Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, pages 97–101
with a Shared Task on Offensive Language Detection.
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Multi-Task Learning using AraBert for Offensive Language Detection

Marc Djandji, Fady Baly, Wissam Antoun, Hazem Hajj
American University of Beirut
{mgd10, fgb06, wfa07,hh63}@aub.edu.lb

Abstract

The use of social media platforms has become more prevalent, which has provided tremendous opportunities for people to connect but has also opened the door for misuse with the spread of hate speech and offensive language. This phenomenon has been driving more and more people to more extreme reactions and online aggression, sometimes causing physical harm to individuals or groups of people. There is a need to control and prevent such misuse of online social media through automatic detection of profane language. The shared task on Offensive Language Detection at the OSACT4 has aimed at achieving state of art profane language detection methods for Arabic social media. Our team "BERTologists" tackled this problem by leveraging state of the art pretrained Arabic language model, AraBERT, that we augment with the addition of Multi-task learning to enable our model to learn efficiently from little data. Our Multitask AraBERT approach achieved the second place in both subtasks A & B, which shows that the model performs consistently across different tasks.
Keywords: Offensive Language, Hate Speech, AraBERT, Multilabel, Multitask Learning

1. Introduction

Offensive language, including hate speech, is a violent behavior that is becoming more and more pervasive across public social media platforms (Fosler-Lussier et al., 2012). Hate speech was found to negatively impact the psychological well-being of individuals and to deteriorate inter-group relations on the societal level (Tynes et al., 2008). As such, detection and prevention mechanisms should be setup to deal with such content. Machine learning algorithms can be employed to automatically detect these behaviors by relying on recent techniques in natural language processing that have shown propitious performance.

A small number of works targeted the problem of simultaneously detecting both hate and offensive speech in Arabic. For example, Haddad et al. (2019) targeted the problem of hate and offensive speech detection for the Tunisian dialect using Support Vector Machine (SVM) and Naive Bayes classifiers trained on hand crafted features. Mulki et al. (2019) targeted the detection of profane language for the Levantine dialect using SVM and NB models trained on hand-crafted features. Although these works provided insights into the features that could be used for Arabic hate and offensive speech detection and introduced datasets for these specific dialects, they are limited to these specific dialects and do not target the problem of developing models that can learn efficiently with little data.

In the 4th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT4) (Mubarak et al., 2020) the shared task on offensive language aimed at offensive and hate speech detection in Arabic tweets. The task is split up into two Subtasks: Subtask A) which aimed at detecting whether a tweet is offensive or not and Subtask B) which aimed at detecting whether a tweet is hate-speech or not. The organizers labeled a tweet as offensive if it contained explicit or implicit insults directed towards other people or inappropriate language. While a tweet labeled as hate speech contains targeted insults towards a group based on their nationality, ethnicity, gender, political or sport affiliation. Each subtask is evaluated independently with a macro-F1 score. The dataset had the following issues that also needed to be addressed: (*i*) The labeled tweets were written in dialectal Arabic which had inconsistent writing style and vocabulary (*ii*) The class labels were highly imbalanced especially in the hate speech case where only 5% of the data was labeled as hate speech.

The models that we experimented with are all based on fine-tuning the Arabic Bidirectional Encoder Representation from Transformer (AraBERT) model (AUBMind-Lab, 2020) with different training classification schemes. To enable the model to learn from little data and not overfit to the dominant class, we train AraBERT in a multitask paradigm. Our contributions can be summarized as follows:

- Comprehensive evaluation including the impact of different sampling techniques and weighted loss functions that penalizes wrong predictions on the minority class in an attempt to balance the data.
- Propose a new model that combines AraBert and multi-task learning to achieve accurate predictions and address data imbalance.
- Propose a model that provides consistent performance on both hate and offensive speech detection with the presence of different Arabic dialects.

The rest of the paper is organized as follows: Section 2. reviews related work on offensive and hate-speech detection. In section 3., we provide details on our models. Section 4.3. provides and discusses the results of the conducted experiments. A conclusion of the work is presented in Section 5.

