crimes worldwide [1]. Moreover, the proliferation of hateful and offensive content on the internet has been discovered to have an adverse impact on people's mental well-being [2], [3].

As a result, the natural language processing (NLP) community has become increasingly interested in automatically detecting hate speech. Hate speech refers to any form of abusive or offensive language, such as insults and threats, that targets individuals or groups based on race, religion, or sexual orientation [4], [5]. Despite the growing research on hate speech, there is still a lack of focus on this issue specifically in the Arabic language [6].

Arabic ranks 4th among the most widely used languages on the internet. However, the nature of Arabic text, with its ambiguity and informality, poses significant challenges in classifying social media content accurately [7]. Furthermore, the Arabic language encompasses multiple varieties with different vocabularies and structures, making the classification tasks more challenging.

The initial studies in Arabic hate speech detection have focused on building appropriate corpora for handling this task, such as the work of Chowdhury *et al.* [8] and Aref *et al.* [9]. These corpora classify a given text into only two categories: hate speech and not-hate speech. However, using a binary approach for handling this task is unreliable [10], [11]. moreover, several studies [12], [13] have adopted machine learning classifiers like support vector machine (SVM) and naive Bayes (NB) to handle this task. Nevertheless, these classifiers require a tedious process of feature engineering to select the most suitable features for training.

In recent years, transformer-based models have revolutionized the field of NLP by achieving the best results in many tasks like text classification [14], [15], question answering, named entity recognition [16], [17], and sentiment analysis [18], [19]. In addition, these models have obtained promising results in hate speech detection in different languages, including Arabic [20]–[23]. However, most of these studies have focused on using bidirectional encoder representations from transformers (BERT)-based models only. Other transformer models like generative pre-trained transformer (GPT) and efficiently learning an encoder that classifies token replacements accurately (ELECTRA) are under-explored in this area. Moreover, the impact of incorporating these models with more complex neural network layers has been investigated in many tasks like aspect term extraction [24], named entity recognition [25], and aspect category detection [26]. Yet further efforts are required to explore the effectiveness of this combined approach for the task of hate speech detection.

To overcome the above-mentioned shortcomings, this paper focuses on enhancing the results of the fine-grained hate speech detection task in Arabic language, which classifies a hate speech text into fine-grained classes, namely: race/ethnicity/nationality, social class, religion/belief, ideology, disability/disease, and gender. To this end, three different transformer-based models are evaluated to convert input tokens into contextualized vector representations. Additionally, we investigate combining these models with a bidirectional gated recurrent unit (BiGRU) layer to further encode semantic and context features from both left and right sides. To our knowledge, this is the first work to use this combination to handle this task in Arabic. Extensive experiments on a reference Arabic dataset are conducted to prove the effectiveness of our proposed method over the baseline and related work models.

The rest of this paper is organized as follows: section 2 provides related work to Arabic hate speech detection task. Section 3 explains our research methodology. The experimental setup is described in section 4, and the results of our experiments are discussed in section 5. Lastly, in section 6, we conclude the paper and suggest directions for future research.

2. RELATED WORK

Hate speech detection task can be performed as a binary (hate or not hate) or a multi-class (fine-grained classes such as race, gender, and social class) classification task. One of the earlier studies in Arabic language is that of Mulki *et al.* [12]. The authors introduced the first publicly-available Levantine hate speech and abusive (L-HSAB) Twitter dataset. They then extracted n-gram features using term frequency weighting to train SVM and NB classifiers. Another work of Chowdhury *et al.* [8] focused on detecting religious hate speech in Arabic. Authors proposed an approach that combines Arabic word embeddings and social network graphs. The researchers specifically emphasized the use of community features to enhance hate speech detection. To conduct their research, they collected a dataset of 3,950 Arabic tweets, with 1,685 tweets labeled as hate speech and 2,265 as non-hate speech. For classification, they utilized 600-dimensional word embeddings in combination with long-short term memory (LSTM) and convolutional neural network (CNN) models to train their detection model.

Additionally, several shared tasks have been conducted to handle this task in Arabic. The first shared task has been organized within the 4th Workshop on open-source Arabic corpora and processing tools (OSACT-4) [27]. The workshop focused on two binary classification tasks: identifying offensive language (subtask A) and detecting hate speech (subtask B). A dataset of 10,000 tweets was provided, with about 20% containing offensive language and 5% classified as hate speech. Among the different approaches tried, the SVM classifier combined with extensive pre-processing techniques showed the best results in subtask B [13].

