Comparing Deep Learning Models for Multi-label Classification of Arabic Abusive Texts in Social Media

Salma Abid Azzi and Chiraz Ben Othmane Zribi
National School of Computer Science, Manouba University, Tunisia

Keywords: Natural Language Processing, Multi-label Classification, Deep Learning, Arabic Language, Abusive Texts, Social Media, Convolutional Neural Networks, Recurrent Neural Networks.

Abstract: Facing up to abusive texts in social networks is gradually becoming a mainstream NLP research topic. However, the detection of its specific related forms is still scarce. The majority of automatic solutions cast the problem into a two-class or three-class classification issue not taking into account its variety of aspects. Specifically in the Arabic language, as one of the most widely spoken languages, social media abusive texts are written in a mix of different dialects which further complicates the detection process. The goal of this research is to detect eight specific subtasks of abusive language in Arabic social platforms, namely Racism, Sexism, Xenophobia, Violence, Hate, Pornography, Religious hatred, and LGBTQ[a] Hate. To conduct our experiments, we evaluated the performance of CNN, BiLSTM, and BiGRU deep neural networks with pre-trained Arabic word embeddings (AraVec). We also investigated the recent Bidirectional Encoder Representations from Transformers (BERT) model with its special tokenizer. Results show that DNN classifiers achieved nearly the same performance with an overall average precision of 85%. Moreover, although all the deep learning models obtained very close results, BERT slightly outperformed the others with a precision of 90% and a micro-averaged F1 score of 79%.

1 INTRODUCTION

The power of social media has leaped forward with the ever-growing number of users' generated data. The high variance of this data led progressively to the creation of new evaluation metrics. In fact, great content nowadays is not necessarily the one that provides valuable information but rather the one that creates interactions between the audiences such as comments, likes, shares... etc. Consequently, content generators who seek views and popularity tended to produce provocative posts as the easiest solution to stimulate all kinds of interactions, especially abusive text comments. This kind of practice, together with the freedom of expression and the anonymity offered by social platforms, further stimulated online abuse towards individuals and groups.

It is alarming nowadays how abusive texts are ubiquitous in several sub-forms. Spreading this kind of text is no longer confined to private conversations, but now invades all social media public sites. Some sub-forms, such as racism or sexism, target vulnerable groups while others are simply based on either distinguishing traits or trendy events. Examples include the rise of the COVID-19 pandemic where a spike of xenophobia against Asians was observed.

While manual solutions are still used to detect such behaviors on social platforms, their high risk of error and the enormous cost in terms of time consumption remain two major drawbacks. Therefore, it has proved essential to implement automatic detection techniques using Natural Language Processing and Artificial Intelligence solutions.

In the past years, it is worthy to mention that this issue was predominately coarse-grained since works generally performed binary or ternary classifications. Nowadays, researchers are tending to go deeper by investigating multiclass classification where an instance belongs to one class among four or more other classes. In our research, we consider transcending these classical classifications to examine at a finer level of granularity the phenomenon of abusive texts in Arabic social media. Eight different classes are investigated, namely: Racism, Sexism, Xenophobia, Religious hatred, Violence, Hate, Pornography, and LGBTQ Hate. In real life, these classes are non-

[a] The "LGBTQ" acronym stands for Lesbian, Gay, Bisexual, Transgender, and Queer or Questioning.

Azzi, S. and Zribi, C.
Comparing Deep Learning Models for Multi-label Classification of Arabic Abusive Texts in Social Media.
DOI: 10.5220/0011141700003266
In *Proceedings of the 17th International Conference on Software Technologies (ICSOFT 2022)*, pages 374-381
ISBN: 978-989-758-588-3; ISSN: 2184-2833

mutually exclusive: a random text may belong to two or more classes at the same time or to none of them. This implies that we are not performing a simple multiclassification but rather a multi-label classification.

As mentioned in prior work (Abid and Zribi, 2020), deep learning architectures have shown a strong learning capacity that led to their highly progressive and extensive use in English and Arabic texts' classification problems. They are continuously proving a high capacity to take up various NLP challenges. On this basis, we decided, in our work, to investigate and compare four famous models for classifying abusive multi-labeled texts in the Arabic language: These models are CNN, BiLSTM, BiGRU, and BERT.