2. Related Work

2.1. Hate and Offensive Speech Detection in English

Hate Speech Detection Schmidt and Wiegand (2017) concluded that the most used models are Support Vector Machine (SVM) and Recurrent Neural Network (RNN) variant. The most used features are surface features such as bag of words, word and character n-gram, word generalization features such as word embeddings, and reported that lexicon features are usually used as a baseline. Waseem

and Hovy (2016) investigated the usefulness of different features for hate speech detection, where they found that among character n-gram, gender, and location features, a combination of character n-gram and gender features yields the best macro-F1 score. Recently, different competitions has been organized to accelerate the development of accurate hate speech detection models. For example, HateEval competition (Basile et al., 2019) targeted the problem of detecting hate speech directed towards women and immigrants in English and Spanish tweets. The winning team in English achieved a 65.1% macro-F1 score using an SVM classifier with an RBF kernel trained on Universal Sentence Encoder embeddings (Cer et al., 2018). Mandl et al. (2019) organized a competition for hate speech detection in Hindi, English, and German, where the winning team for English hate speech detection has used a Long-short term memory (LSTM) with attention model.

Offensive Speech Detection Offensive speech detection was the topic of interest in the last offenseval competition (Zampieri et al., 2019), where it was shown that BERT (Devlin et al., 2018) trained with two epochs and 64 maximum sequence length achieved the first place outperforming Convolutional Neural Network (CNN), LSTM and SVM baselines and an ensemble of LSTM and Bidirectional-Gated Recurrent Unit (Bi-GRU) on word2vec embeddings.

Hate and Offensive Speech Detection Few works in the literature have targeted the problem of detecting hate and offensive tweets. Davidson et al. (2017) provided a dataset that contains both hate and offensive examples. They proposed the use of a combination of (bi, uni, tri)-gram features weighted by TF-IDF, lexicon sentiment score for each tweet, and Flesch-Kincaid grade level and Flesch Reading Ease scores. It was found that logistic regression with the L2 norm provided the best results among other shallow classifiers.

In summary, The most used features in the literature are character and word n-gram, TF-IDF feature weighing, Flesch-Kincaid grade, and ease of reading scores, word embeddings. The most popular classifiers in the literature are SVM, Logistic regression, LSTM, CNN, GRU, BERT. The best performing models are BERT and SVM with RBF kernel on sentence embeddings for offensive and hate-speech detection, respectively. The current work does not address the problem of providing a model that can learn efficiently from little data.

2.2. Hate and Offensive Speech Detection

Hate Speech Detection An extensive overview of the different works on hate speech detection was done by (Al-Hassan and Al-Dossari, 2019), but very few works in the literature target the problem of Arabic hate speech detection. Albadi et al. (2018) introduced the first dataset containing 6.6K Arabic hate-speech tweets targeting religious groups. The authors compared a lexicon-based classifier, SVM classifier trained with character n-gram features, and a Deep Learning approach consisting of a GRU trained on AraVec embeddings (Soliman et al., 2017). The GRU approach outperformed all other approaches with a 77% F1 score.

Offensive Speech Detection For offensive speech detection in Arabic, different approaches can be found in the literature. Alakrot et al. (2018), introduced a dataset for offensive speech in Arabic collected from 15K YouTube comments. For classifying the different comments, the data was preprocessed by removing stop words and diacritics, correcting misspelled words, then tokenization and stemming was performed in order to extract features that are used by a binary SVM classifier. Mohaouchane et al. (2019), explored the use of different Deep Learning architectures for offensive language detection. AraVec embeddings of each comment were used to train several models: CNN-LSTM, CNN-BiLSTM with attention, Bi-LSTM, and CNN model on the dataset proposed in (Alakrot et al., 2018) where the CNN model was found to provide the best F1 score. In Mubarak and Darwish (2019) 36 million tweets were collected and used it to train a FastText deep learning model and SVM classifier on character n-gram features where it was found that the Arabic FastText DL model provided the best results.

Hate and Offensive Speech Detection A very limited number of works targeted the problem of detecting both hate and offensive speech in Arabic. Haddad et al. (2019) created a dataset of 6K tweets containing hate and offensive speech in the Tunisian dialect. For binary (offensive, non-offensive) and multi-class (offensive, hate, or normal) classification of hate and offensive speech, the authors extracted several n-gram features from each tweet and applied Term Frequency (TF) weighing to select the most effective features. The extracted features were then used to develop an SVM and Naive Bayesian (NB) classifiers. The NB classifier provided superior performance with 92.3% and 83.6% F1 scores for binary and multi-class classification, respectively. Mulki et al. (2019) introduced a dataset of 6K tweets containing hate and offensive speech in the Levantine dialect. Similar to (Haddad et al., 2019), they extracted n-gram features with TF weighing and used the features to develop an SVM and NB classifiers. The NB classifier was found to be superior.

