Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 207 (2022) 769–778

Procedia Computer Science

www.elsevier.com/locate/procedia

26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)

Combining FastText and Glove Word Embedding for Offensive and Hate speech Text Detection

Nabil Badri[a,*], Ferihane Kboubi[a,*], Anja Habacha Chaibi[a,*]

[a]*RIADI Laboratory, ENSI School, University of Manouba, 2010 La Manouba, Tunisia.*

Abstract

Over the past decade, increased use of social media has led to an increase in hate content. To address this issue new solutions must be implemented to filter out this kind of inappropriate content. Because manual filtering is difficult, several studies have been conducted in order to automate the process.

This paper introduces a method based on a combination of Glove and FastText word embedding as input features and a BiGRU model to identify hate speech from social media websites.

The obtained results show that our proposed model **(BiGRU_Glove_FT)** is effective in detecting inappropriate content. This model detect hate speech on OLID dataset, using an effective learning process that classifies the text into offensive and not offensive language. The performance of the system attained 84%, 87%, 93%, 90% accuracy, precision, recall, and f1-score respectively.

Keywords: Inappropriate content classification; Machine/Deep learning; Neural networks; NLP; Glove/FastText word embeddings

1. Introduction

With the widespread of internet, huge increase in smartphone usage rate in recent years, the freedom of expression privilege granted after the Tunisian revolution, the mask of anonymity that the internet provides, and despite the fact that Twitter's terms of use forbid such inappropriate content, the spread of derogatory and hate speech has increased. It became easy to spread inappropriate content on social media, such as Twitter and Facebook, against individuals or groups. Furthermore, toxic language can take various forms, such as cyberbullying, which was one of the major reasons behind suicide [20]. The overlap between subtasks is illustrated by the variety of labels used in prior work [51]. For example, in annotating for cyberbullying events, [49] identifies discriminative remarks (racist, sexist) as a subset of "insults", and focuses on classifying different subtypes of cyberbullying using an SVM [48], whereas [32] classifies similar remarks as "hate speech" or "derogatory language". According to United Nations strategy and plan

* Corresponding author. Tel.: +216 71 600 444 ; fax: +216 71 600 449.
E-mail address: nabil.badri@ensi-uma.tn, ferihane.kboubi@fsegt.utm.tn, anja.habacha@ensi-uma.tn

1877-0509
10.1016/j.procs.2022.09.132

of action on hate speech[1], has no international legal definition, but it is hinged on incitement, which is an explicit and deliberate act aimed at discrimination, hostility and violence. Similarly, offensive speech has been defined as the text, which uses abusive slurs or derogatory terms [18], which in many contexts have been confused with hate speech. The two terms "offensive language" and "hate speech" identification are sub-fields of natural language processing (NLP). The challenge is that online inappropriate content are predominantly expressed in natural language text, and therefore, efficient extraction tools and analysis of inappropriate content from unstructured text is necessary. Such tools, often adopt algorithms from (NLP), information retrieval, machine Learning and deep learning disciplines. In the context of protecting children from inappropriate content in mobile Apps, some existing research [21], provide mechanisms for parents to rate the maturity levels of smartphone apps, enabling them to choose maturity-appropriate apps for their children. Unfortunately, the manual way of filtering out toxic content is unscalable, and detecting or removing such content manually from the web is a tedious task. This motivates researchers to identify automated ways, that is able to detect such hateful content on the social media.

In this paper, we perform extensive experiments to tackle the problem of inappropriate content classification using machine learning and neural network models with hyper-parameter optimization and word embeddings features on two different English datasets. We will introduce our proposed models **(BiGRU_Glove_FT)** which consists of a combination of FastText (FT) and Glove word embedding as input features with BiGRU model.

The rest of the paper is structured as follows. In section 2, we present the related literature for offensive and hate speech classification. In section 3, we present our proposed methodology. In section 4, we introduce the experimentations and results. In section 5, we take the discussion of implementation challenges notice. In section 6, we finish with a conclusion and future work.

2. Related Work

The methodology and technologies of protection against inappropriate, dubious and harmful information are under close state attention all over the world [5, 9]. Similarly, an annotation schema for socially unacceptable discourse practices was proposed by Fiˇser et al. [14]. Although social media sites such as Twitter generally prohibit hate speech and other forms of abuse against an individual or a group of people[2], indeed, [17] have conducted a study of different datasets regarding hate-speech identification in Spanish, such speech has thrived for several reasons. First, the relative anonymity of the internet has emboldened perpetrators, who might otherwise fear the consequences of such harassment [7]. Second, social media sites primarily rely upon manual screening and reporting of abusive texts, which is unscalable to the amount of data [57]. For decades, machine learning (ML) approaches targeting NLP problems have been based on shallow models (e.g., SVM and logistic regression) trained on very high dimensional and sparse features. The majority of methods focus on extracting features from text. Some research used lexical features including dictionaries [3] and bag-of-words [34]. These features were found to be enable to comprehend the meaning of the sentences. [26] present a hybrid model to detect aggressive posts containing images and text on social media, while [25] used several classical machine learning methods, as well as different deep learning models on the task of automatic detection of hateful and offensive speech through the example of the HASOC 2019 challenge.

Despite the problems encountered by machine learning (ML), we find several ML based solutions for the automatic detection have been proposed and have shown good results, such as the Naive Bayes (NB) [34], Logistic Regression (LR) [33] and Support Vector Machines (SVM) [33] classifier.