Moreover, our study does not only exploit modern standard Arabic data but also covers dialectical Arabic with its variety of forms used in real life and social platforms. It is also noteworthy that the dataset we created to conduct our experiments has a multi-platform nature as it was retrieved from two different social networks, which are Twitter and YouTube.

This paper is organized as follows: In section 2, we go through the existing research on Arabic abusive texts detection in social networks. Section 3 explains the steps we took to prepare our multi-label dataset. Afterward, our baseline model is detailed together with the implemented deep learning architectures in section 4. Finally, we highlight and discuss the experimental results for future research in section 5.

2 RELATED WORKS

In this section, we review previous research studies on abusive language detection in Arabic social media. Works are classified by their level of granularity. Firstly, we introduce multiclass classification contributions which are relatively scarce, especially in the Arabic language. Then, related binary and ternary classification contributions are presented.

Multiclass Classification: Recent work by (Al-Hassan and Al-Dossari, 2021) aimed to classify Arabic tweets into 5 distinct classes: none, religious, racial, sexism, or general hate. As classes are mutually exclusive, authors in (Al-Hassan and Al-Dossari, 2021) defined the general hate class as ”Any general type of hate which is not mentioned in the previous classes. Whether it contains: general hatred, obscene, offensive and abusive words that are not related to religion, race or sex”. In the same work, the evaluation of various deep learning models showed that adding a layer of CNN to LTSM enhances the overall performance of detection with 72% precision, 75% recall, and 73% F1 score. (Duwairi et al., 2021) also took into consideration the existence of hate speech subtypes and created ArHS: A Multiclass Arabic Hate Speech Dataset. They followed a lexicon-based approach using Twitter4J API for crawling and relied on crowdsourcing for annotation. The final size of ArHS consisted of 9833 tweets classified into Misogyny, Racism, Religious Discrimination, Abusive, and Normal. To conduct their experiments, (Duwairi et al., 2021) additionally investigated the performance of two publically available datasets after reannotating them to fit the multiclass structure of ArHS. Binary, ternary, and multi-class classification were carried out on both ArHS and the combined Dataset. The CNN-LSTM and the BiLSTM-CNN architectures achieved both the best accuracy for multi-class classification with 73% and 65% respectively on ArHs and the combined dataset.

Two-class and Three-class Classification: (Abu Farha and Magdy, 2020) is the ”SMASH” team submission to the OSACT4: Open-Source Arabic Corpora and Corpora Processing Tools shared tasks on offensive language (Subtask A) and hate speech detection (Subtask B) in the Arabic language. The dataset provided contains 10,000 tweets split into training, development, and testing sets. Extremely imbalanced, only 19% of the tweets are tagged as offensive and 5% of the tweets are tagged as hate speech. The authors carried out various experiments covering a variety of approaches that include deep learning, transfer learning and multitask learning. Results showed that the multitask learning models achieved the best results with a macro F1 score of 0.904 for subtask A and 0.737 for subtask B. The same dataset was used in (Saeed et al., 2020) and (Hassan et al., 2020). (Saeed et al., 2020) named their own approach “ESOTP” (Ensembled Stacking classifier over Optimized Thresholded Predictions of multiple deep models). It is a classification pipeline where they trained NN, BLSTM, BGRU, and BLSTM+CNN 550 times. The predictions were used as a new training set for an ensemble of a Na¨ıve Bayes classifier, a Logistic Regression model, a Support Vector Machine, a Nearest Neighbours classifier, and a Random Forest. ESTOP achieved 87.37% F1 for subtask A (ranked 6/35) and 79.85% for subtask B (ranked 5/30). As for (Hassan et al., 2020), a system combination of Support Vector Machines (SVMs) and Deep Neural Networks (DNNs) achieved the best results for offensive language detection and ranked 1st in the official results with an F1-score of 90.51% while SVMs were more effective than DNNs for hate speech detection with F1-macro score of 80.63%. The same dataset was also used in (Aldjanabi et al., 2021) along with two other publically available datasets to develop a classification system using multi-task learning (MTL). Significant performance was achieved in the three of them.