Furthermore, the open-source Arabic corpora and processing tools (OSACT-5) workshop [28] expanded the shared task by introducing a more detailed categorization of hate speech in Arabic. Tweets labeled as hateful were further classified into six categories: race, religion, ideology, disability, social class, and gender. This extension aimed to explore the different dimensions and specific aspects of hate speech within these categories, leading to a more comprehensive analysis and understanding. The team of

Bennessir *et al.* [29] achieved first place in this competition for this task. They evaluated different pre-trained models in a multi-task fashion with task specific layers based on quasi-recurrent neural networks for each subtask. The second system was submitted by AlKhamissi and Diab [30]. The MARBERTv2 [31] model was used to encode the input text, following which it was fed into three dedicated classification heads designed for specific tasks. Each of these class-specific heads consists of a multilayered feed-forward neural network incorporating layer normalization.

A recent work of Althobaiti [32] proposed a BERT-based approach to detect Arabic hate speech and offensive language using the OSACT-5 dataset. They also explored the integration of sentiment analysis and emoji descriptions as supplementary features, alongside the textual content of the tweets. Their proposed method achieved enhanced results for offensive language detection and hate speech detection tasks. However, it achieved lower results for the fine-grained detection task (macro F1-score=25.25%). Al-Hassan and Al-Dossari [33] provided a new Arabic hate speech dataset that consists of 11K tweets. The dataset was collected using Twitter API and was manually annotated into five 5 distinct categories: none, religious, racial, sexism or general hate. The performance of the SVM classifier was evaluated in comparison to various deep learning models, specifically LTSM, CNN+LTSM, gated recurrent unit (GRU), and CNN+GRU. Among these models, the CNN+LSTM model yielded the best results. In a recent paper of Almaliki *et al.* [34], another dataset of 9,352 tweets was collected and labelled manually into three main classes: hateful, and abusive. They then utilized a BERT-based model, called ABMM to classify tweets into one possible class. However, these two new datasets are not publicly available. Therefore, we selected the public OSCAT-5 dataset to conduct the experiments in this study.

In contrast to most previous studies, our study is one of the few works that tackled the Arabic hate speech detection task in a fine-grained fashion. Moreover, we investigate combining transformer-based models with a BiGRU layer to further improve the encoded semantic and contextual features. To the best of our knowledge, this is the first time to adopt this combination to address this task in Arabic.

3. METHOD

3.1. Pre-processing

Data pre-processing is a crucial step in classification tasks. It helps remove unnecessary tokens that might not contribute to the task at hand and can even negatively impact the final results. To ensure optimal performance, we applied the following pre-processing steps to the dataset:

- Removing dates, time, numbers (both in English and Arabic), URLs, and Twitter-specific symbols like RT (re-tweet) and @ (mention symbol).
- Removing Arabic diacritics, which are the small markings used to indicate vowel sounds in Arabic text.
- Normalizing words by standardizing their format and removing unwanted punctuation marks and symbols.
- Removing elongation, which means reducing repeated letters to a single occurrence. For example, converting "طويييييل/*loooooog*" to "طويل/*Long*".

3.2. Model overview

The architecture of our proposed model is shown in Figure 1. First, a transformer-based model is employed to generate contextualized word embeddings from the input text. Then, a BiGRU layer is used to further encode semantic and context features from both left and right sides. Finally, a dense layer with a SoftMax activation function is utilized to provide the final label prediction.

3.3. Transformer

The transformer [35] architecture is based on the concept of self-attention mechanism, which allows the model to capture dependencies between different elements in a sequence. Unlike traditional recurrent neural networks (RNNs) that process sequential data sequentially, the transformer can process the entire sequence in parallel, making it more efficient and allowing for better capturing of long-range dependencies. various transformer-based models have been released such as BERT [36], GPT [37], ELECTRA [38], generalized autoregressive pre-training (XLNET) [39], and text-to-text transfer transformer (T5) [40]. These models have revolutionized the field of NLP by achieving state-of-the-art results in many tasks such as question answering [41] and named entity recognition [42].

3.3.1. ELECTRA

The ELECTRA architecture is a variant of the transformer-based models designed for pre-training language representations. It consists of two main components: the generator and the discriminator. The generator corrupts the input text by replacing some words, and the discriminator learns to distinguish between the original uncorrupted text and the generator's output. The generator and discriminator are trained iteratively to improve the language representation.