In recent years, deep learning-based approaches have become increasingly popular for this task, a variety of model designs and methods have blossomed in the context of natural language processing (NLP). Deep learning methods employ multiple processing layers to learn hierarchical representations of data, and have produced state-of-the-art results in many domains [54]. Many algorithms for modeling terms have been suggested [28, 44]. In addition, FastText, GloVe and Word2Vec were used as word embeddings, which shows comparatively better results [42]. They achieved their best results by using GloVe + CNN + LSTM + BestAug, where BestAug is combination of PosAug and ThreshAug. In another research, [31] proposed ETHOS dataset to develop AI based hate speech detection framework that have used FastText, Glove and FastText + Glove as word embeddings with CNNs, BiLSTMs and LSTMs. In [40],

[1] https://www.un.org/en/genocideprevention/documents/
[2] https://help.twitter.com/en/rules-and-policies/twitter-rules

deep neural network (DNN) with static BERT embeddings outperforms the same deep neural network which is using word embedding as FastText, GloVe or FastText + GloVe in all metrics. [32] perform hate speech detection on Yahoo Finance and News data, using supervised learning classification, while [57] solved the problem of detecting 'long-tail' hate speech in twitter datasets that lack unique and discriminative features using DNN.

The predictive modeling techniques Continus Bag-Of-Words (CBOW), Skip-Gram [23], and FastText [13] reduce the loss of predicting the target words from the context words provided the vector representations. FastText, on the other hand, considers each phrase to be made up of character n-grams. ELMo, which is state-of-the-art modeling technique assigns each term a representation that is based on the whole corpus of sentences to which it belongs [36].

[4] used four pre-trained models in hate speech task, which were trained on large datasets for solving problem statements similar to the task at hand, and compared the results generated by them with results generated by a CNN. They experimentally showed best result with their ensemble models and sometimes even outperforms existing methods. These models are: BERT [12], RoBERTA [27], XLNet [52], and DistilBERT [45]. [11] demonstrated that a simple deep learning framework outperforms most state-of-the-art approaches in several NLP tasks such as named-entity recognition (NER), semantic role labeling (SRL), and POS tagging. [10] were the first work to show the utility of pre-trained word embeddings. They proposed a neural network architecture that forms the foundation to many current approaches. The work also establishes word embeddings as a useful tool for NLP tasks. However, the immense popularization of word embeddings was arguably due to [30] who proposed the continuous bag-of-words (CBOW) and skip-gram models to efficiently construct high-quality distributed vector representations. Since then, numerous complex deep learning based algorithms have been proposed to solve difficult NLP tasks. [16], used a deep learning models partially trained on the 6,909 english twitter hatespeech dataset created by (Waseem 2016)[3], and annotated by CrowdFlower users[4], to address hate speech. They implemented a convolutional neural network (CNN) with random word vectors, word2vec word vectors and character n-gram, were considered as feature embeddings when training the CNN model, in order to classify social media text as one of four categories: "Racism", "Sexism", "Both (racism & sexism)", and "Non-hate-speech". [1], provided a technique for categorizing annotators into groups based on their annotation behavior, with the assumption that such categorization represents variables such as cultural background, common social behavior, and other aspects. [37], worked on detecting offensive language in tweets using LSTM, while Zhang and Luo implemented a convolutional neural network (CNN) and a gated recurrent unit (GRU), a kind of recurrent neural network (RNN), to classify social media text as one of four categories: 'non-hate', 'sexism', 'racism', or 'both' [57]. Considering that GRUs have fewer parameters and thus may train faster, so some researches [56], choose GRU as their recurrent neurons. [53], shows that [46] works, uses a series of convolution and maxpooling layers to create a Convolution Latent Semantic Model (CLSM) aimed at learning low-dimensional semantic representations of search queries and web documents. CLSM uses character trigram and word n-gram features and is trained using click-through data. Results on a real-world dataset showed better performance over state of the art methods such as DSSM [22]. Researchers have tried using CNN and LSTM architectures [43] in the context of other text mining problems such as Named Entity Recognition (NER) and Sentiment Analysis [38], [50], and [2].

The review on the related work done in this field shows importants notes such as, –(a) the models trained after extracting N-gram features from text give better results [32], and [16], (b) the TFIDF approach on the bag-of-words features also show promising results [8], (c) the advantages of Glove [39] and FastText [6] for word representation, (d) GRUs is stretchy, easy to handle and have fewer parameters and thus may train faster [53], and (e) the good results that the roBERTa algorithm provided in resolving such NLP problems [27].

3. Proposed methodology

Based on the review of features and the prominent classifiers used for text classification in the past work, we present, in this section, our approach of inappropriate English textual content detection. Our approach is based on a deep learning classification model (BIGRU) and a combination of word embedding techniques (Glove and FastText) as features.

Our proposed BiGRU-Glove-FT model takes an input text and outputs the probability of this input text belonging to the inappropriate class (Offensive or Not Offensive). The input text is fed into the model in the form of a combination

[3] http://github.com/zeerakw/hatespeech
[4] https://www.crowdflower.com/

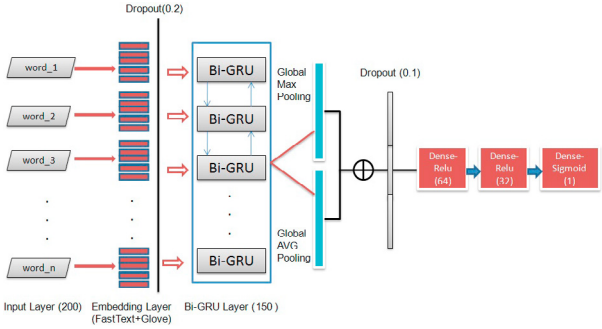

Fig. 1. The Bi-GRU model with combination of FastText and Glove embedding.

of "Glove and FastText" word embedding matrix. BiGRU-Glove-FT model consists of eight sequential layers—(a) Input Layer, (b) Embedding Layer, (c) SpacialDropout layer, (d) Bidirectional(GRU) layer, (e) 1D Globalaverage pooling and 1D Global max pooling layer, (f) Concatenation layer, (g) Dropout layer, and finally (h) Dense layer as shown in Figure 1.