Authors in (Faris et al., 2020) created their own Twitter dataset targeting the problem of hate speech in Arab countries. 3696 tweets were annotated to Hate, Normal, or Neutral, and promising results were achieved using a combination of CNN and LSTM with AraVec. Finally, (Alshalan and Al-Khalifa, 2020) first made sure that the dataset they created covers racist, religious, and ideological hate speech by choosing oriented keywords while retrieving data from Twitter. The resulting 9316 tweets were later labeled as normal, abusive, or hateful. Then, several neural network models were investigated based on CNN, RNN, and BERT. Results showed that CNN achieved the best performance of 79% as F1-score.

3 DATASET CONSTRUCTION

In our research, we decided to go deeper in investigating abusive language by considering eight of its various manifestations which are Racism, Sexism, Xenophobia, Religious hatred, Violence, Hate, Pornography, and LGBTQ Hate. We differentiate each by the targeted vulnerable groups and their intended characteristics like religious belief, gender, sexual orientation... etc. Unfortunately, abusive words in real life and social media are not necessarily classified under a single specific subform. As a matter of fact, abusive people generally believe they have all the right to discriminate against others whether on the basis of their races, genders, sexual orientation, or even all at once. The following example is a tweet expressing sexism, LGBTQ Hate, and religious hatred at the same time.

الله يلعنك يا ابليس، يعني مليت من روتينك وطلعت لنا جيل المثليين و النسويات. ما يكفي اليهود والكفار والملحدين !

”May Allah’s curse be upon you, Satan. You got tired of your routine so you made us a generation of gays and feminists. Jews, unbelievers, and atheists have not been enough for you!”

That’s why, we are dealing in this paper with a multilabel classification problem where a sample may be assigned to multiple labels or none of them, unlike multiclass classification problems where one and only one label can be assigned to each sample.

This section describes the process we went through to construct our dataset. To the best of our knowledge, it is the first multi-label, multi-platform, dialectical Arabic dataset so far.

3.1 Data Collection and Annotation

To perform the data collection, we used two famous social platforms which are Twitter and YouTube, each with a different extraction technique. Firstly, we retrieved data from Twitter using the application program interface (API) with a newly created developer account. We developed a python program to fetch the tweets based on a keyword approach. To ensure pertinence to our target application, we built the dedicated keyword list so that it covers all the classes in our work. Over 50 000 tweets were collected from all over the Arabic world.

As per the YouTube platform, a topic-based approach was applied with the main aim to create a cross-platform dataset that does not rely solely on a keyword-based search. In fact, most of the existing datasets restrict the extracted abusive texts to those containing explicit keywords and neglect the context-aware abusive ones. As mentioned in (Alakrot et al., 2018), YouTube represents a wide range of societal attitudes and thus is appropriate as a source for investigating the interaction between people. That’s why, we firstly selected trendy topics and events in the Arabic world that had triggered, in one way or another, abusive attitudes. Examples include the political conflict in the Tunisian Parliament between two feminine members where one of whom was brunette. This event evoked racist attitudes and interactions especially with the shared videos recording the entire incident. Videos calling for Jihadism belong also to the pool of content that stimulate violent comments and religious hatred attitudes. After that, we selected the relevant comments containing any of the abusive subforms to be included in our dataset.

After an extensive data cleaning from any irrelevant or redundant samples, we performed manual annotation for the 6000 texts. We decided to conduct a two-steps procedure since we are dealing with a multilabel classification problem. The first step consists in deciding whether the text is abusive or not, while the second step is to assign the corresponding eight labels only to the samples already annotated as abusive. In table 1, we provide what constitutes each of the subclasses in our research.

1914 out of the 6000 lines (31%) were labeled as Normal while the rest is marked as abusive. As shown in figure 1, the instances belonging at least to one of the abusive subclasses exceed those that do not belong to any of them. The percentage of the clean instances for each subclass is then depicted in figure 2.

Table 1: General overview of the dataset along with the classes' description.