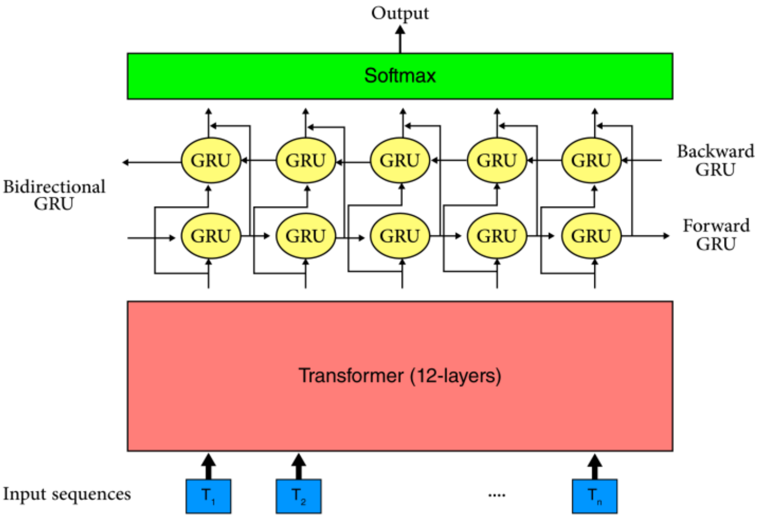

Figure 1. The architecture of our proposed model

3.3.2. BERT

BERT is another transformer-based model widely used for various NLP tasks, including text classification, question answering, and named entity recognition. BERT employs a masked language model (MLM) objective during pre-training, where it learns to predict masked or randomly replaced words in a given sentence. The transformer model in BERT allows the model to capture both the left and right context of a word, making it bidirectional and enabling a deeper understanding of the text.

3.3.3. GPT

GPT is a series of transformer-based models developed by OpenAI, starting with GPT-1 and followed by GPT-2 and GPT-3. The GPT models are trained using unsupervised learning on massive amounts of text data. They utilize a transformer architecture, which includes self-attention mechanisms and feed-forward neural networks, to capture dependencies and patterns in language.

For the above transformer-based models, we select the last hidden state of the [*CLS*] token as the final representation of the whole input text, which is then fed into the BiGRU layer. It is worth mentioning that for ELECTRA, the [*CLS*] token is not part of the original pre-training task but is added during fine-tuning for classification tasks. The final output of these models is formulated as (1).

$$x = \mathrm{H}^{[\mathrm{CLS}]} \in \mathrm{R}^{\mathrm{dim}} \tag{1}$$

where *dim* denotes the embedding size and its value is 768 for the three models.

3.4. BiGRU

GRU is a type of RNNs that are commonly used in deep learning architectures for sequential data processing tasks, such as NLP and time series analysis. It was designed to address some of the limitations of traditional RNNs by introducing gating mechanisms that control the flow of information within the network. Figure 2 illustrates the architecture of GRU. The key components of GRU unit are update and reset gates. The update gate determines how much of the previous hidden state should be retained and combined with the current input. The reset gate decides how much of the previous hidden state should be ignored when computing the current hidden state. Equations (2) to (5) illustrate the calculation formula:

$$z_t = \sigma(W_{zx}x_t + U_{zh}h_{t-1}) \tag{2}$$

$$\mathrm{r_t} = \sigma(\mathrm{W_{rx}x_t} + \mathrm{U_{rh}h_{t-1}}) \tag{3}$$

$$\tilde{h}_t = \tanh(W_{cx}x_t + r_t \odot U_{ch}h_{t-1}) \quad (4)$$

$$h_t = (1 - z_t)\odot o_t + z_t \odot h_{t-1}) \quad (5)$$

The symbol σ represents the sigmoid function. The operator $\odot$ signifies the element-wise multiplication of matrices. W and U are weight matrices that are adjusted during the learning process.

Instead of processing the input sequence only in the forward direction, we use a BiGRU layer to also process the sequence in the reverse direction simultaneously. The key idea behind BiGRU is to capture both past and future context for each element in the sequence. The outputs from both the forward and backward GRU units are concatenated to form the final representation of each element, as follows:

$$\overrightarrow{h_t} = GRU\left(x_t, \overrightarrow{h_{t-1}}\right) \quad (6)$$

$$\overleftarrow{h_t} = GRU\left(x_t, \overleftarrow{h_{t+1}}\right) \quad (7)$$

The output of the hidden layer h_t in BiGRU at time t is formed by combining both the forward and backward states:

$$h_t = \left[\overrightarrow{h_t} \oplus \overleftarrow{h_t}\right] \quad (8)$$

Lastly, the output of BiGRU is denoted as (9).

