# Classification of Cyberbullying Text in Arabic

Benaissa Azzeddine Rachid
*RIADI laboratory, NSCS*
*University of Manouba*
La Manouba, Tunisia
benaissaazzedine@hotmail.fr

Harbaoui Azza
*RIADI laboratory, NSCS*
*University of Manouba*
La Manouba, Tunisia
Azza.Harbaoui@gmail.com

Hajjami Henda Ben Ghezala
*RIADI laboratory, NSCS*
*University of Manouba*
La Manouba, Tunisia
hhbg.hhbgg@gmail.com

***Abstract*—The increase in electronic devices and social media use has allowed face-to-face bullying integrate the cyber space. Cyberbullying is an increasing problem that affects its victims worldwide both mentally and physically. Acting upon this phenomenon is of highly importance. Several researches were conducted on cyberbullying classification in English language and less on Arabic. In this paper, we conducted a series of experiments using neural network models (Convolutional and Recurrent Neural Networks) and pre-trained word embeddings in an attempt to classify cyberbullying instances on an Arabic channel news comments dataset. Best models achieved 0.84 F1-score on a balanced version of the aforementioned dataset.**

***Index Terms*—Arabic cyberbullying classification, Machine/Deep learning, Neural networks, Natural language processing, Arabic word embeddings**

## I. Introduction

Social networks have invaded peoples lives and allowed them to create their virtual world where they can share pictures, play video games, texting messages and more other functionalities. Although this virtual playground seems to be personal and safe, it allows the penetration of bad peers; and with the widespread use of electronic devices and time people spend on the internet (internet addiction), users are 24/7 confronted to aggressive persons. These persons have benefited from the advance in technological devices to expand their reach and the extent of their harm [1]. The phenomenon is known as Cyberbullying and is defined as the use of electronic devices repeatedly and intentionally in order to harm a person or a group of persons which cannot defend themselves easily. Online bullying can take place in different forms ranging from sending harmful and aggressive messages to creating fake profiles and sharing of personal and embarrassing pictures of the victims. As opposed to traditional bullying, cyberbullying is more dangerous as the bullying posts can reach a large audience and remains posted for a long time. In addition to its widespread prevalence, cyberbullying has more serious effects than face-to-face bullying. It has been noted that victims who experienced cyberbullying suffers from psychiatric symptoms including depression, anxiety, low self-esteem, suicidal ideation and attempts and Posttraumatic Stress Disorder (PTSD) [2], [3].

These terrible consequences and the dangerous emotional, physical and psychiatric effects that cyberbullying leaves on its victims show the dire need to combat this rising problem. In the past few years, several social media platforms adopted a defensive mechanism against cyberbullying by relying on keyword-based systems that remove aggressive posts and messages if these latter contain words from a defined list of profane and insulting words. However, because of the huge amount of user-generated content and the explicit vocabulary of cyberbullying, it became necessary to move into intelligent and self-learning systems in order to reach a better online content monitoring. Therefore, Machine Learning and Deep Learning models have found their way in cyberbullying detection. Over the last years, cyberbullying detection has been formulated as a text classification problem and especially in English. As cyberbullying is a global issue, the Arab nation is also concerned about it and very few researches have been attended to work on Arabic cyberbullying detection.

In this work, we propose to tackle the problem of Arabic cyberbullying classification using Neural Network models (Recurrent Neural Networks and Convolutional Neural Networks) and word embeddings features on three versions of a dataset of 32K deleted comments from Aljazeera.net.

The remainder of this paper is as follows. Section 2 introduces related works both in Arabic and English cyberbullying classification. Section 3 presents our methodology and experimental setup. Results are reported in Section 4 followed by a conclusion in Section 5.

## II. Related Works

Fueled by the increased power of graphics processing units (GPUs), Neural Networks has invaded several Natural Language Processing tasks making new state-of-the-art results in named entity recognition, machine translation, text classification and sentiment analysis [4].

For many years, cyberbullying classification has been performed using traditional machine learning models with a large set of features. It is only recently that Neural Networks are used for textual online bullying classification. In this section, we introduce several works conducted on cyberbullying (CB) classification in Arabic and English languages.

The first paper that attempts to conduct Arabic cyberbullying classification is the one by [5]. Researchers used scrapping techniques to collect data from Facebook and Twitter, and were able to build a dataset of 91431 English and 35273 Arabic tweets. The dataset was separated into two classes namely Cyberbullying and Non-Cyberbullying and manually annotated by yes and no respectively. Authors used the WEKA toolkit to

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

preprocess text data and used Nave Bayes and Support Vector Machines as classifiers. Both classifiers achieved a high overall F-measure (higher than 0.905) with high F-measure on Non-cyberbullying class and a low one on the other class due to the class imbalance of the dataset (6% bullying content).