Class	Description
Racism	When a text contains any kind of discrimination against others based on their race.
Sexism	Any prejudice based on one's sex or gender.
Religious hatred	It is the manifestation of hatred towards persons by reference to their religious belief or lack of religious belief.
Xenophobia	Any dislike or hatred of a group of persons belonging to a different country, region, or tribe.
Violence	All sort of text that contains any kind of glorification or incitement to commit violent acts.
Hate	Threatening and hateful speech through which a person intends to humiliate, discriminate, or express hatred towards others.
Pornography	The depiction of any sexual content that may also promote pedophilia or sexual violence.
LGBTQ Hate	Any kind of texts that express violence against lesbian, gay, bisexual, transgender, and queer (LGBTQ) individuals.

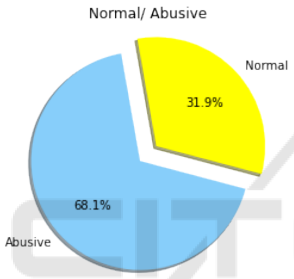

Figure 1: The percentage of normal and abusive texts in the dataset.

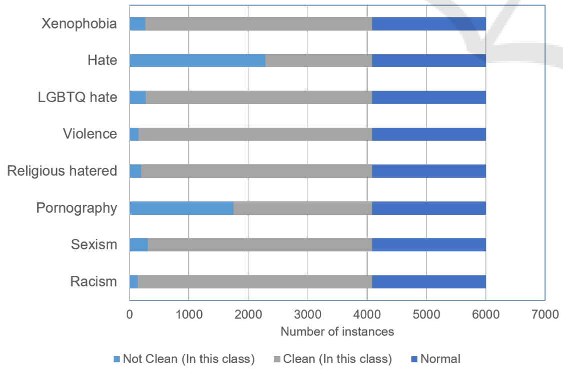

Figure 2: Labels distribution in the dataset.

3.2 Preprocessing

In order to feed a coherent form of data as an input to our models, we firstly performed several preprocessing steps, which are vital when dealing with quite noisy and colloquial texts from social media. It is further crucial for the Arabic language written in different dialects and known by its inherited complexity. The following are the steps we performed:

- Removing Arabic and English punctuations, URLs, mentions, emails, dates, numbers, and all emojis.
- Eliminating Arabic diacritics (Fatha أَ, Kasra إِ, Damma أُ .. etc.) as well as the consecutive repeated characters to be replaced with only one character.
- Extracting non-Arabic letters
- Performing the normalization Step: We replaced the(آ ، آ ؛ إ ، أ ، أ ، أ) with (ا) and the (ؤ ، ئ ي ، ة ،) with (ى ، ه ، و ، ى).

3.3 Word Embeddings

In this step, we need to transform our textual data to a particular form understandable by our neural network models. To do so, we used word embeddings. It is the term describing words' distributed representation as feature vectors. Each and every word has its meaningful representation as a vector with a particular dimension so that words that are similar in meaning are close to each other in the vector space. As indicated in (Faris et al., 2020), embedding means a dense vector, where its length is a parameter that is set previously and the components of the dense vector are parameters that are learned during the training process. A higher dimension of the embedding corresponds to a better ability for learning the semantic meaning of words, but also the need for more large training data (Faris et al., 2020). One of the most efficient techniques of learning word embeddings is Word2Vec

(Mikolov et al., 2013) which has two implementation models: The Continuous Bag-of-Words (CBoW) and the Skip-Gram (SG) models. The CBOW model learns the embeddings by predicting the current word using the context as an input, while the continuous skip-gram takes the current word as input and learns the embeddings by predicting the surrounding words (Mikolov et al., 2013).

With regard to the Arabic language, a number of pre-trained word vectors are currently available among which we cite AraVec (Mohammad et al., 2017). It is a set of pre-trained distributed word embedding trained on three datasets (Wikipedia, Twitter, and texts from Arabic web pages) with the two models ofword2vec (CBOW and SKIP-G). The one used in our experiment is Aravec SkipGram 300D-embeddings trained on tweets as the majority of our dataset is from Twitter.

4 DEEP LEARNING MODELS

In this section, we describe the experiments we conducted to compare deep learning models for the detection of abusive languages in Arabic social media. We fully outline all the architectures we adopted for each model after tuning the hyperparameters to reach optimum performance.