In summary, The most used features in the literature are character n-gram, stemming, and tokenization. The most popular classifiers in the literature are SVM, NB, LSTM, CNN, GRU. The best performing systems employed a CNN model and AraVec embeddings for offensive speech detection and a GRU model on AraVec embeddings for hate-speech detection. Very little work can be found in the literature for Arabic hate and offensive speech detection. The current work does not address the multiple dialects and little data challenges for these tasks.

3. Proposed Models

We based our approaches on the recently released AraBERT model. AraBERT is a Bidirectional representation of a text sequence, pretrained on a large Arabic corpus that achieved state of the art performance on multiple Arabic NLP tasks. Our best model is based on augmenting AraBERT with Multitask Learning, which solves the data imbalance problem by leveraging information from multiple tasks simultaneously. We also compare our best model

with other approaches that are used to solve class imbalance issues such as balanced batch sampling and Multilabel classification.

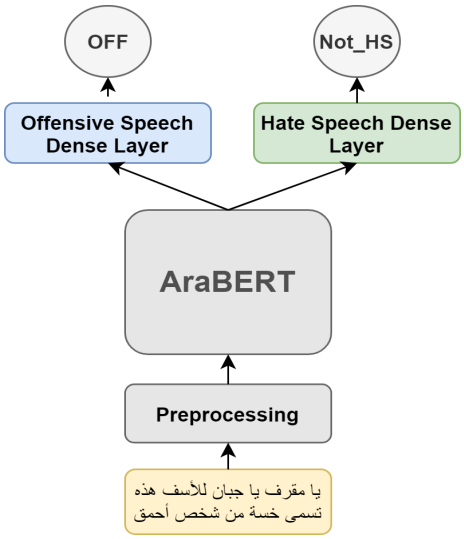

Figure 1: The trained Multitask Learning model given an input offensive tweet

3.1. Multitask Learning (MTL)

Multitask Learning is a learning paradigm that endows the developed models with the human-like abilities of transferring the important learned information between related tasks in what is called inductive transfer of knowledge under the assumption that commonalities exist between the learned tasks. Furthermore, the main advantages of MTL are that it reduces the requirements for large amounts of labeled data, improves the performance of a task with fewer data by leveraging the shared information from the related tasks with more data, and enables the model to be robust to missing observations for some tasks (Caruana, 1997; Qiu et al., 2017). Given that little data is available for both hate and offensive classes, we use an MTL approach to augment the initial AraBERT model such that it can learn both tasks simultaneously, which reduces the overfitting effect induced by the dominant not offensive and not hate examples. Our MTL-Arabert model consists of two components as can be seen in Figure 1: a part that gets trained by all the tasks' data in order to extract a general feature representation for all the tasks and a task-specific part that gets trained only by the task-specific examples to capture the task-specific characteristics.

1. Shared Part: Contains the pretrained AraBert model that gets tuned by the combined loss of both tasks in order to learn a shared set of information between both tasks

2. Task-specific layers: These consist of a task-specific dense layer that are dedicated to extracting the unique information per task.

3.2. Other Approaches

Multilabel Classification Multilabel classification is the task of classifying a single instance with multiple labels. We considered using this approach for two main reasons. Firstly, the subtasks are very coherent as they both try to solve problems that behaviorally fall under the same general idea, detecting violent behaviors. Secondly, considering that subtask B has very little hate speech labeled data and that all hate speech data is also labeled as offensive, we assumed that a multilabel classifier would help leverage and provide a better understanding of the hate speech instances as they are being trained simultaneously with the offensive instances. We also explored oversampling the Task B instances and made sure that each training batch included samples of hate speech data.

Weighted Cross-Entropy loss Cross-entropy loss is useful in classification tasks, since the loss increases as the predicted probability diverges from the actual label. The Weighted version, penalizes each class differently, according to the given weight. The weighted cross-entropy loss of a class i with weight W_i is shown in 1, the weight vector is given in 2

$$\mathcal{L}(x_i) = -W_i \log\left(\frac{\exp(x_i)}{\sum_j \exp(x_j)}\right) \quad (1)$$

$$W_i = \frac{N^o Samples}{N^o Classes \times Count(i)} \quad (2)$$

Balanced batch sampling We re-sample the dataset in such a way that we under-sample the majority class and over-sample the minority class at the same time. Which reduces information loss due to under-sampling, and minimizes overfitting due to over-sampling, since the over/under-sampling is done to a lesser extent compared to independently implementing over/under-sampling.