In their second work, Haidar and co-workers used neural networks to classify Arabic cyberbullying text. They used the same dataset in their first paper and included more preprocessing steps like removing hyperlinks and non-Arabic characters. The dataset comprises 3015 CB samples and 31875 NCB samples which were then encoded in one-hot embeddings. In the experiments, authors used several version of a Feed Forward Neural Network (FFNN) by tweaking some hyper-parameters namely: the number of hidden layers, number of training epochs and different batch sizes. The best model, which achieved 94.56% validation accuracy, is the 7-layers FFNN trained for two epochs and a batch size equals to 16.
To the best of our knowledge, these two papers are the only one treating the task of Arabic cyberbullying classification in addition to the paper by [6] which is out of the scope of our paper since authors have not used any machine learning or deep learning method. Instead, they used a keyword-based system that uses a weighting mechanism to classify real-time cyberbullying tweets based on a bullying strength. Also the papers [8] and [7] discussed the automatic detection of anti-social behaviour and cyberbullying in Arabic text and its challenges.

In contrast to Arabic, a growing body of work is conducted around English cyberbullying classification. Several researches on the previous task are presented next.
In the paper [9], authors investigated the performance of Recurrent Neural Networks and Convolutional Neural Networks for classifying cyberbullying text motivated by the high results these networks achieved in similar text-based classification tasks. They reimplemented three out of the best deep learning models namely: Convolutional Neural Network by Kim [10], Hybrid Convolutional-Long Short Term Memory (C-LSTM) in [11] and the mixed CNN-LSTM-DNN by Ghosh et Veale [12]. The first model consists of a CNN layer with different filter window sizes (3, 4 and 5) concatenated at a maxpooling layer and followed by a dropout of rate=0.5 and a softmax output. The second model is composed of a CNN layer followed by a RNN layer of type Long Short Term Memory (LSTM) with a dropout rate equal to 0.5. The last model comprises two CNN layers, two LSTM layers followed by a 300-unit Dense layer. The three models were tested on two versions of a social network dataset (Formspring dataset). The first version that represents the original dataset comprises 13160 labeled samples for cyberbullying with 2205 CB posts and 10955 NCB posts. The second version is a balanced version of the dataset with 2205 samples for both classes (CB and NCB classes). Regarding text representation, authors used three pre-trained word embeddings namely: Google-News, Twitter and Formspring word embeddings, with the latter pre-trained on the entire vocabulary of the dataset. The results on the unbalanced version of the dataset showed a slightly outperformance of the CNN by Kim with Google-News embeddings on other models reaching 0.848 F-measure. However, regarding the CB class only, the C-LSTM model with Twitter embeddings reached the highest F-measure 0.444. The latter model also outperformed the others on the balanced version of the dataset with 0.842 F-measure.

The majority of works conducted on online bullying classification, as stated in [13], target only one Social Media Platform (SMP). The authors decided to deal with this last bottleneck as well as two others (addressing only one topic of bullying and use of handcrafted features) using deep learning techniques. Three types of dataset were used namely: a 12k-posts teen oriented Question & Answer forum (Formspring), a 16k-comments from Twitter microblogging platform and a 100k-comments from Wikipedia collaborative knowledge repository. Experiments included Convolutional Neural Networks (CNN), unidirectional and bidirectional Long Short Term Memory (LSTM and BLSTM) and a Bidirectional LSTM with attention. Regarding the inputs to these models, authors used three types of word embeddings tuned to learn task-specific embeddings: random initialized word embeddings, Glove embeddings [14] and Sentiment Specific Word Embeddings (SSWE) [15]. Due to the class imbalance nature of the three datasets, with the cyberbullying class being much larger than the non-cyberbullying class, authors oversampled this latter thrice in all three datasets. The experiments showed that deep learning models surpassed several machine learning benchmarks. The highest F1-measure achieved is 0.93 on the oversampled versions of the three datasets when SSWE embeddings are used in the initialization step. Additionally, authors investigated Transfer Learning of knowledge from one dataset into another one. They deduced that the BLSTM model with attention used with feature level transfer learning (embeddings trained on a dataset were utilized for CB classification in other datasets) achieved a F1-measure higher than 0.93 on all three datasets.