$$h = \{h_{1,}h_2, \dots, h_n\} \quad (9)$$

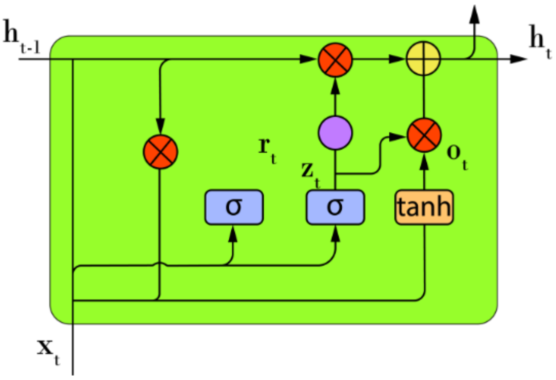

Figure 2. The structure of diagram of GRU

3.5. Output layer

The final representation of the BiGRU layer is fed into a final output layer. This layer consists of a dense layer with a SoftMax activation function to generate the label predictions. The output of this layer can be represented as (10).

$$\hat{y} = SoftMax(Wh + b) \quad (10)$$

Here, $\hat{y}$ represents the predicted probabilities. W is a weight matrix that can be adjusted during training. In addition, b indicates the bias vector for the classification layer.

3.6. Training process

During the training process, the main goal is to reduce the loss function between the predicted outcomes and the actual results. Given that the objective in this study involves solving a multi-class classification task, the loss function selected to measure the discrepancy between predicted and actual class probabilities is the categorical cross-entropy. The calculation formula for this is as (11).

$$L(\hat{y}, y) = -\sum_{i=1}^{N}\sum_{k=1}^{L} y_j^k \log\left(\hat{y}_i^k\right) \quad (11)$$

y_j^k represents the true label for a specific category, while $\hat{y}_i^k$ represents the predicted probabilities for that category. N represents the total number of training samples. L denotes the number of labels involved in the classification task.

4. EXPERIMENTS

4.1. Dataset

The evaluated dataset was provided by the OSCAT-5 shared task. It consists of about 12,698 tweets, annotated as clean, offensive, or hate speech. 6 classes were used to label hate speech tweets namely disability, social class, race, gender, religion, and ideology. Table 1 illustrates the distribution size of the dataset. More details about this data can be found in Mubarak *et al.* [43].

Table 1. Distribution size of the used dataset

Class	Total	Train	Test
Offensive	4,463	3,137	617
Hate-race	366	260	78
Hate-religion	38	27	7
Hate-ideology	190	144	32
Hate-disability	3	2	1
Hate-social class	101	72	19
Hate-gender	641	456	133
Normal	8,235	5,714	1,654

4.2. Experimental settings

We implemented the proposed model in Python using the TensorFlow and Keras libraries. For the BERT model, we used the base version of the MARBERTv2 model. The base version of AraELECTRA [44] was used for the ELECTRA model, and for GPT, we utilized the base version of the AraGPT2 [45] model. More details about the pre-training dataset of each model are shown in Table 2. Moreover, the hyper-parameters in this study were selected using the grid search optimization technique, as illustrated in Table 3.

Table 2. Size and source of the pre-training dataset of each transformer-based model

Model	Type of Arabic	Source	Size of pre-trained data
AraELECTRA	Modern standard Arabic (MSA)	Various Arabic corpora like El-Khair [46] and OSIAN [47]	8.6B tokens
AraGPT2	MSA	The same dataset as AraELECTRA	8.6B tokens
MARBERTv2	MSA+ dialectical Arabic (DA)	MSA corpora such as: OSCAR [48] and OSIAN [47] in addition to Arabic Tweets	29B tokens

Table 3. Experimental hyper-parameters

Hyper-parameter	Value
GRU hidden units	128
Optimizer	Adamax
Batch size	32
Max sequence length	128
Learning rate	5e-5
Number of epochs	5

4.3. Evaluation metrics

To assess the effectiveness of our model in comparison to the baseline and related work models, we utilized precision, recall, and F1-score metrics. Since the dataset is imbalanced, the macro score of these metrics is calculated. Equations (12)-(14) illustrate the calculation of each metric:

$$Macro\ F1 = \frac{2 \times MP \times MR}{MP + MR} \tag{12}$$

$$MP = \frac{1}{L} \sum_j^L MP_j \tag{13}$$

$$MR = \frac{1}{L} \sum_j^L MR_j \tag{14}$$

In the previous equations, L represents the total number of classes. MP_j refers to the precision of class j. Besides, MR_j represents the recall of class j.

5. RESULTS AND DISCUSSION

5.1. Results of different transformer-based models

We first conducted experiments to select the best transformer model. The evaluation results are illustrated in Figure 3. The worst results are achieved using the AraGPT2 model, which shows that this model might be not suitable for text classification tasks. Additionally, the MARBERTv2 model outperformed significantly the AraElectra model by more than 20% in terms of macro F1-score, indicating that this model is more accurate in handling this task. Moreover, the dataset used to pre-train the MARBERTv2 model is a combination of MSA and DA tweets, which aligns with the evaluated data in this study. Therefore, we used the MARBERTv2+BiGRU model for comparison with the baseline and related work models in the next subsection.