4. Experiments

4.1. Data Description

The dataset for both tasks is the same containing 10K tweets that were annotated for offensiveness with labels (OFF or NOT_OFF) and hate speech with labels (HS or NOT_HS). The data was split by the competition organizers into 70% training set, 10% development set, and 20% test set. Table 1 shows the data distribution among the different labels and splits. By examining Table 1, it can be seen that the data is very imbalanced having only 5% of the examples labeled as hate speech and 20% of the examples labeled as offensive in the training dataset, which makes the tasks much harder and calls for methods that can learn efficiently from little data.

Table 1: The data distribution for both tasks. The first two rows show the class distribution of task A. The second two rows show the class distribution of task B

Class	Training	Developement
NOT_OFF	5468	821
OFF	1371	179
NOT_HS	6489	956
HS	350	44

4.2. Preprocessing

For preprocessing the data, we tokenized Arabic words with the Farasa Arabic segmenter (Abdelali et al., 2016) so that the input would be compatible with the AraBERT input. For example, "المدرسة - *Almadrasa*" becomes "ال + مدرس + ة - *Al+ madras +T*". We also removed all mentions of the user tokens "USER", retweet mentions "RT USER:", URL tokens, the "<LF>" tokens, diacritics, and emojis. As for hashtags, we replaced the underscore within a hashtag "_" with a white space to regain separate understandable tokens, and we pad the hashtag with a whitespace as well. For instance, "أبو_ظبي" turns into "#أبو ظبي". We should also mention that these preprocessing steps are precisely applied to all the experiments conducted for both subtasks.

4.3. Results

Both tasks were evaluated using the unweighted-average F1 of all classes, which is the macro-F1 score. Given the high imbalance in the dataset and that the macro-F1 score is penalized by the minority class, achieving a high macro-F1 score is challenging. Table 2 and 3 provide the results of our models on the development and test set, respectively. All three models were trained on the whole training set for five epochs with a batch size of 32 and a sequence length of 256 in a GPU-accelerated environment. The epoch-model that achieved the highest macro-F1 score on the dev-set is reported in Table 2.

Table 2: The performance of the different approaches on the development set for both tasks using the Macro-F1 score metric. It can be seen that the Multitask approach outperforms all other approaches

Model	Macro-F1	
	Offensive Language	Hate Speech
AraBERT	89.56	80.60
AraBERT-S*	87.24	79.42
AraBERT-W**	88.17	79.85
AraBERT-SW***	90.02	78.13
Multilable AraBERT	89.41	79.83
Multilable AraBERT*	89.55	80.81
Multitask AraBERT	**90.15**	**83.41**

* AraBERT with balanced batch sampling
** AraBERT with weighted loss
*** AraBERT with both balanced batch sampling and weighted loss

Table 3: The performance of the Multitask Learning (MTL) model on the test set for both tasks using the Macro-F1 score metric.

Model	Task A: Macro-F1	Task B: Macro-F1
Multitask AraBERT	90	82.28

We only show the results of our best MTL model on the test data in Table 3 as provided by the competition organizers. Our Multitask approach shows consistent performance on both the dev and test sets across both tasks. The results show that training both tasks jointly in a Multitask setting improves the model generalizability with the presence of little data for each task. The results for the hate speech task are not as good as the offensive language task due to the minimal number of hate speech training examples, which constitute 5% of the training data. Although when combined, balanced batch sampling and weighted loss achieved the second best results on task A. When used separately, both approaches performed worse than the baseline model. This might be due to the overfitting effect of oversampling the minority class.

While examining the false predictions of our MTL model on the dev set, we noticed that the model was classifying tweets with a negative sentiment as offensive tweets. While it is intuitive for offensive tweets to have a negative sentiment by nature, our model did not capture the fact that not all tweets with negative sentiment are offensive. On another note, the use of words that are offensive in a non-offensive context was found to confuse the model. For example, the words "كلب" and "لئيم" in the following tweets (720, 828), respectively, were not used with an offensive intent and made the model classify both tweets as offensive.