Social media posts are mostly noisy and may contain symbols and misspelled words. [16] made use of these words with incorrect spellings by mapping each word into a vector that represents its phonetic code. The vectors were then fed into a so-called Pronunciation Convolutional Neural Network (PCNN). The model consists of a convolutional layer with different filter sizes (1, 2 and 3) followed by a max-pooling layer and a Softmax layer with dropout. Two unbalanced datasets were used in the experiments. The first one is a Twitter dataset that comprises 1313 tweets with 38% bullying content, and the second one is the Formspring dataset which contains 23243 sentences and 7% bullying content from the total. To handle class imbalance of the datasets, authors used Threshold-Moving (TM), Cost Function Adjusting (CFA) and a hybrid solution that combines the two techniques (TM-CFA). The PCNN model achieved 0.98 F1-socre on the Twitter dataset and 0.562 F1-socre (0.453 Recall) on the Formspring dataset outperforming Linguistic Inquiry and Word Count based machine learning models and two CNN baseline models that uses Google Word2Vec and randomly generated vectors respectively. The class imbalance handling techniques

have proven to be effective and boosted the results on the two datasets. On the Twitter dataset the hybrid technique TM-CFA were more effective than the others, whereas on the Formspring dataset, the cost function adjusting (CFA) improved PCNN performance (0.571 and 0.606 F1-score and Recall respectively). Besides comment-level cyberbullying classification, some other researchers worked on session-level cyberbullying classification. We cite the work of [17] which used a customized convolutional neural network capable of distinguishing between cyberbullying and cyberbagression sessions and the paper by [18] in which authors developed a cyberbullying framework using Hierarchical Attention Network for the detection of cyberbullying samples at the session-level.

## III. Methodology

In this section, we present our approach to Arabic cyberbullying classification. This was experimentally investigated by the use of deep learning models with pre-trained word embeddings as features. In the following, the detailed materials on which our approach relies on are explained.

### A. Datasets

The dataset used in our experiments is the one provided by [19]. The dataset consists of 32K comments that were deleted from the Arabic news channel Aljazeera.net. Due to the channels Community Rules and Guidelines site, any comment which is found to be offensive, racist, sexist, personal attack, inciting violence, non-relevant or advertising, is removed by the channels moderators. The comments were annotated by three CrowdFlower workers as obscene, offensive and clean. The annotation resulted in 533 obscene, 25506 offensive and 5653 clean comments written in Modern Standard Arabic (MSA) and different dialects. This dataset was chosen to work with since the comments contained in it go along with the definition of cyberbullying that is defined in this work as any comment which aims at hurting a person or a group of persons. Three versions of the dataset were used:

- The first version (AJComments-Original) consists of the original dataset which we relabeled into two classes by merging the obscene and offensive classes into one class namely the cyberbullying class (CB - 26039 posts) and the clean class was renamed to non-cyberbullying class (NCB - 5653 posts)
- The second version (AJComments-Balanced) is a balanced version of the previous dataset where both classes (the CB and NCB class) comprises the same number of samples which is equal to 5653 comments. The downsampling of the CB class was conducted by keeping the 5653 samples with the longer length (number of words).
- The third version (AJComments-Unbalanced) is an unbalanced version of the original dataset where the offensive class is dropped and only the obscene and clean class remains as the CB class and NCB class respectively.
  The intuition behind using this last dataset is that cyberbullying naturally happens in a much smaller ratio than 5 to 1 [9], and by conducting experiments on such dataset we would approach a realistic cyberbullying scenario [20].

### B. Preprocessing

The Arabic language is a rich and morphologically complex language that is the native tongue of more than 300 million people worldwide. It is a script language written from left to right and comprises an alphabet of 28 letters. Vowels in Arabic are expressed by Diacritics (harakat) which are symbols placed above or below the letters to add distinct pronunciation and grammatical formulation [21]. These latter are used for tashkil which has a meaning of forming in order to provide information about the correct pronunciation of the words [22].
Our preprocessing steps includes the following:

- Removal of Arabic and English punctuations,
- Removal of html codecs, numbers and symbols,
- Removal of words of size one,
- Removal of diacritics,
- Normalization of Arabic text

### C. Features used

In the experiments, we decided to use word embeddings as input to our deep learning models. Each word in the vocabulary is represented by a 300-dimensional vector. Two pre-trained word embeddings were used for the initialization. The first set of pre-trained word embeddings is the Continuous Bag-Of-Words version of the AraVec embeddings [23] trained on World Wide Web pages.
The second set of pre-trained embeddings is the one provided in [24], in which word vectors were trained on online encyclopedia Wikipedia and the Common Crawl corpus using an extension of the fastText model (Fasttext embeddings).

During the generation of the word vectors, we encountered several out-of-vocabulary words (OOV). These words were given embeddings by training a fastText model on the training dataset. The model represents each word as a bag of character n-grams and assigns each subword n-grams a vector value which are then summed up to build the embedding of the oov word [25].

### D. Models used

Due to the satisfying performance of neural networks on text classification task and previous works conducted on English cyberbullying classification, we investigate the use of several deep learning models for the classification of Arabic cyberbullying text. The models used in the experiments are based on Convolutional Neural Networks, Recurrent Neural Networks (Long Short Term Memory and Gated Recurrent Unit) and combination of both.