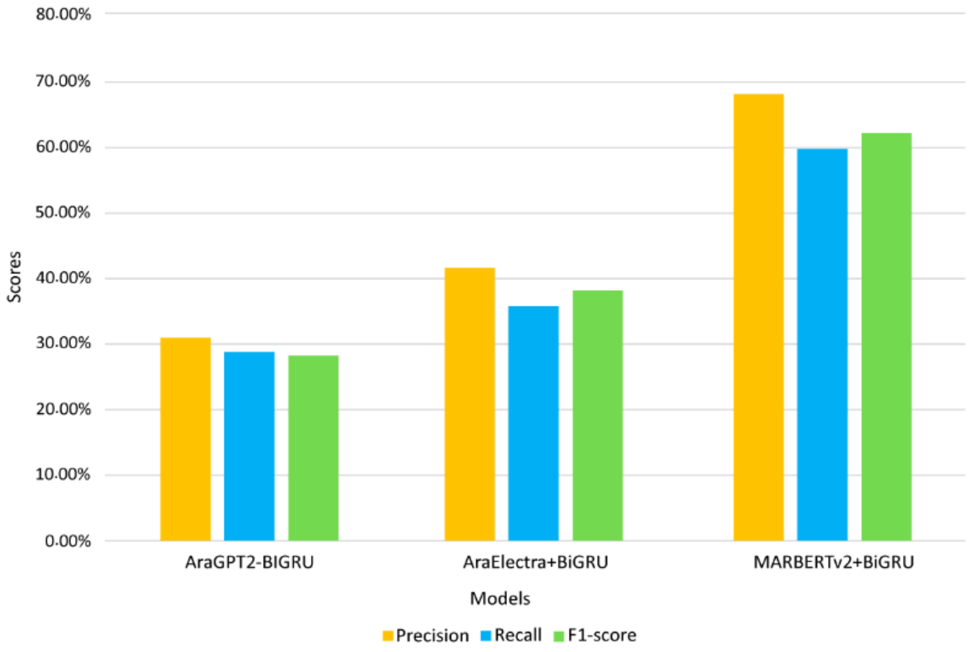

Figure 3. Comparison results of transformer-based models

5.2. Comparative analysis

The proposed model is compared with the following baseline models: i) Majority baseline [28]: It is the baseline model released with the OSCAT-5 shared task dataset; ii) MARBERTv2: the BERT model was fine-tuned with a linear layer and a SoftMax activation function; and iii) MARBERTv2-BiLSTM: the BiGRU layer is replaced with a BiLSTM layer to evaluate how it affects the model's performance. In addition to these baselines, results of the following related work models, evaluated on the same Arabic dataset as in this study, are also reported, including iCompass [29], Meta-Ai [30], AlexU-AIC [49], and UPV [50], and CLN [32]. A detailed description of these models can be found in section 2. The comparison results are shown in Table 4.

The results indicate that the proposed model outperformed the baseline model by an overall improvement of more than 47% in terms of F1-score. Besides, our model achieved better results over all related work models that were evaluated on the same dataset, with an enhancement of more than 8% over the best result reported in the literature by the iCompass model. Moreover, our proposed model outperformed prior models [49], [50] that adopted complex architectures like ensemble models and multi-task learning methods. This indicates the efficiency of our combined approach that exploits the strengths of both BERT and BiGRU model to encode relevant features for detecting hate speech in the input text.

On the other hand, MARBERTv2+BiGRU and MARBERTv2+BiLSTM models outperformed the MARBERTv2+linear model by more than 5% in F1-score, proving the efficiency of incorporating the fine-tuned MARBERTv2 model with more complex neural network layers to further encode context and semantic features. Additionally, our model achieved better results than the MARBERTv2-BiLSTM model. This can be justified thanks to the structure of GRU, which has a smaller number of parameters than LSTM, making the learning process easier.