3.1. Preprocessing

To improve the performance of our method, it is essential to perform some preprocessing steps to clean out the text. The following is a list of the pre-processing operations performed.

First, all the numbers, punctuation marks, urls (http:// or www.) and symbols (emoji, #tags, mention) were removed from the tweet as they do not contain information related to sentiment. After that, tokenization and lowercasing was applied to the tweets. Tokenization was done using tokenizer from NLTK package. Finally, stop words are removed. The list is obtained from NLTK package.

3.2. FastText and Glove word embedding

We used a combination of two word embeddings techniques as input to the deep learning classification model. We chose a 300-dimensional vector to represent each word in the vocabulary. The initialization was done using two pre-trained word embeddings. The word embedding techniques used for this work are:

• FastText: FT [24] algorithm created by Facebook assumes every word to be n-grams of character. It helps to give the vector representations for out of vocabulary words. For the current work, FastText embeddings[5] is used for generating token vectors of dimension 300. Each vector corresponding to the tweet is generated by taking the average of token vectors.

• Glove: Glove [35] learns word embeddings by dimensionality reduction of the co-occurrence count matrix. Instead of learning raw co-occurrence probabilities, it learns ratios of co-occurrence probabilities to distinguish relevant words from irrelevant words. In our work, we used GloVe embeddings[6] trained on a large Common Crawl (840B tokens, 2.2M vocab, cased, 300d vectors, 2.03 GB download).

3.3. Bidirectional GRU (BiGRU)

The GRU architecture only preserves past information. To understand word context better, it is also important to look at the future information given in the sequence, this can be done by using bidirectional GRU. We implemented our model based on neural networks such as BiGRU to solve the problem of Hate Speech detection on social media. Indeed, BiGRU model is a sequence processing model that consists of two GRUs. One taking the input in a forward direction, and the other in a backwards direction. It is a bidirectional recurrent neural network with only the input and forget gates. BiGRU is able to save important information from the texts. In addition, it is also able to look at word from past and future state that enables the model to understand word context better than machine learning models.

[5] https://fasttext.cc/docs/en/english-vectors.html
[6] https://nlp.stanford.edu/projects/glove/

4. Experimentations and results

In this section, firstly we begin by presenting the dataset we used for the evaluation, then we describe the experimental setting, and finally we present the achieved results. For evaluation of our proposed methods and comparison with our models we have used Accuracy (Acc), F1-score (F1), Recall (R), Precision (P), Weighted Average and F1-macro.

4.1. Datasets

In this subsection, we present the two datasets used in the experimentations, then we describe how they fit the scope of our analysis. We experimented with two datasets (OLID [7] and Hatespeech[8]). The first dataset contains 16K annotated tweets (Zampieri et al. 2019b), 4640 tweets are labeled as Offensive, and 9460 as Not Offensive. The second dataset contains 12406K annotated tweets (Founta et al. 2018), 53851 tweets are labeled as Normal, 27150 as Abusive, 14030 as Spam, and 4965 as Hateful. These two datasets had already been split into Train set and Test set. Thus, the results are comparable across different models and settings. The class distributions of the datasets are shown in Table 1 and 2.

4.2. Experimental Setup

For the experiementations, we use Keras6 with Tensorflow7 back-end. We run the experiments on a Google Colab environment with Graphic Processing Units (GPUs). In terms of training, a maximum of 30 epochs is allowed and we employ a separate validation set to perform early stopping: training is interrupted if the validation F1-score did not drop in 8 consecutive epochs and the weights of the best epoch are restored. All the datasets were split in 60/20/20 training, validation and test respectively. It is important to notice that the same model (with the same architecture, number of layer, number of unit and parameters) is used for all datasets. Overall, the model, including the pre-trained word embeddings, contains approximately 407,401 trainable parameters (i.e., weights).

For benchmarking purposes, a number of traditional machine learning models were used namely: Multinomial Nave Bayes (MNB), Logistic Regression (LR), Random Forest (RF), eXtreme Gradient Boosting machines (XGB) [41], and Support Vector Machines with RBF kernel (SVM). Along with these models, varieties of word representation methods were included: Count Vectors as features, Term Frequency Inverse Document Frequency (TF-IDF), N-gram word level TF-IDF and N-gram character level TF-IDF. And based on the satisfactory performance of neural networks on text classification tasks and previous work on English inappropriate content classification, we study the use of RoBERTa model [27] and [19] for the text classification of inappropriate content. Results are reported in the next subsection.

4.3. Results

We train each model on training dataset. The performance of each algorithm is analyzed based on the standard classification evaluation metrics which are: Accuracy (Acc), F1-score (F1), Recall (R), Precision (P), Weighted Average and F1-macro. The performance of these five algorithms are compared. To get better results several hyperparameters were used and incorporated. This report shows all the results along with the best performing models among all. We ran a variety of preliminary evaluations before submitting the final results. We optimized the hyper-parameters to maximize F1, R, P, Weighted Average and F1-macro.