Convolutional Neural Network or ConvNets are a type of feed-forward neural networks that learn weights associated with local filters by performing a series of convolution operations and thus capturing spatial and temporal dependencies. RNNs are a type of neural networks with loops in them; they are used to process temporal sequence of data (like text) and

maintains a memory of past information. LSTMs and GRUs are a special kind of Recurrent nets that use gated mechanism to keep and forget information from previous units and thus learning long-term dependencies. Bidirectional LSTM/GRU processes data in both forward and backward direction to benefit from past and future information of current time frame.

For the experiments, we tested various models consisting of several layers. All the models have identical input layer and embedding layer. The latter is followed by either a Convolutional layer, a Bidirectional LSTM or a Bidirectional GRU which are also followed by either a pooling layer (max pooling or average pooling) or an Attention layer. In addition, we tested with hybrid models that consists of a Convolutional layer followed by either a BiLSTM or BiGRU plus a pooling or an attention layer, but also a BiLSTM or a BiGRU on top of a Convolutional layer followed by a pooling or an attention layer.

The tests also included a Kim-CNN like model [10] with 128 filters and different window sizes (2, 3 and 4). Pooling layers or attention layers follow each of the Convolutional layers. Two other models based on the previous one were also tested. The first model consists of a BiLSTM or a BiGRU on top of the Multichannel Convolutional layer, whereas on the second one, the outputs from the pooling layers are concatenated and then passed to either a BiLSTM or a BiGRU.

All these models have an identical dense layer (128 unit) followed by a dropout layer (dropout rate = 0.2) and a sigmoid layer as output. In total, 34 models were tested.

### *E. Experimental Setup*

All the experiments were conducted on a Google Colab environment[1] using Keras API and Scikit-learn library. The three datasets were split into 60/20/20 for train, validation and test data respectively. Each post of the datasets was truncated to the size of post ranked at 95 percentile. The models are trained with a batch size equal to 128 and the earliy stopping hyperparameter tuned on to save the best model only. Additionally, the embeddings were fine-tuned to learn task-specific word embeddings.

For benchmarking purposes, a number of traditional machine learning models were used namely: Multinomial Nave Bayes (MNB), Logistic Regression (LR), Linear SVC, Random Forest (RF), eXtreme Gradient Boosting machines (XGBoost) [26] and Support Vector Machines with RBF kernel (SVM).
Along with these models, varieties of word representation methods were included: Bag-of-words (BoW), Term Frequency Inverse Document Frequency (TF-IDF), N-gram word level TF-IDF and N-gram character level TF-IDF (2 and 3 grams). For each of these models, a 10-fold Cross Validation is performed.

Apart from these experiments, we wanted to test the DL models on a social media dataset in order to see how these models would behave to unseen data of different type that they were trained on. To achieve this goal, a Twitter dataset was used which is made available by the same authors of the AJComments datasest [19]. The Twitter dataset comprises 203 obscene, 444 offensive and 453 clean tweets. Since the experiments were conducted as a binary classification, the dataset was relabeled to CB class by merging the obscene and offensive classes (647 tweets) and the clean class was relabeled to NCB class (453 tweets). In the experiments, the AJComments datasets were used to train the models and the Twitter dataset was split into 80/20 for test and validation data respectively. Results are reported in the next section.

[1]https://colab.research.google.com

## IV. Results & Discussion

The results are presented in terms of accuracy in the balanced version of the dataset, whereas on the two other unbalanced datasets, the results are reported in terms of precision, recall and F1-measure on the minority class only. We chose this particular metrics because of the class imbalance nature of the datasets since on the AJComments-Unbalanced, the number of cyberbullying samples is 10 times bigger than the non-cyberbullying class whereas on the AJComments-Original dataset, non-cyberbullying instances are 50 times larger than cyberbullying instances which negatively impacts the performance of the classifiers.

### *A. Results on AJComments-Original*

Tables I and II report the results of the machine learning and deep learning models respectively. For these latter, and because of the number of models, we only report models result that achieved an overall good F1-score. All other models either have performed very poorly or have achieved a F1-score $\leq$ 0.30.

TABLE I
Machine Learning models' results on AJComments-Original

| | Metrics | | |
|---|---|---|---|
| **Models** | **Precision** | **Recall** | **F1-score** |
| MNB+BoW | 0,68 | 0,08 | 0,15 |
| LR+BoW | 0,56 | 0,21 | 0,30 |
| Linear SVC+char TF-IDF n-gram | 0,48 | 0,38 | 0,42 |
| RF+BoW | 0,50 | 0,16 | 0,24 |
| XGBoost+char TF-IDF n-gram | 0,79 | 0,04 | 0,07 |
| **SVM+ char TF-IDF n-gram** | **0,68** | **0,26** | **0,38** |

The highest F1-score reached is 0,44 by the CNN-LSTM-AVERAGEPOOL model when used with AraVec embeddings, whereas when using Fasttext embeddings, the CNN-BiGRU-ATTENTION model was the best one. We have noticed from the experiments, that using a Bidirectional LSTM/GRU on top of a Multichannel-CNN lead to very poor performance, however, when BiLSTM/BiGRU are used on top of a simple CNN model, they achieve acceptable results.