Table 4. Comparative evaluation with existing models. The best results are marked in bold, while results with "*" were extracted from original papers

Model	Precision	Recall	F1-score
Baseline [28]	12.8%*	14.3%*	13.5%*
CLN [32]	53.42%*	22.03%*	25.25%*
ICompass [29]	54.8%*	53.1%*	52.8%*
Meta-AI [30]	55.1%*	50.8%*	51.9%*
AlexU-AIC [49]	49.0%*	47.0%*	47.6%*
UPV [50]	54.3%*	36.9%*	42.3%*
MARBERTv2+linear	58.28%	53.07%	53.09%
MARBERTv2+BiLSTM	63.09%	55.64%	58.51%
MARBERTv2+BiGRU (ours)	**67.86%**	**59.64%**	**61.68%**

On the other hand, the model's performance was affected by the data imbalance. Specifically, the training dataset contains imbalanced class proportions. When it comes to hate speech, the majority of instances, specifically 5,714 out of 6,675, are classified as "Not Hate". Additionally, for example, there are only two instances of hate speech based on disability, 27 instances of hate speech based on religion, and a considerably higher count of 260 instances of hate speech based on race. Therefore, additional efforts are required to address this problem and further enhance the achieved results for this task.

6. CONCLUSION

In this study, an enhanced model is proposed to handle the fine-grained hate speech detection task in Arabic language. Three different transformer-based models namely MARBERTv2, AraELECTRA, and AraGPT2 were examined to select the best model for this task. Additionally, we combined these models with a BiGRU layer to further improve the semantic and contextual features. The experimental results showed that the MARBERTv2 outperformed significantly the AraELECTRA and AraGPT2. Furthermore, the MARBERv2+BiGRU obtained the best results over the baseline and related work models, indicating the effectiveness of incorporating the BERT model with more complex neural network layers to encode context and semantic information that are relevant for the task of hate speech detection.

Future work directions include combining the BERT model with other neural network architectures to evaluate its impact on the overall results. In addition, we plan to explore other transformer-based models such as T5 and XLNET to handle this task. Moreover, we intend to investigate different methods to address the challenge of imbalanced dataset and further enhance the achieved results.

REFERENCES

[1] M. A. Paz, J. Montero-Díaz, and A. Moreno-Delgado, "Hate speech: a systematized review," *SAGE Open*, vol. 10, no. 4, Oct. 2020, doi: 10.1177/2158244020973022.

[2] K. Gelber and L. McNamara, "Evidencing the harms of hate speech," *Social Identities*, vol. 22, no. 3, pp. 324–341, May 2016, doi: 10.1080/13504630.2015.1128810.

[3] R. António, R. Guerra, and C. Moleiro, "Cyberbullying during COVID-19 lockdowns: prevalence, predictors, and outcomes for youth," *Current Psychology*, vol. 43, no. 2, pp. 1067–1083, Jan. 2024, doi: 10.1007/s12144-023-04394-7.

[4] H. Elzayady, M. S. Mohamed, K. M. Badran, and G. I. Salama, "Detecting Arabic textual threats in social media using artificial intelligence: an overview," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 25, no. 3, pp. 1712–1722, Mar. 2022, doi: 10.11591/ijeecs.v25.i3.pp1712-1722.

[5] S. Vilar-Lluch, "Understanding and appraising 'hate speech,'" *Journal of Language Aggression and Conflict*, vol. 11, no. 2, pp. 279–306, Sep. 2023, doi: 10.1075/jlac.00082.vil.

[6] R. Khezzar, A. Moursi, and Z. Al Aghbari, "arHateDetector: detection of hate speech from standard and dialectal Arabic Tweets," *Discover Internet of Things*, vol. 3, no. 1, Mar. 2023, doi: 10.1007/s43926-023-00030-9.

[7] R. Bensoltane and T. Zaki, "Aspect-based sentiment analysis: an overview in the use of Arabic language," *Artificial Intelligence Review*, vol. 56, no. 3, pp. 2325–2363, Mar. 2023, doi: 10.1007/s10462-022-10215-3.

[8] A. Ghosh Chowdhury, A. Didolkar, R. Sawhney, and R. R. Shah, "ARHNet - leveraging community interaction for detection of religious hate speech in Arabic," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 2019, pp. 273–280, doi: 10.18653/v1/P19-2038.

[9] A. Aref, R. Husni Al Mahmoud, K. Taha, and M. Al-Sharif, "Hate speech detection of Arabic shorttext," in *9th International Conference on Information Technology Convergence and Services (ITCSE 2020)*, May 2020, pp. 81–94, doi: 10.5121/csit.2020.100507.

[10] M. Sanguinetti, F. Poletto, C. Bosco, V. Patti, and M. Stranisci, "An Italian Twitter corpus of hate speech against immigrants," *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*, 2018.

[11] S. Assimakopoulos, R. V. Muskat, L. van der Plas, and A. Gatt, "Annotating for hate speech: the MaNeCo corpus and some input from critical discourse analysis," in *Proceedings of the 12th Language Resources and Evaluation Conference*, 2020, pp. 5088–5097.