We optimized the performance of all models by tuning their parameters using the validation set. In the case of SVM, we tuned the parameter C and also tried various kernels. The best performance was found with C = 10 and rbf kernel. In table 3, the best result was achieved by Naive Bayes and SVM models, one with Count Vectors and the second with Word Level TF-IDF as features, which is an interesting indicator that, although NB and SVM are considered to be among the simplest traditional machine learning models, it can yield satisfactory results on this task. Results on the OLID dataset were slightly different, and this time, these two machine learning models achieved the best results also.

The results for all five machine learning models are shown in the table 3 below, and compared to the BiLSTM and SVM models used by Marcos Zampier, 2019. In the first NB model, Count Vectors were considered as feature

[7] https://sites.google.com/site/offensevalsharedtask/olid
[8] https://github.com/ENCASEH2020/hatespeech-twitter

Table 1. OLID Dataset.

Classes	Train	Test	Total tweets
Offensive	4400	240	4640
Not Offensive	8840	620	9460
	13240	860	14100

Table 2. Hate speech Dataset.

Classes	Train	Test	Total tweets
Normal	11213	2817	53851
Abusive	21781	5369	27150
Spam	43003	10848	14030
Hateful	3999	966	4965
	79996	20000	99996

Table 3. Final results of the classical baselines machine learning models, for the OLID dataset. The **bold** figures represent the best scores and underline represents the second best.

OLID Dataset (2 classes)											
***		Not			Off			Weighted Average			***
Models	Acc	P	R	F1	P	R	F1	P	R	F1	F1-Macro
Models on Count Vectors											
Naive bayes (NB)	0.80	**0.84**	0.89	0.86	0.66	**0.57**	**0.61**	0.79	0.80	0.79	**0.74**
Logistic Regression (LR)	**0.81**	0.83	0.93	0.87	0.72	0.50	0.59	**0.80**	**0.81**	0.79	0.73
Support Vector Machine (SVM)	0.78	0.77	**0.98**	0.86	0.81	0.27	0.40	0.78	0.78	0.73	0.63
XGBoost (XGB)	0.79	0.78	**0.98**	0.87	**0.85**	0.29	0.43	**0.80**	0.79	0.75	0.65
Random Forest (RF)	**0.81**	0.82	0.94	**0.88**	0.75	0.48	0.59	**0.80**	**0.81**	**0.80**	0.73
Models on Word Level TF IDF Vectors											
Naive Bayes (NB)	0.78	0.77	**0.99**	0.86	**0.89**	0.23	0.36	0.80	0.78	0.72	0.61
Logistic Regression (LR)	**0.82**	**0.82**	0.96	**0.88**	0.82	0.44	0.57	**0.82**	**0.82**	**0.80**	**0.73**
Support Vector Machine (SVM)	0.81	**0.82**	0.94	**0.88**	0.75	**0.48**	**0.59**	0.80	0.81	**0.80**	**0.73**
XGBoost (XGB)	0.79	0.78	0.98	0.87	0.87	0.28	0.42	0.80	0.79	0.74	0.65
Random Forest (RF)	0.81	**0.82**	0.94	**0.88**	0.75	**0.48**	**0.59**	0.80	0.81	**0.80**	**0.73**
Models on Ngram Level TF IDF Vectors											
Naive bayes (NB)	**0.75**	0.75	0.96	**0.85**	0.66	0.18	0.29	**0.73**	**0.75**	0.69	0.57
Logistic Regression (LR)	0.74	**0.76**	0.95	0.84	0.63	0.20	0.31	0.72	0.74	0.69	0.58
Support Vector Machine (SVM)	0.74	**0.76**	0.87	0.81	0.47	**0.30**	**0.37**	0.68	0.71	0.69	**0.59**
XGBoost (XGB)	0.73	0.73	**0.99**	0.84	**0.70**	0.06	0.11	0.72	0.73	0.64	0.47
Random Forest (RF)	0.72	**0.76**	0.93	0.84	0.58	0.25	0.35	0.71	0.74	**0.70**	**0.59**
Models on Character Level TF IDF Vectors											
Naive bayes (NB)	0.77	0.76	**0.98**	0.86	0.81	0.22	0.34	0.78	0.77	0.71	0.60
Logistic Regression (LR)	**0.81**	0.81	0.95	**0.88**	0.77	0.44	**0.56**	**0.80**	**0.81**	**0.79**	**0.72**
Support Vector Machine (SVM)	**0.81**	**0.82**	0.92	0.87	0.69	**0.46**	**0.56**	0.78	0.79	0.78	0.71
XGBoost (XGB)	0.79	0.80	0.95	0.86	0.73	0.38	0.49	0.78	0.79	0.76	0.68
Random Forest (RF)	0.78	0.78	**0.98**	0.87	**0.85**	0.28	0.42	**0.80**	0.79	0.74	0.65
BiLSTM (Marcos Zampieri,2019)	-	0.83	0.95	0.89	0.81	0.48	0.60	0.82	0.82	0.81	0.75
SVM (Marcos Zampieri,2019)	-	0.80	0.92	0.86	0.66	0.43	0.52	0.76	0.78	0.76	0.69

embeddings when training the network, this baseline model achieved P, R and F1 in (offensive class) values of 66%, 57% and 61% however, 84%, 89% and 86% in (Not Offensive class) respectively, marking an improvement in R and F1 in (Offensive class) compared to the BiLSTM model, but at the expense of lower R and F1 in (Not Offensive class). While the P, R and F1 in (Not Offensive class), was slightly different compared to the BiLSTM and SVM model.