TABLE II
DEEP LEARNING MODELS' RESULTS ON AJCOMMENTS-ORIGINAL

| | | Metrics | | |
|---|---|---|---|---|
| **Embeddings** | **Models** | **Precision** | **Recall** | **F1** |
| AraVec Embeddings | BLSTM-ATTENTION | 0,55 | 0,29 | 0,38 |
| | BLSTM-AVGPOOL | 0,66 | 0,21 | 0,32 |
| | BLSTM-MAXPOOL | 0,56 | 0,24 | 0,34 |
| | CNN-ATTENTION | 0,61 | 0,30 | 0,41 |
| | CNN -AVGPOOL | 0,58 | 0,27 | 0,37 |
| | multiCNN -MAX-BLSTM | 0,61 | 0,21 | 0,32 |
| | multiCNN -AVG-BGRU | 0,59 | 0,24 | 0,34 |
| | **CNN-BLSTM-AVG** | **0,55** | **0,36** | **0,44** |
| | CNN-BLSTM-MAX | 0,56 | 0,27 | 0,37 |
| | CNN-BGRU-MAX | 0,63 | 0,20 | 0,31 |
| | BLSTM-CNN-ATT | 0,59 | 0,25 | 0,35 |
| | BLSTM-CNN-AVG | 0,62 | 0,21 | 0,32 |
| | BGRU-CNN-ATT | 0,63 | 0,24 | 0,35 |
| | BGRU-CNN-MAX | 0,56 | 0,22 | 0,31 |
| Fasttext Embeddings | CNN-ATTENTION | 0,64 | 0,27 | 0,38 |
| | multiCNN -AVG-BGRU | 0,61 | 0,21 | 0,32 |
| | MultiCNN-MAX | 0,59 | 0,24 | 0,34 |
| | **CNN-BGRU-ATT** | **0,55** | **0,36** | **0,44** |

### B. Results on AJComments-Unbalanced

Table III and IV depicts the performance of ML and DL models respectively. The results on this dataset were much higher than the first one, due to the class imbalance issue explained in the beginning of this section. Since results increased, the decision was to deem only DL models with F1-score higher than 0,50.

TABLE III
MACHINE LEARNING MODELS' RESULTS ON AJCOMMENTS-UNBALANCED

| | Metrics | | |
|---|---|---|---|
| **Models** | ***Precision*** | ***Recall*** | ***F1-score*** |
| MNB+BoW | 0,87 | 0,35 | 0,50 |
| LR+BoW | 0,91 | 0,37 | 0,53 |
| **Linear SVC+ char TF-IDF n-gram** | **0,88** | **0,59** | **0,71** |
| RF+ TF-IDF | 0,80 | 0,41 | 0,56 |
| XGBoost+char TF-IDF n-gram | 0,93 | 0,37 | 0,53 |
| SVM+char TF-IDF n-gram | 0,98 | 0,37 | 0,54 |

The Linear SVC model combined with character level TF-IDF N-grams was the best performing model. It achieves a F1-score of 0,71 outperforming all other models. DL models achieved slightly lower results, with the Multichannel CNN Kim's like model followed by attention layers reaching 0,67 F1-score when used with AraVec embeddings, whereas a simple one-layered CNN model with attention reached 0,66 F1-score when combined with Fasttext embeddings. Once again, using complex hybrid models did not give any good results.

TABLE IV
DEEP LEARNING MODELS' RESULTS ON AJCOMMENTS-UNBALANCED

| | | Metrics | | |
|---|---|---|---|---|
| **Embeddings** | ***Models*** | ***Precision*** | ***Recall*** | ***F1-score*** |
| **AraVec Embeddings** | BLSTM-MAXPOOL | 0,67 | 0,41 | 0,51 |
| | BGRU-ATTENTION | 0,72 | 0,40 | 0,51 |
| | BGRU -MAXPOOL | 0,68 | 0,47 | 0,56 |
| | CNN-ATTENTION | 0,72 | 0,42 | 0,53 |
| | CNN -MAXPOOL | 0,62 | 0,43 | 0,51 |
| | **MultiCNN-ATT** | **0,83** | **0,56** | **0,67** |
| | MultiCNN-AVG | 0,88 | 0,42 | 0,57 |
| | MultiCNN-MAX | 0,65 | 0,45 | 0,53 |
| | BGRU-CNN-MAX | 0,77 | 0,39 | 0,52 |
| **Fasttext Embeddings** | BGRU -MAXPOOL | 0,69 | 0,60 | 0,64 |
| | **CNN-ATTENTION** | **0,89** | **0,52** | **0,66** |
| | CNN -AVGPOOL | 0,93 | 0,45 | 0,61 |
| | CNN -MAXPOOL | 0,81 | 0,46 | 0,59 |
| | MultiCNN-AVG | 0,87 | 0,42 | 0,56 |
| | MultiCNN-MAX | 0,77 | 0,42 | 0,54 |
| | CNN-BGRU-ATT | 0,85 | 0,46 | 0,60 |