[12] H. Mulki, H. Haddad, C. Bechikh Ali, and H. Alshabani, "L-HSAB: a levantine Twitter dataset for hate speech and abusive language," in *Proceedings of the Third Workshop on Abusive Language Online*, 2019, pp. 111–118, doi: 10.18653/v1/W19-3512.

[13] F. Husain, "OSACT4 shared task on offensive language detection: intensive preprocessing-based approach," in *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, 2020, pp. 53–60.

[14] M. Li, K. Yin, and M. Wang, "Ptr4BERT: automatic semisupervised Chinese government message text classification method based on transformer-based pointer generator network," *Advances in Multimedia*, vol. 2022, pp. 1–11, Aug. 2022, doi: 10.1155/2022/6540696.

[15] J. Gong *et al.*, "Hierarchical graph transformer-based deep learning model for large-scale multi-label text classification," *IEEE Access*, vol. 8, pp. 30885–30896, 2020, doi: 10.1109/ACCESS.2020.2972751.

[16] X. Wang, X. Xu, D. Huang, and T. Zhang, "Multi-task label-wise transformer for Chinese named entity recognition," in *ACM Transactions on Asian and Low-Resource Language Information Processing*, Apr. 2023, vol. 22, no. 4, pp. 1–15, doi: 10.1145/3576025.

[17] S. Silalahi, T. Ahmad, and H. Studiawan, "Transformer-based named entity recognition on drone flight logs to support forensic investigation," *IEEE Access*, vol. 11, pp. 3257–3274, 2023, doi: 10.1109/ACCESS.2023.3234605.

[18] R. Pan, J. A. García-Díaz, F. Garcia-Sanchez, and R. Valencia-García, "Evaluation of transformer models for financial targeted sentiment analysis in Spanish," *PeerJ Computer Science*, vol. 9, May 2023, doi: 10.7717/peerj-cs.1377.

[19] F. Wang *et al.*, "TEDT: transformer-based encoding–decoding translation network for multimodal sentiment analysis," *Cognitive Computation*, vol. 15, no. 1, pp. 289–303, Jan. 2023, doi: 10.1007/s12559-022-10073-9.

[20] M. Bilal, A. Khan, S. Jan, S. Musa, and S. Ali, "Roman Urdu hate speech detection using transformer-based model for cyber security applications," *Sensors*, vol. 23, no. 8, Apr. 2023, doi: 10.3390/s23083909.

[21] H. Saleh, A. Alhothali, and K. Moria, "Detection of hate speech using BERT and hate speech word embedding with deep model," *Applied Artificial Intelligence*, vol. 37, no. 1, Dec. 2023, doi: 10.1080/08839514.2023.2166719.

[22] M. U. Arshad, R. Ali, M. O. Beg, and W. Shahzad, "UHated: hate speech detection in Urdu language using transfer learning," *Language Resources and Evaluation*, vol. 57, no. 2, pp. 713–732, Jun. 2023, doi: 10.1007/s10579-023-09642-7.

[23] M. AbdelHamid, A. Jafar, and Y. Rahal, "Levantine hate speech detection in twitter," *Social Network Analysis and Mining*, vol. 12, no. 1, Dec. 2022, doi: 10.1007/s13278-022-00950-4.

[24] A. S. Fadel, M. E. Saleh, and O. A. Abulnaja, "Arabic aspect extraction based on stacked contextualized embedding with deep learning," *IEEE Access*, vol. 10, pp. 30526–30535, 2022, doi: 10.1109/ACCESS.2022.3159252.

[25] N. Alsaaran and M. Alrabiah, "Classical Arabic named entity recognition using variant deep neural network architectures and BERT," *IEEE Access*, vol. 9, pp. 91537–91547, 2021.

[26] R. Bensoltane and T. Zaki, "Combining BERT with TCN-BiGRU for enhancing Arabic aspect category detection," *Journal of Intelligent & Fuzzy Systems*, vol. 44, no. 3, pp. 4123–4136, Mar. 2023, doi: 10.3233/JIFS-221214.

[27] H. Mubarak, K. Darwish, W. Magdy, T. Elsayed, and H. Al-Khalifa, "Overview of OSACT4 Arabic offensive language detection shared task," in *Proceedings of the 4th Workshop on open-source arabic corpora and processing tools, with a shared task on offensive language detection*, 2020, pp. 48–52.

[28] H. Mubarak, H. Al-Khalifa, and A. Al-Thubaity, "Overview of OSACT5 shared task on Arabic offensive language and hate speech detection," in *Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection*, 2022, pp. 162–166.