We also found that the model has learned that a tweet cannot be hate-speech unless it is offensive, which would be ideal in case the offensive prediction was perfect. However, in our case, this also made the model falsely predict three tweets as hate-speech after they were falsely predicted as offensive. Furthermore, tweets 785, 881 in the dev set were found to be mislabeled as hate speech, and the model was able to detect this error showing a good understanding of what characterizes hate speech in a tweet. Finally, we found our model to falsely predict tweets that mostly contain mockery, sarcasm, or quoting other offensive/hateful statements.

Future work should explore the use of data augmentation techniques such as adversarial examples and learning from little data approaches such as meta-learning in order to enable state-of-the-art Natural Language Understanding (NLU) models such as AraBERT to be trained efficiently with little data.

5. Conclusion

The presence of hate speech and offensive language on Arabic social platforms is a major issue affecting the social lives of many individuals in the Arab world. The lack of annotated data and the presence of different dialects constitutes major challenges for automated Arabic offensive and hate speech detection systems. In this paper, we proposed the use of pre-trained Arabic BERT for accurate classification of the different tweets. We further augment the AraBERT model using Multitask Learning to enable the model to jointly learn both tasks efficiently with the presence of little labeled data per-task. Our results show the superiority of our proposed Multitask AraBERT model over single-task and Multilabel AraBERT. We explore different methods in order to cope with the presence of imbalanced training classes such as the use a weighted loss function and data re-sampling techniques, but found these methods to not introduce any improvements. Our method achieved the second place on both tasks in the OSACT4 competition.

6. References

Abdelali, A., Darwish, K., Durrani, N., and Mubarak, H. (2016). Farasa: A fast and furious segmenter for arabic. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 11–16.

Al-Hassan, A. and Al-Dossari, H. (2019). Detection of hate speech in social networks: a survey on multilingual corpus. In *6th International Conference on Computer Science and Information Technology*.

Alakrot, A., Murray, L., and Nikolov, N. S. (2018). Towards accurate detection of offensive language in online communication in arabic. *Procedia computer science*, 142:315–320.

Albadi, N., Kurdi, M., and Mishra, S. (2018). Are they our brothers? analysis and detection of religious hate speech in the arabic twittersphere. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 69–76. IEEE.

AUBMind-Lab. (2020). https://github.com/aub-mind/arabert.

Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Pardo, F. M. R., Rosso, P., and Sanguinetti, M. (2019). Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63.

Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1):41–75.

Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., et al. (2018). Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Eleventh international aaai conference on web and social media*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fosler-Lussier, E., Riloff, E., and Bangalore, S. (2012). Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Haddad, H., Mulki, H., and Oueslati, A. (2019). T-hsab: A tunisian hate speech and abusive dataset. In *International Conference on Arabic Language Processing*, pages 251–263. Springer.

Mandl, T., Modha, S., Majumder, P., Patel, D., Dave, M., Mandlia, C., and Patel, A. (2019). Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, pages 14–17.

Mohaouchane, H., Mourhir, A., and Nikolov, N. S. (2019). Detecting offensive language on arabic social media using deep learning. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 466–471. IEEE.

Mubarak, H. and Darwish, K. (2019). Arabic offensive language classification on twitter. In *International Conference on Social Informatics*, pages 269–276. Springer.

Mubarak, H., Darwish, K., Magdy, W., Elsayed, T., and Al-Khalifa, H. (2020). Overview of osact4 arabic offensive language detection shared task. 4.

Mulki, H., Haddad, H., Ali, C. B., and Alshabani, H. (2019). L-hsab: A levantine twitter dataset for hate speech and abusive language. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 111–118.

Qiu, M., Zhao, P., Zhang, K., Huang, J., Shi, X., Wang, X., and Chu, W. (2017). A short-term rainfall prediction model using multi-task convolutional neural networks. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 395–404. IEEE.

Schmidt, A. and Wiegand, M. (2017). A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10.

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

Tynes, B. M., Giang, M. T., Williams, D. R., and Thompson, G. N. (2008). Online racial discrimination and psychological adjustment among adolescents. *Journal of adolescent health*, 43(6):565–569.

Waseem, Z. and Hovy, D. (2016). Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., and Kumar, R. (2019). Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). *arXiv preprint arXiv:1903.08983*.