### C. Results on AJComments-Balanced

Results of the machine learning and deep learning models on the balanced version of the original dataset are presented respectively in Table V and Table VI. Results are reported in terms of accuracy. All models performed good on this dataset with the lowest performing model reaching an accuracy equal to 0,71. We chose to report models accuracy that are higher or equal to 0,82.

As it can be noticed from the tables, Random Forest, XGBoost and SVM combined with character level TF-IDF N-grams reached the highest accuracy (85%). A slightly lower performance was achieved by CNN-AVERAGEPOOL model whien used with AraVec and CNN-BiLSTM-MAXPOOL model when used with Fasttext embeddings. For the first time in these experiments, the hybrid complex models gave good results with the BiLSTM-Multichannel-CNN with attention performing aqually with the previous best models.

### D. Results on Twitter dataset

As explained in the previous section, in the second part of the experiments, all models were trained on one of the AJComments datasets and tested on the Twitter dataset. The results reported below concern the models' performance when trained on the balanced version of the AJComments dataset

TABLE V
MACHINE LEARNING MODELS' RESULTS ON AJCOMMENTS-BALANCED

| **Models** | **Accuracy** |
|---|---|
| MNB+char TF-IDF n-gram | 0,70 |
| LR+char TF-IDF n-gram | 0,84 |
| Linear SVC+BoW | 0,81 |
| **RF+char TF-IDF n-gram** | **0,85** |
| **XGBoost+char TF-IDF n-gram** | **0,85** |
| **SVM+char TF-IDF n-gram** | **0,85** |

TABLE VI
DEEP LEARNING MODELS' RESULTS ON AJCOMMENTS-BALANCED

| **Embeddings** | **Models** | **Accuracy** |
|---|---|---|
| AraVec Embeddings | BLSTM-ATTENTION | 0,82 |
| | BLSTM-AVGPOOL | 0,82 |
| | BGRU -AVGPOOL | 0,83 |
| | **CNN -AVGPOOL** | **0,84** |
| | BLSTM- multiCNN -AVG | 0,82 |
| | MultiCNN-MAX | 0,83 |
| | CNN-BLSTM-ATT | 0,82 |
| | CNN-BLSTM-AVG | 0,83 |
| | CNN-BLSTM-MAX | 0,82 |
| | CNN-BGRU-MAX | 0,82 |
| | BGRU-CNN-MAX | 0,82 |
| Fasttext Embeddings | BGRU -MAXPOOL | 0,83 |
| | **BLSTM-multiCNN-ATT** | **0,84** |
| | BGRU- multiCNN -ATT | 0,82 |
| | **CNN-BLSTM-MAX** | **0,84** |
| | CNN-BGRU-AVG | 0,82 |
| | CNN-BGRU-MAX | 0,83 |
| | BLSTM-CNN-MAX | 0,83 |
| | BGRU- multiCNN -ATT | 0,82 |

(AJComments-Balanced). The models resulted in low F1-score on the NCB class when trained on the AJComments-Original dataset, and low F1-score on the CB class when trained on the AJComments-Unbalanced dataset. Table VII and VIII highlight the best performing models in terms of Precision, Recall and F1-score in both Cyberbullying and Non-cyberbullying class respectively.

This is a selection of the best performing models. As it can be noticed, all models perform well in classifying non-cyberbullying tweets, whereas they struggle in classifying cyberbullying tweets with the exception of the CNN-MAXPOOLING model (Table VIII). Even if the number of CB tweets is slightly bigger (61% bullying content) than the NCB tweets (59% clean content), all models (except the CNN-MAXPOOL model) misclassified almost half of the bullying