4.1 Convolutional Neural Network (CNN)

It is a class of deep neural networks that was initially used in image recognition and processing. Yet, CNN models have subsequently been shown to be effective for NLP and have achieved excellent results (Kim, 2014). With an architecture inspired by neurons in human and animal brains, the main benefit of CNN compared to its predecessors is that it automatically identifies the relevant features without any human supervision (Gu et al., 2017). The key components in this architecture are the convolutional, pooling, and Fully Connected layers. In our work, we used our 300 dimensioned embedding layer as an input to a one-dimensional convolutional layer with 64 filters and a kernel size equal to 3. Max pooling is also used to extract the most important features. Finally, we applied a fully-connected layer composed of 128 neurons. For the experiments, we empirically chose 5 epochs and set the size of the batch to 32.

4.2 Bidirectional Long Short-Term Memory (BiLSTM)

While RNNs are characterized by their ability to remember information from previous activations and thus being suited for the context dependencies, LSTMs, as a special type were designed to improve on this ability. As a matter of fact, Long short-term memory networks are quite powerful at keeping in memory activations from the long-term past to learn from that context. Still, future information might also be important to capture. For instance, when working with textual data, words situated after the current one would give the complete idea about the semantic correlation in the sentence. Based on this idea, the bidirectional LSTMs (BiLSTM) were designed. It consists of two LSTMs going in opposite directions.
The input to our architecture is a 300 embedding layer attached to two parallel blocks of Bidirectional Long-Short Term Memory with 128 units. A flatten layer and fully-connected layer of 128 neurons are then attached with 20% dropout regularization.

4.3 Bidirectional Gated Recurrent Units (BiGRU)

Gated Recurrent Unit (GRU) is an RNN variant proposed by (Cho et al., 2014). Its design is quite similar to LSTM with more simple calculations and implementations. Its aim is to shorten the training process and solve the vanishing gradient problem. In our experiments, we simply replaced the Bidirectional Long-Short Term Memory blocks with Bidirectional Gated recurrent units of the same size. The 20% dropout regularization and sigmoid activation layer were also conserved at the end.

The same following settings were applied to all the architectures:

Loss function: Binary-crossentropy
Optimization: Adam
Regularisation: Dropout
Activation function: ReLu

4.4 BERT Model

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) is a language transformation model trained on a large amount of data. It is a novel approach that has shown state-of-the-art results on various NLP problems. BERT is mostly distinctive by its deep bidirectionality which means that the model learns information from left to right and from right to left at once thanks to the

Masked Language Model (MLM). Based on the transformers architectures that include encoders and decoders, BERT makes use of only the encoder mechanism. It involves two different steps which are: Pre-training and Fine-tuning. BERT was trained on unlabelled data. For finetuning, the BERT model is first initialized with the pre-trained parameters and all of the parameters are fine-tuned using labeled data from the downstream tasks (Devlin et al., 2019). Moreover, the BERT training procedure of word embedding differs from other word embedding models. Unlike other deep learning models, it has additional embedding layers in the form of Segment Embeddings and Position Embeddings.

During the experiment, we used the base version which contains 12 encoder layers stacked on top of each other. We tried to conserve the same parameters as deep neural networks to obtain an effective comparison: A dropout of 20% and a sigmoid activation layer were applied. For the training, 5 epochs were chosen empirically along with a batch size of 32.

5 EXPERIMENTAL RESULTS AND DISCUSSION

When it comes to performing multilabel classification experimentations, a single instance prediction becomes a set of labels that may be fully correct, partially correct (with different levels of correctness) or fully incorrect (Sorower, 2022a). The concept of partially correct, which does not exist in a classical binary problem, directly affects the evaluation strategy depending on the actual problem. At some point, we need to decide on what basis are we going to evaluate the performance of our model. Is it up to which number of classes can it correctly classify or on the basis of its ability to predict all the labels correctly? In the following, we are going to present the most common metrics that are used in multilabel classification followed by a depiction of our experiment's results based on a set of chosen metrics. Finally, a discussion about the obtained results is presented.