In the second approach, Word Level TF IDF Vectors were taken as feature embeddings to learn the Naïve Bayes model, resulting in clearly (99%) improved recall in (Not Offensive class) compared to the BiLSTM and SVM model, but at the expense of lower P and F1. While in (Offensive class), marking an improvement in P (89%) compared to the BiLSTM and SVM model, but at the expense of lower R and F1. Also, with these features, the P, R and F1 in (Weighted Average), was slightly different compared to the BiLSTM model. Table 4 shows the results of Hate speech language classification. Indeed, the best results were obtained with the RF, LR, SVM, and XGB models. Although, these models are considered to be among the simplest traditional machine learning models, it can yield satisfactory results on this task. While, table 5 and 6 below, presents the overall results of various techniques on the test set of

the evaluation dataset. It is interesting to note that the other deep learning techniques such as RoBERTa, also perform significantly better (82%, 83%, 81%, P, R, and F1 respectively) than SVM baseline algorithms.

Table 4. Final results of the classical baselines machine learning models, for the Hate speech dataset. The **bold** figures represent the best scores.

Hate speech Dataset (4 classes)																	
***		Abusive			Hateful			Normal			Spam			Weighted Average			***
Models	Acc	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	F1-M
Models on Count Vectors																	
NB	0.54	0.00	0.00	0.00	0.00	0.00	0.00	0.54	1.00	0.70	0.00	0.00	0.00	0.29	0.54	0.38	0.18
LR	0.54	0.00	0.00	0.00	0.00	0.00	0.00	0.54	1.00	0.70	0.00	0.00	0.00	0.29	0.54	0.38	0.18
SVM	0.54	0.00	0.00	0.00	0.00	0.00	0.00	0.54	1.00	0.70	0.00	0.00	0.00	0.29	0.54	0.38	0.18
XGB	0.54	0.00	0.00	0.00	0.00	0.00	0.00	0.54	1.00	0.70	0.00	0.00	0.00	0.29	0.54	0.38	0.18
RF	**0.81**	**0.87**	**0.91**	**0.89**	**0.65**	0.26	0.37	**0.82**	**0.91**	**0.86**	**0.63**	0.45	0.52	**0.80**	**0.81**	**0.80**	**0.66**
Models on Word Level TF IDF Vectors																	
NB	0.77	0.81	0.82	0.81	**0.82**	0.06	0.12	0.77	0.91	0.84	0.65	0.39	0.49	0.77	0.77	0.75	0.56
LR	**0.81**	**0.87**	**0.91**	**0.89**	0.68	0.23	0.34	0.81	0.93	**0.86**	0.65	0.38	0.48	**0.80**	**0.81**	0.79	0.64
SVM	**0.81**	**0.87**	**0.91**	**0.89**	0.65	0.26	0.37	**0.82**	0.91	**0.86**	0.63	0.45	0.52	**0.80**	**0.81**	**0.80**	**0.66**
XGB	0.79	0.86	0.88	0.87	0.72	0.21	0.33	0.76	**0.96**	0.85	**0.66**	0.12	0.20	0.77	0.79	0.74	0.56
RF	**0.81**	**0.87**	**0.91**	**0.89**	0.65	0.26	0.37	**0.82**	0.91	**0.86**	0.63	0.45	0.52	**0.80**	**0.81**	**0.80**	**0.66**
Models on Ngram Level TF IDF Vectors																	
NB	0.72	0.82	0.66	0.73	0.65	0.10	0.18	0.70	0.91	0.80	0.55	0.29	0.38	0.71	0.72	0.69	0.52
LR	**0.74**	0.85	0.70	**0.77**	**0.75**	0.12	0.21	**0.71**	0.95	**0.81**	**0.65**	0.21	0.32	**0.74**	**0.74**	**0.70**	0.53
SVM	**0.74**	0.85	0.70	**0.77**	0.68	0.14	0.23	**0.71**	0.95	**0.81**	0.62	0.21	0.32	0.73	**0.74**	**0.70**	0.53
XGB	0.66	**0.88**	0.42	0.57	0.70	0.10	0.18	0.63	**0.98**	0.77	0.63	0.08	0.14	0.70	0.66	0.60	0.41
RF	0.72	0.81	**0.71**	0.76	0.50	0.15	0.23	**0.71**	0.91	0.80	0.54	0.23	0.33	0.71	0.72	0.69	0.53
Models on Character Level TF IDF Vectors																	
NB	0.74	0.80	0.76	0.78	**0.81**	0.10	0.18	0.75	0.86	0.80	0.53	0.46	0.49	0.74	0.74	0.72	0.56
LR	0.80	**0.87**	**0.89**	**0.88**	0.70	0.20	0.31	0.80	0.93	**0.86**	0.63	0.36	0.46	**0.79**	0.80	0.78	0.63
SVM	**0.81**	**0.87**	**0.89**	**0.88**	0.66	0.24	0.36	**0.81**	0.92	**0.86**	0.62	0.41	0.49	**0.79**	**0.81**	**0.79**	**0.65**
XGB	0.80	**0.87**	**0.89**	**0.88**	0.73	0.21	0.33	0.78	**0.95**	**0.86**	**0.66**	0.24	0.36	**0.79**	0.80	0.77	0.60
RF	0.78	**0.87**	0.86	0.86	0.67	0.18	0.29	0.77	**0.95**	0.85	0.62	0.23	0.34	0.77	0.78	0.75	0.58

In general, results of RoBERTa and BiGRU are comparable, the best results were obtained with these two models, which is not surprising since these models are considered by many to be the state of the art. BiGRU also performs best for hate speech detection, followed by RoBERTa.