TABLE VII
DEEP LEARNING MODELS' RESULTS ON TWITTER DATASET - NCB CLASS

| | | **Metrics** | | |
|---|---|---|---|---|
| **Embeddings** | **Models** | **Precision** | **Recall** | **F1-score** |
| AraVec Embeddings | BGRU-AVGPOOL | 0,52 | 0,85 | 0,65 |
| | CNN-MAXPOOL | 0,51 | 0,43 | 0,47 |
| | MultiCNN-AVGPOOL-BLSTM | 0,51 | 0,82 | 0,63 |
| | MultiCNN-AVGPOOL-BGRU | 0,50 | 0,73 | 0,60 |
| | MultiCNN-ATTENTION | 0,50 | 0,77 | 0,61 |
| | CNN-BLSTM-MAXPOOL | 0,52 | 0,77 | 0,62 |
| | CNN-BGRU-ATTENTION | 0,53 | 0,86 | 0,65 |
| Fasttext Embeddings | BGRU-ATTENTION | 0,51 | 0,80 | 0,63 |
| | CNN-MAXPOOL | 0,51 | 0,74 | 0,60 |
| | MultiCNN-AVGPOOL-BGRU | 0,50 | 0,79 | 0,61 |
| | CNN-BGRU-ATTENTION | 0,51 | 0,79 | 0,62 |

TABLE VIII
DEEP LEARNING MODELS' RESULTS ON TWITTER DATASET - CB CLASS

| | | **Metrics** | | |
|---|---|---|---|---|
| **Embeddings** | **Models** | **Precision** | **Recall** | **F1-score** |
| AraVec Embeddings | BGRU-AVGPOOL | 0,81 | 0,45 | 0,58 |
| | CNN-MAXPOOL | 0,64 | 0,71 | 0,67 |
| | MultiCNN-AVGPOOL-BLSTM | 0,78 | 0,45 | 0,57 |
| | MultiCNN-AVGPOOL-BGRU | 0,72 | 0,49 | 0,59 |
| | MultiCNN-ATTENTION | 0,74 | 0,46 | 0,57 |
| | CNN-BLSTM-MAXPOOL | 0,76 | 0,51 | 0,61 |
| | CNN-BGRU-ATTENTION | 0,82 | 0,46 | 0,59 |
| Fasttext Embeddings | BGRU-ATTENTION | 0,77 | 0,47 | 0,59 |
| | CNN-MAXPOOL | 0,73 | 0,51 | 0,60 |
| | MultiCNN-AVGPOOL-BGRU | 0,75 | 0,46 | 0,57 |
| | CNN-BGRU-ATTENTION | 0,77 | 0,47 | 0,59 |

tweets (higher Recall on the NCB class than on the CB class). These results show that the models were incapable of capturing bullying content in social media when trained on a data collected from a channel news site commentary.

By the end of the experiments, we summarized our findings in the following points:

- Using complex hybrid DL models is not significant unless used on a balanced dataset
- Simple and combined CNN-RNN models achieve good performance
- Machine learning models are very competitive and in some cases outperforming DL models when it comes to classify Arabic cyberbullying instances
- Linear SVC+character level TF-IDF N-grams trained on an unbalanced dataset are a good candidate for a real-world cyberbullying scenario
- Deep learning models trained on bullying news site comments are not adequate for classifying instances from a different type such as social networks data

## V. Conclusion & Future works

This paper has given an account of testing numerous deep learning models for the task of classifying cyberbullying instances on both balanced and unbalanced versions of an Arabic news channel deleted comments dataset. The results of this study indicates the effectiveness of using simple and combined Convolutional and Recurrent Neural Networks (CNN/LSTM/GRU) coupled with Arabic pre-trained word embeddings (AraVec and Fasttext) achieving 84% F1-score on a balanced dataset. Machine learning models were also included in the experiments and showed competitive performance to DL models and even outperfoming the latter in few experiments. The 34 deep learning models were also tested on a Twitter dataset which is from a different type that they were trained on and achieved a F1-score $\approx 0.62$ as a best result.

For future works, we aim to enhance the models' classification results by using a different type of embeddings (contextualized word embedding) and testing transfer learning techniques for Arabic data.

## References


[1] Hinduja, S. & Patchin, J. W. (2019). Cyberbullying Identification, Prevention, and Response. Cyberbullying Research Center (cyberbullying.org).

[2] Field T. Cyberbullying: A narrative review. J Addict Ther Res. 2018; 2: 010-027. https://dx.doi.org/10.29328/journal.jatr.1001007

[3] Ranney ML, Patena JV, Nugent N, Spirito A, Boyer E, et al. PTSD, cyberbullying and peer violence: Prevalence and correlates among adolescent emergency department patients. Gen Hosp Psychiatry. 2016; 39: 32-38. Ref.:Ref.: https://tinyurl.com/y82y9ogf

[4] Rachid, B.; Azza, H. and Henda, B. (2018). Sentiment Analysis Approaches based on Granularity Levels.In Proceedings of the 14th International Conference on Web Information Systems and Technologies - Volume 1: WEBIST, ISBN 978-989-758-324-7, pages 324-331. DOI: 10.5220/0007187603240331

[5] Haidar, Batoul & Maroun, Chamoun & Serhrouchni, Ahmed. (2017). A Multilingual System for Cyberbullying Detection: Arabic Content Detection using Machine Learning. Advances in Science, Technology and Engineering Systems Journal. 2. 275-284. 10.25046/aj020634.