5.1 Evaluation Metrics

Exact Match Ratio: One trivial way around would be just to ignore partially correct (consider them as incorrect) and extend the accuracy used in single label case for multi-label prediction. This is called Exact Match. Exact Match is the strictest metric, indicating the percentage of samples that have all their labels classified correctly (Sorower, 2022a).
Accuracy: Accuracy for each instance is defined as the proportion of the predicted correct labels to the total number (predicted and actual) of labels for that instance. Overall accuracy is the average across all instances (Sorower, 2022b).
Precision: It computes the fraction of the correctly predicted labels by the total number of labels. Then, it averages it by all instances.
Recall: It is the proportion of the True positives (correct classification) to the total number of predicted labels, averaged over all instances.
F1 Measures: It determines the harmonic mean between precision and accuracy. To obtain a single score, we need to average all the F1 scores per class using the following techniques: Macro averaging, weighted averaging, or micro averaging. The first one is very straightforward: It computes the mean of all the F1-scores by the number of classes without taking into account any other factor. The weighted averaging F1 multiplies each F1-score by a support value that refers to the number of actual occurrences of the class in the dataset. Finally, micro-averaged F1-Score looks at all the samples together. It is a global proportion of true positives out of all observations.

Due to the enormous imbalance in our dataset, computing accuracy would be insignificant to evaluate a model's performance. Its use implies that false negatives and false positives have equal costs. It is not the case for imbalanced data. It is always very accurate to predict the majority class which may lead to erroneous conclusions. For instance, if 95% of the data belongs to class A, and the model predicts that all the data belongs to that class, then our model is 95% accurate. The metric in this case is misleading. Besides, the use of " Exact Match Ratio" ignores the classifiers' chance of being partially correct. It is a strict metric with an assumption that the good model is only the one that correctly predicts all the classes simultaneously. That's why, our classification results will be presented in terms of precision, recall, and micro-averaged F1-Score.

5.2 Baseline Model

Detecting abusive language in social networks is a task that has been mostly framed as a supervised machine learning problem before the emergence of deep learning architectures (Abid and Zribi, 2020). Great results were achieved with various classifiers including Support Vector Machine (SVM) and Naïve Bayes (NB). That was our starting point to create a machine learning model to serve as a baseline. The aim is to compare it with the intended deep learning methods and see how much the performance would be improved. Based on previous findings, the SVM and NB

Table 2: Overall classification results.

Model Metric	SVM	CNN	BiLSTM	BiGRU	BERT
Precision	0.54	0.83	0.82	0.85	**0.90**
Recall	0.64	**0.73**	0.72	0.72	0.71
Micro-averaged F1-Score	0.57	0.78	0.77	0.78	**0.79**

were chosen as classifiers and the TF-IDF and AraVec as feature extraction techniques. In order to train our models, we preserved 80% of the dataset for training and 20% for testing. The best performance was achieved by the SVM classifier with TF-IDF (Precision: 54% and F1 Measure = 57%).

5.3 Results and discussion

Table 2 shows the precision, recall, and macro-average F1 results for the multilabel classification task using SVM, CNN, BiLSTM, BiGRU, and BERT classifiers. it is obviously clear that deep neural networks with Aravec Word embeddings and BERT transformer model with its special tokenizer outperform the SVM approach using Aravec and TF-IDF (Term Frequency-Inverse Document Frequency) feature extraction technique.

DNN-based classifiers, which are in our case CNN, BiLSTM, and BiGRU achieved nearly the same level of performance. The recall is relatively similar with a bit better achievement for the CNN model against all classifiers. The best micro-averaged F1-Score among DNN classifiers was also given by BiGRU and CNN (0.78) while the best precision value of 0.85 was achieved by BiGRU. From table 2, it can also be seen that the BERT fine-tuning approach achieved the best performance in terms of precision and micro-averaged F1-Score. Compared to feature-based approaches, an improvement of more than 17% has been obtained.

In figure 3, we present the confusion matrix for the classification of one class using BERT which is ”Hate”. Numbers show that confusion is mainly occurring in this class where the number of False positives and false negatives is the highest. This may be caused by the ambiguous definition of the class that misleads the annotation process. We notice also, from our error analysis with all the confusion matrices, that ”false negative” is the error happening very often with the different labels. We assume it is due to the imbalance in class distribution within the dataset. To overcome such a problem, a lot of solutions can be used such as data sampling and data augmentation. However, when it comes to multi-labeled datasets, the task becomes more challenging due to the labels’ correlation.