We compare our **BiGRU_Glove_FT** model with five baselines— (a) RoBERTa, (b) BiGRU with FastText only, (c) BiGRU with Glove only, (d) BiLSTM and SVM (Marcos Zampieri,2019). This model gives the best performance on F1-score which measures the overall quality of classification and also shows significant improvement over the baseline approaches as shown in table 5 below.

BiGRU_Glove_FT model achieved better Acc, P, R, and F1 values of (84%), (87%), (93%), and (90%) in Not Offensive classe, and P(77%), R(63%) and F1(69%) in Offensive classe respectively, marking an improvement in these metrics compared to RoBERTa model.

5. Discussion

Models are mainly affected by imbalance of the dataset, indeed, the models got confused between offensive and not offensive labels which suggests that these models are not able to capture the context in the sentence. In this work, we used class weights to improve class imbalance, and in order to have good results, we tried to use several hyper-parameters on the different models before submitting the final results.

Moreover, what we find the most interesting is the fact that both combinations of the word embedding (FasText + Glove) yield somewhat better results than RoBERTa or BiGRU with one word embedding only, indicating that this combination is useful. The results also prove that combining two word embedding such as Glove and FastText is better than individual word embedding and is especially helpful in improving precision as shown in the significant improvement of precision (more than 84% when compared to RoBERTa). In this paper, we find also several ML based solutions for the automatic detection have shown good results, such as NB, SVM, RF, and LR despite the difficulties that machine learning has.

Table 5. Final results of the baselines and our experiments, for the OLID datasets. The **bold** figures represent the best scores. Legend: 'F1-M' = F1-Macro.

OLID Dataset (2 classes)											
		Not			Off			Weighted Average			
Models	Acc	P	R	F1	P	R	F1	P	R	F1	F1-M
Baseline RoBERTa	**0.83**	0.84	**0.94**	**0.89**	**0.78**	0.53	0.63	0.82	**0.83**	0.81	0.76
Baseline BiGRU & Fasttext (FT) Embedding Only	**0.83**	**0.86**	0.92	**0.89**	0.74	**0.61**	0.67	**0.83**	**0.83**	**0.83**	**0.78**
Baseline BiGRU & Glove Embedding Only	**0.84**	**0.86**	0.93	**0.89**	0.76	**0.61**	**0.68**	**0.83**	**0.84**	**0.83**	**0.78**
BiLSTM (Marcos Zampieri,2019)	-	0.83	**0.95**	**0.89**	**0.81**	0.48	0.60	0.82	0.82	0.81	0.75
SVM (Marcos Zampieri,2019)	-	0.80	0.92	0.86	0.66	0.43	0.52	0.76	0.78	0.76	0.69
Our experiments: BiGRU model with GloVe and Fasttext embedding (BiGRU_Glove_FT)											
BiGRU_Glove_FT	**0.84**	**0.87**	**0.93**	**0.90**	**0.77**	**0.63**	**0.69**	**0.84**	**0.84**	**0.84**	**0.79**

Table 6. Final results of the baselines and our experiments, for the Hate speech dataset. The **bold** figures represent the best scores.

Hate speech Dataset (4 classes)																	
***	Abusive				Hateful			Normal			Spam			Weighted Average			***
Acc	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	F1-Macro
Baseline RoBERTa																	
0.81	**0.85**	**0.93**	**0.89**	0.59	0.32	0.42	**0.85**	**0.87**	**0.86**	**0.61**	0.53	**0.57**	**0.80**	**0.81**	**0.81**	**0.68**
Baseline BiGRU & Fasttext Embedding Only																	
0.76	**0.89**	**0.85**	**0.87**	0.38	0.53	0.45	**0.90**	**0.74**	**0.81**	0.47	**0.78**	0.59	**0.81**	**0.76**	**0.78**	**0.68**
Baseline BiGRU & Glove Embedding Only																	
0.37	0.35	**0.98**	0.51	0.00	0.00	0.00	0.00	0.00	0.00	0.45	**0.80**	0.58	0.16	0.37	0.22	0.27
Our experiments: BiGRU model with GloVe and Fasttext embedding (BiGRU_Glove_FT)																	
0.76	**0.88**	**0.85**	**0.87**	0.31	**0.59**	0.41	**0.89**	**0.74**	**0.81**	**0.51**	**0.72**	**0.60**	**0.81**	**0.76**	**0.78**	**0.67**

6. Conclusion and Future Work

This paper presented a novel system **(BiGRU_Glove_FT)** geared to automatically classify two kinds of online inappropriate content, focusing on Twitter. Indeed, the use of social networks motivated us to train contextual embedding based on the Twitter dataset, and use the information learned in this language model to identify offensive language and hate speech in the text. We tested some supervised machine learning classifier for hateful and offensive content in Twitter, we worked on two datasets such as OLID dataset, a new dataset with annotation of type and target of offensive language. It is the official dataset of the shared task SemEval 2019 Task 6: Identifying and Categorizing Offensive Language in Social Media [55], and Hate speech dataset which is the official dataset of the shared task: Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior [15]. The models with hyper-parameters yielded the best results in both tasks. We concluded that superior results on different NLP tasks have been produced by neural networks based on dense vector representations. This trend is triggered by the effectiveness of word embedding [29, 30] and methods of deep learning [47]. Deep learning enables multi-level automatic feature representation learning. In contrast, traditional machine learning based NLP systems liaise heavily on hand-crafted features. Such hand-crafted features are time-consuming and often incomplete. The classification results rely on deep learning models showed very high levels of performance at reducing false positives and produced promising results with respect to false neg-

atives. In the future, we plan to explore more the use of deep neural network architectures for the task of hate speech detection. We will investigate the application of other word embedding techniques. We would extend this focus to additional datasets such as Arabic dataset.