[6] D. Mouheb, M. H. Abushamleh, M. H. Abushamleh, Z. A. Aghbari and I. Kamel, "Real-Time Detection of Cyberbullying in Arabic Twitter Streams," 2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS), CANARY ISLANDS, Spain, 2019, pp. 1-5. doi: 10.1109/NTMS.2019.8763808

[7] A. H. Alduailej and M. B. Khan, "The challenge of cyberbullying and its automatic detection in Arabic text," 2017 International Conference on Computer and Applications (ICCA), Doha, 2017, pp. 389-394. doi: 10.1109/COMAPP.2017.8079791

[8] Al-Hassan, Areej & Al-Dossari, Hmood. (2019). DETECTION OF HATE SPEECH IN SOCIAL NETWORKS: A SURVEY ON MULTILINGUAL CORPUS. 83-100. 10.5121/csit.2019.90208.

[9] Rosa, Hugo & Martins de Matos, David & Ribeiro, Ricardo & Coheur, Luisa & Carvalho, Joao. (2018). A Deeper Look at Detecting Cyberbullying in Social Networks. 1-8. 10.1109/IJCNN.2018.8489211.

[10] Y. Kim, Convolutional Neural Networks for Sentence Classification, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), pp. 17461751, 2014.

[11] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, A C-LSTM Neural Network for Text Classification, CoRR, abs/1511.08630, nov 2015.

[12] A. Ghosh and T. Veale, Fracking Sarcasm using Neural Network, in 7th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis, pp. 161169, 2016.

[13] Agrawal, Sweta & Awekar, Amit. (2018). Deep Learning for Detecting Cyberbullying Across Multiple Social Media Platforms. 10.1007/978-3-319-76941-7_11.

[14] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In EMNLP, pages 1532{1543, 2014.

[15] D. Tang, F.Wei, N. Yang, M. Zhou, T. Liu, and B. Qin. Learning sentiment-speci
c word embedding for twitter sentiment classi
cation. In ACL, pages 1555{1565, 2014.

[16] X. Zhang et al., "Cyberbullying Detection with a Pronunciation Based Convolutional Neural Network," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, 2016, pp. 740-745. doi: 10.1109/ICMLA.2016.0132

[17] Zhong, Haoti & Miller, David & Squicciarini, Anna. (2019). Flexible Inference for Cyberbully Incident Detection: European Conference, ECML PKDD 2018, Dublin, Ireland, September 1014, 2018, Proceedings, Part III 10.1007/978-3-030-10997-4_22.

[18] Cheng, Lu & Guo, Ruocheng & Silva, Yasin & Hall, Deborah & Liu, Huan. (2019). Hierarchical Attention Networks for Cyberbullying Detection on the Instagram Social Network.

[19] Mubarak, Hamdy & Darwish, Kareem & Magdy, Walid. (2017). Abusive Language Detection on Arabic Social Media. 52-56. 10.18653/v1/W17-3008.

[20] Rosa, Hugo & Carvalho, Joao & Calado, Pavel & Martins, Bruno & Ribeiro, Ricardo & Coheur, Luisa. (2018). Using Fuzzy Fingerprints for Cyberbullying Detection in Social Networks. 1-7. 10.1109/FUZZ-IEEE.2018.8491557.

[21] Ayedh, Abdullah & TAN, Guanzheng & Alwesabi, Khaled & Rajeh, Hamdi. (2016). The Effect of Preprocessing on Arabic Document Categorization. Algorithms. 9. 27. 10.3390/a9020027.

[22] Alhanjouri, Mohammed A. (2017). Pre Processing Techniques for Arabic Documents Clustering. International Journal of Engineering and Management Research (IJEMR), Volume: 7, Number: 2, Vandana Publications. http://hdl.handle.net/20.500.12358/24635

[23] Soliman, A.B., Eissa, K., & El-Beltagy, S.R. (2017). AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP. ACLING.

[24] Grave, E., Bojanowski, P., Gupta, P., Joulin, A., & Mikolov, T. (2018). Learning Word Vectors for 157 Languages. ArXiv, abs/1802.06893.

[25] Bojanowski, Piotr & Grave, Edouard & Joulin, Armand & Mikolov, Tomas. (2016). Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics. 5. 10.1162/tacl_a_00051.

[26] Chen, Tianqi & Guestrin, Carlos. (2016). XGBoost: A Scalable Tree Boosting System. 785-794. 10.1145/2939672.2939785.