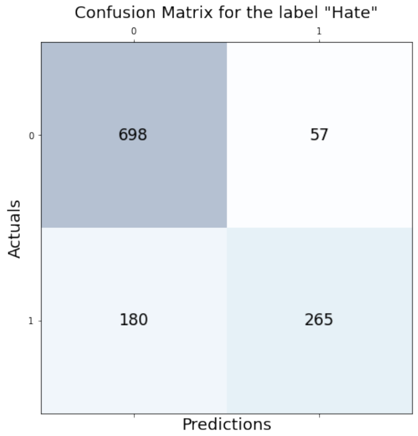

Figure 3: Confusion matrix for the label ”Hate”.

6 CONCLUSION

In the research presented in this paper, we aimed to investigate various state-of-the-art models for detecting abusive language in Arabic social media. Solving such a problem is crucial to protect the users from the dangerous effects that may occur. The performance of different deep learning architectures was evaluated which are the CNN, the bidirectional LSTM, the bidirectional GRU, and BERT. The final results firstly show that all of the models largely surpass the Support Vector Machine (SVM) traditional model that we previously used as our baseline. On the other hand, BERT with its own features extraction technique and its classification procedure gives the best results in terms of precision (90%) and micro-averaged F1-Score (79%).

We strongly believe that the current work can be improved in several ways in the future. First of all, we aim to enrich our dataset and explore different techniques in order to address classes imbalance. Also, we will consider extending our experiments by carrying them out on other available datasets. Finally, additional features can also be fed to the models such as the users’ geographical location and the texts’ time of posting.

In our future work, we will investigate more hybrid deep learning models based on CNN and RNN variants. In these hybrid networks, we will be focusing on taking advantage of each structure's strong specificities to see if it would improve our classification results. We are planning, as well, to use other word embedding techniques like FastText and Glove which proved to be strongly effective in recent contributions. Finally, we will also focus on experimentations among BERT models and explore the result of using its recent variants like AraBERT on the detection of abusive language in Arabic.

REFERENCES

Abid, S. and Zribi, C. (2020). From machine learning to deep learning for detecting abusive messages in arabic social media: Survey and challenges.

Abu Farha, I. and Magdy, W. (2020). Multitask learning for arabic offensive language and hate-speech detection.

Al-Hassan, A. and Al-Dossari, H. (2021). Detection of hate speech in arabic tweets using deep learning. *Multimedia Systems*.

Alakrot, A., Murray, L., and Nikolov, N. (2018). Dataset construction for the detection of anti-social behaviour in online communication in arabic. *Procedia Computer Science*, 142:174–181.

Aldjanabi, W., Dahou, A., Al-qaness, M. A. A., Elsayed Abd Elaziz, M., Helmi, A., and Damasevicius, R. (2021). Arabic offensive and hate speech detection using a cross-corpora multi-task learning model. *Informatics*, 8:69.

Alshalan, R. and Al-Khalifa, H. (2020). A deep learning approach for automatic hate speech detection in the saudi twittersphere. *Applied Sciences*, 10:8614.

Cho, K., Merrienboer, B., Gulcehre, C., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805.

Duwairi, R., Hayajneh, A., and Quwaider, M. (2021). A deep learning framework for automatic detection of hate speech embedded in arabic tweets. *Arabian Journal for Science and Engineering*, 46:1–14.

Faris, H., Aljarah, I., Habib, M., and Castillo, P. (2020). Hate speech detection using word embedding and deep learning in the arabic language context. pages 453–460.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, L., Wang, G., Cai, J., and Chen, T. (2017). Recent advances in convolutional neural networks.

Hassan, S., Samih, Y., Mubarak, H., Abdelali, A., Rashed, A., and Chowdhury, S. (2020). Alt submission for osact shared task on offensive language detection. Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools:11–16.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26.

Mohammad, A. B., Eissa, K., and El-Beltagy, S. (2017). Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, 117:256–265.

Saeed, H. H., Calders, T., and Kamiran, F. (2020). Osact4 shared tasks: Ensembled stacked classification for offensive and hate speech in arabic tweets. In *OSACT*.

Sorower, M. (2022a). A literature survey on algorithms for multi-label learning.

Sorower, M. (2022b). A literature survey on algorithms for multi-label learning.