References

[1] Akhtar, S., Basile, V., Patti, V., 2020. Modeling annotator perspective and polarized opinions to improve hate speech detection, in: Proceedings of the AAAI Conference on Human Computation and Crowdsourcing, pp. 151–154.

[2] Alayba, A.M., Palade, V., England, M., Iqbal, R., 2018. A combined cnn and lstm model for arabic sentiment analysis, in: International cross-domain conference for machine learning and knowledge extraction, Springer. pp. 179–191.

[3] Alshari, E.M., Azman, A., Doraisamy, S., Mustapha, N., Alkeshr, M., 2018. Effective method for sentiment lexical dictionary enrichment based on word2vec for sentiment analysis, in: 2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP), IEEE. pp. 1–5.

[4] Banerjee, S., Chakravarthi, B.R., McCrae, J.P., 2020. Comparison of pretrained embeddings to identify hate speech in indian code-mixed text, in: 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), IEEE. pp. 21–25.

[5] Baykan, E., Henzinger, M., Marian, L., Weber, I., 2009. Purely url-based topic classification, in: Proceedings of the 18th international conference on World wide web, pp. 1109–1110.

[6] Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics 5, 135–146.

[7] Brown, S.E., Battle, J.S., 2019. Ostracizing targets of workplace sexual harassment before and after the# metoo movement. Equality, Diversity and Inclusion: An International Journal .

[8] Burnap, P., Williams, M.L., 2016. Us and them: identifying cyber hate on twitter across multiple protected characteristics. EPJ Data science 5, 1–15.

[9] Calado, P., Cristo, M., Moura, E., Ziviani, N., Ribeiro-Neto, B., Gonçalves, M.A., 2003. Combining link-based and content-based methods for web document classification, in: Proceedings of the twelfth international conference on Information and knowledge management, pp. 394–401.

[10] Collobert, R., Weston, J., 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning, in: Proceedings of the 25th international conference on Machine learning, pp. 160–167.

[11] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural language processing (almost) from scratch. Journal of machine learning research 12, 2493–2537.

[12] Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

[13] d'Sa, A.G., Illina, I., Fohr, D., 2020. Bert and fasttext embeddings for automatic detection of toxic speech, in: 2020 International Multi-Conference on:"Organization of Knowledge and Advanced Technologies"(OCTA), IEEE. pp. 1–5.

[14] Fišer, D., Erjavec, T., Ljubešić, N., 2017. Legal framework, dataset and annotation schema for socially unacceptable online discourse practices in slovene, in: Proceedings of the first workshop on abusive language online, pp. 46–51.

[15] Founta, A.M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., Kourtellis, N., 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. arXiv preprint arXiv:1802.00393 .

[16] Gambäck, B., Sikdar, U.K., 2017. Using convolutional neural networks to classify hate-speech, in: Proceedings of the first workshop on abusive language online, pp. 85–90.

[17] García-Díaz, J.A., Jiménez-Zafra, S.M., García-Cumbreras, M.A., Valencia-García, R., 2022. Evaluating feature combination strategies for hate-speech detection in spanish using linguistic features and transformers. Complex & Intelligent Systems , 1–22.

[18] Gaydhani, A., Doma, V., Kendre, S., Bhagwat, L., 2018. Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. arXiv preprint arXiv:1809.08651 .

[19] Giglou, H.B., Rahgooy, T., Razmara, J., Rahgouy, M., Rahgooy, Z., 2021. Profiling haters on twitter using statistical and contextualized embeddings, in: CLEF.

[20] Hinduja, S., Patchin, J.W., 2010. Bullying, cyberbullying, and suicide. Archives of suicide research 14, 206–221.

[21] Hu, B., Liu, B., Gong, N.Z., Kong, D., Jin, H., 2015. Protecting your children from inappropriate content in mobile apps: An automatic maturity rating framework, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 1111–1120.

[22] Huang, P.S., He, X., Gao, J., Deng, L., Acero, A., Heck, L., 2013. Learning deep structured semantic models for web search using clickthrough data, in: Proceedings of the 22nd ACM international conference on Information & Knowledge Management, pp. 2333–2338.

[23] Ibrohim, M.O., Setiadi, M.A., Budi, I., 2019. Identification of hate speech and abusive language on indonesian twitter using the word2vec, part of speech and emoji features, in: Proceedings of the International Conference on Advanced Information Science and System, pp. 1–5.

[24] Joulin, A., Grave, E., Bojanowski, P., Mikolov, T., 2016. Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 .

[25] Kovács, G., Alonso, P., Saini, R., 2021. Challenges of hate speech detection in social media. SN Computer Science 2, 1–15.

[26] Kumari, K., Singh, J.P., Dwivedi, Y.K., Rana, N.P., 2021. Multi-modal aggression identification using convolutional neural network and binary particle swarm optimization. Future Generation Computer Systems 118, 187–197.

[27] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 .