[29] M. A. Bennessir, M. Rhouma, H. Haddad, and C. Fourati, "ICompass at Arabic hate speech 2022: detect hate speech using QRNN and transformers," in *Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection*, 2022, pp. 176–180.

[30] B. AlKhamissi and M. Diab, "Meta AI at Arabic hate speech 2022: multitask learning with self-correction for hate speech classification," in *Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection*, 2022, pp. 186–193.

[31] M. Abdul-Mageed and A. Elmadany, "ARBERT & MARBERT: deep bidirectional transformers for Arabic," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021, vol. 1: Long Pa, pp. 7088–7105.

[32] M. J. Althobaiti, "BERT-based approach to Arabic Hate speech and offensive language detection in Twitter: exploiting emojis and sentiment analysis," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 5, 2022, doi: 10.14569/IJACSA.2022.01305109.

[33] A. Al-Hassan and H. Al-Dossari, "Detection of hate speech in Arabic tweets using deep learning," *Multimedia Systems*, vol. 28, no. 6, pp. 1963–1974, Dec. 2022, doi: 10.1007/s00530-020-00742-w.

[34] M. Almaliki, A. M. Almars, I. Gad, and E.-S. Atlam, "ABMM: Arabic BERT-mini model for hate-speech detection on social media," *Electronics*, vol. 12, no. 4, Feb. 2023, doi: 10.3390/electronics12041048.

[35] A. Vaswani *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[36] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

[37] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, 2019.

[38] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," *arXiv preprint arXiv:2003.10555*, 2020.

[39] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems*, vol. 32, 2019.

[40] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[41] A. Figueroa, "Refining fine-tuned transformers with hand-crafted features for gender screening on question-answering communities," *Information Fusion*, vol. 92, pp. 256–267, Apr. 2023, doi: 10.1016/j.inffus.2022.12.003.

[42] S. Srivastava, B. Paul, and D. Gupta, "Study of word embeddings for enhanced cyber security named entity recognition," *Procedia Computer Science*, vol. 218, pp. 449–460, 2023, doi: 10.1016/j.procs.2023.01.027.

[43] H. Mubarak, S. Hassan, and S. A. Chowdhury, "Emojis as anchors to detect Arabic offensive language and hate speech," *arXiv preprint arXiv:2201.06723*, 2022.

[44] W. Antoun, F. Baly, and H. Hajj, "Araelectra: pre-training text discriminators for Arabic language understanding," *arXiv preprint arXiv:2012.1551*, 2020.

[45] W. Antoun, F. Baly, and H. Hajj, "Aragpt2: pre-trained transformer for Arabic language generation," *arXiv preprint arXiv:2012*, 2020.

[46] I. A. El-Khair, "1.5 billion words Arabic corpus," *arXiv preprint arXiv:1611.04033*, 2016.

[47] I. Zeroual, D. Goldhahn, T. Eckart, and A. Lakhouaja, "OSIAN: open source international Arabic news corpus-preparation and integration into the CLARIN-infrastructure," in *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, 2019, pp. 175–182, doi: 10.18653/v1/W19-4619.

[48] P. J. O. Suárez, B. Sagot, and L. Romary, "Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures," *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7), 2019: Leibniz-Institut für Deutsche Sprache*, 2019.

[49] A. Shapiro, A. Khalafallah, and M. Torki, "AlexU-AIC at Arabic hate speech 2022: contrast to classify," in *Proceedings of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection*, 2022, pp. 200–208.

[50] A. F. M. de Paula, P. Rosso, I. Bensalem, and W. Zaghouani, "UPV at the Arabic hate speech 2022 shared task: offensive language and hate speech detection using transformers and ensemble models," in *Proceedings of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection*, 2022, pp. 181–185.

BIOGRAPHIES OF AUTHORS

Rajae Bensoltane received her PhD in computer science (2023) from the University of Ibn Zohr, Morocco. She is an assistant professor at the Faculty of Sciences in Agadir. Her research interests include natural language processing, sentiment analysis, and information retrieval techniques. She is currently a member of laboratory of innovation in mathematics and intelligent systems. She can be contacted at email: r.bensoltane@uiz.ac.ma.

Taher Zaki is an associate professor and Vice Dean of the Faculty of Applied Sciences at Ibn Zohr University. He received his PhD in computer science from the University of Rouen, France in 2014. Dr. Zaki supervises several PhD theses in various research areas of computer science, including information retrieval, digital image processing, pattern recognition, text mining, data mining, and knowledge management. He is currently a member of laboratory of innovation in mathematics and intelligent systems and can be contacted at the following email address: t.zaki@uiz.ac.ma.