[28] MacAvaney, S., Yao, H.R., Yang, E., Russell, K., Goharian, N., Frieder, O., 2019. Hate speech detection: Challenges and solutions. PloS one 14, e0221152.

[29] Mikolov, T., Karafiát, M., Burget, L., 2010. Jan ˇcernocky, and sanjeev khudanpur. 2010. recurrent neural network based language model, in: Eleventh annual conference of the international speech communication association, pp. 1045–1048.
[30] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013. Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, pp. 3111–3119.
[31] Mollas, I., Chrysopoulou, Z., Karlos, S., Tsoumakas, G., 2020. Ethos: an online hate speech detection dataset. arXiv preprint arXiv:2006.08328 .
[32] Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y., 2016. Abusive language detection in online user content, in: Proceedings of the 25th international conference on world wide web, pp. 145–153.
[33] Oriola, O., Kotzé, E., 2020. Evaluating machine learning techniques for detecting offensive and hate speech in south african tweets. IEEE Access 8, 21496–21509.
[34] Pandey, Y., Sharma, M., Siddiqui, M.K., Yadav, S.S., 2022. Hate speech detection model using bag of words and naïve bayes, in: Advances in Data and Information Sciences. Springer, pp. 457–470.
[35] Pennington, J., Socher, R., Manning, C.D., 2014. Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532–1543.
[36] Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L., 2018. Deep contextualized word representations. arXiv preprint arXiv:1802.05365 .
[37] Pitsilis, G.K., Ramampiaro, H., Langseth, H., 2018. Detecting offensive language in tweets using deep learning. arXiv preprint arXiv:1801.04433 .
[38] Qin, D., Yu, J., Zou, G., Yong, R., Zhao, Q., Zhang, B., 2019. A novel combined prediction scheme based on cnn and lstm for urban pm 2.5 concentration. IEEE Access 7, 20050–20059.
[39] Raj, R., Srivastava, S., Saumya, S., 2020. Nsit & iiitdwd@ hasoc 2020: Deep learning model for hate-speech identification in indo-european languages., in: FIRE (Working Notes), pp. 161–167.
[40] Rajput, G., Punn, N.S., Sonbhadra, S.K., Agarwal, S., 2021. Hate speech detection using static bert embeddings, in: International Conference on Big Data Analytics, Springer. pp. 67–77.
[41] Ramraj, S., Uzir, N., Sunil, R., Banerjee, S., 2016. Experimenting xgboost algorithm for prediction and classification of different datasets. International Journal of Control Theory and Applications 9, 651–662.
[42] Rizos, G., Hemker, K., Schuller, B., 2019. Augment to prevent: short-text data augmentation in deep learning for hate-speech classification, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 991–1000.
[43] Romaissa Beddiar, D., Saroar Jahan, M., Oussalah, M., 2021. Data expansion using back translation and paraphrasing for hate speech detection. arXiv e-prints , arXiv–2106.
[44] Saha, P., Mathew, B., Goyal, P., Mukherjee, A., 2019. Hatemonitors: Language agnostic abuse detection in social media. arXiv preprint arXiv:1909.12642 .
[45] Sanh, V., Debut, L., Chaumond, J., Wolf, T., 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108 .
[46] Shen, Y., He, X., Gao, J., Deng, L., Mesnil, G., 2014. A latent semantic model with convolutional-pooling structure for information retrieval, in: Proceedings of the 23rd ACM international conference on conference on information and knowledge management, pp. 101–110.
[47] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C., 2013. Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the 2013 conference on empirical methods in natural language processing, pp. 1631–1642.
[48] Van Hee, C., Lefever, E., Verhoeven, B., Mennes, J., Desmet, B., De Pauw, G., Daelemans, W., Hoste, V., 2015a. Detection and fine-grained classification of cyberbullying events, in: International Conference Recent Advances in Natural Language Processing (RANLP), pp. 672–680.
[49] Van Hee, C., Verhoeven, B., Lefever, E., De Pauw, G., Daelemans, W., Hoste, V., 2015b. Guidelines for the fine-grained analysis of cyberbullying. Technical Report. version 1.0. Technical Report LT3 15-01, LT3, Language and Translation ....
[50] Wang, J., Yu, L.C., Lai, K.R., Zhang, X., 2016. Dimensional sentiment analysis using a regional cnn-lstm model, in: Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers), pp. 225–230.
[51] Waseem, Z., Davidson, T., Warmsley, D., Weber, I., 2017. Understanding abuse: A typology of abusive language detection subtasks. arXiv preprint arXiv:1705.09899 .
[52] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., Le, Q.V., 2019. Xlnet: Generalized autoregressive pretraining for language understanding. arXiv preprint arXiv:1906.08237 .
[53] Yenala, H., Jhanwar, A., Chinnakotla, M.K., Goyal, J., 2018. Deep learning for detecting inappropriate content in text. International Journal of Data Science and Analytics 6, 273–286.
[54] Young, T., Hazarika, D., Poria, S., Cambria, E., 2018. Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine 13, 55–75.
[55] Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R., 2019. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). arXiv preprint arXiv:1903.08983 .
[56] Zhang, L., Zhou, Y., Duan, X., Chen, R., 2018. A hierarchical multi-input and output bi-gru model for sentiment analysis on customer reviews, in: IOP conference series: materials science and engineering, IOP Publishing. p. 062007.
[57] Zhang, Z., Luo, L., 2019. Hate speech detection: A solved problem? the challenging case of long tail on twitter. Semantic Web 10, 925–945.