JTML at SemEval-2019 Task 6: Offensive Tweets Identification using Convolutional Neural Networks

Johnny Torres
ESPOL University
jomatorr@espol.edu.ec

Carmen Vaca
ESPOL University
cvaca@fiec.espol.edu.ec

Abstract

In this paper, we propose the use of a Convolutional Neural Network (CNN) to identify offensive tweets. We use an end-to-end model (i.e., no preprocessing) and fine-tune pre-trained embeddings (FastText) during training for learning words' representation. We compare the proposed CNN model to a baseline model, such as Linear Regression, and several neural models. The results show that CNN outperforms other models, and stands as a simple but strong baseline in comparison to other systems submitted to the Shared Task.

1 Introduction

The fast growth of online social networks (OSNs) has provided a medium for users to express their ideas and opinions about any topic. However, some users post offensive content which may deter other users from engaging in online discussions. Despite the tools provided by some OSNs to *block* other users and *report* offensive content, the manual verification of these events are limited in scale and costs due to a large number of malicious events performed by users or bots. Therefore, it is critical to developing automated tools to moderate the content that are robust to ambiguity, sarcasm, and adversarial attacks (Fortuna and Nunes, 2018).

Offensive language detection is an active research area, and several research efforts aim to contribute datasets, propose taxonomies, and improve current models to identify offensive content. In this direction, Zampieri et al. (2019b) proposed a shared task for Identifying and Categorizing Offensive Language in Social Media. The shared task is composed of the following subtasks: **a)** Offensive language identification, **b)** Automatic categorization of offense types, and **c)** Offense target identification.

tweet	Subtask A	B	C
If the tournament of shit ain't on here...	OFF	UNT	-
He is so full of BS!	OFF	TIN	IND
swear niggas make me wanna turn this phone off	OFF	TIN	GRP
Kick the absolute shite out of the car.	OFF	TIN	OTH

Table 1: Examples of Offensive Tweets in the dataset.

Table 1 shows some examples in the dataset of the shared task, and the labels in each of the subtasks. The labels indicate if the tweet is Offensive (OFF) and if it is an untargeted (UNT) or targeted (TIN) offense. The targets of the offensive tweets are individual (IND), group (GRP), other (OTH). This paper contributes specifically to the subtask A in the shared task.

The unstructured and noisy nature of user-generated content on OSNs poses a challenge for classification models. Traditional approaches use a sparse representation for text data, such as the bag of words (BOW) or TF-IDF (Manning et al., 2008).

We propose a model based on Convolutional Neural Networks (CNN) to identify and categorize offensive language on tweets. The learning representation relies on FastText pre-trained word embeddings (Mikolov et al., 2018). Although, this paper focus only on the first subtask, it can be extended to learn the other subtasks.

The rest of the paper describes related work in section 2. Then, we explain in detail our proposed model in section 3, and section 4 shows the results. Finally, we outline the conclusions and future work in section 5.

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 657–661
Minneapolis, Minnesota, USA, June 6–7, 2019.

2 Related Work

Previous work has studied several types of online misbehavior such as aggression (Cheng et al., 2015), cyberbullying (Pieschl et al., 2015), hate speech (Saleem et al., 2017), offensive, and abusive language identification (Waseem et al., 2017).

The major challenge in studying online misbehavior is the several forms it can take and the lack of a standard definition (Saleem et al., 2017). (Waseem et al., 2017) proposed a typology of abusive language sub-tasks. Similarly, a taxonomy proposed to detect toxic messages on Wikipedia discussion pages demonstrated the impact on community health both on and offline (Wulczyn et al., 2017). Wikimedia Foundation found that 54% of contributors decreased participation in the activities when they suffer harassment [1]. The same impact could happen on social media when aggression and offensive language deter other users from engaging in online discussions.

Previous works introduced several datasets like the Internet Argument Corpus (Walker et al., 2012) and the "Hate Speech Twitter Annotations" corpus (Waseem and Hovy, 2016). Most datasets are small in comparison to the Wikipedia dataset (Wulczyn et al., 2017), which enables to train neural models on a large-scale dataset.

In the multilingual aspect, several studies tackle languages other than English like Chinese (Su et al., 2017), Slovene (Fišer et al., 2017), and related shared tasks such as GermEval (Wiegand et al., 2018). However, the studies tackle each language individually due to the difficulty for automated systems to handle multiple languages as idiomatic expressions are dependent on the location and culture. Recent research on identifying profanity vs. hate speech highlighted the challenges of distinguishing between profanity and threatening language which may not contain profane language (Malmasi and Zampieri, 2018).

Recent surveys by (Schmidt and Wiegand, 2017) and (Fortuna and Nunes, 2018) summarizes the taxonomies and methods proposed for detecting abusive language. Also, recent work by (Davidson et al., 2017) introduces the Hate Speech Detection dataset used in several studies (Malmasi and Zampieri, 2017; ElSherief et al., 2018; Zhang et al., 2018).

[1] https://upload.wikimedia.org/wikipedia/commons/5/52/Harassment_Survey_2015_-_Results_Report.pdf

3 Methodology

The model architecture, shown in Figure 1, is a slight variant of the CNN architecture proposed by Kim (2014). We define $\mathbf{x}_i \in \mathbb{R}^k$ as the k-dimensional word vector (i.e., word embeddings) corresponding to the i-th word in the tweets. We padded the tweets to make all equal length, and represent a tweet of length n as

$$\mathbf{x}_{1:n} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \ldots \oplus \mathbf{x}_n, \quad (1)$$

where $\oplus$ is the concatenation operator. In general, we refer $\mathbf{x}_{i:i+j}$ to the concatenation of words $\mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_{i+j}$. Then, we apply a convolution operation that uses a *filter* $\mathbf{w} \in \mathbb{R}^{hk}$, over a window of h words to produce a new feature. For example, we generate feature c_i from a window of words $\mathbf{x}_{i:i+h-1}$ by

$$c_i = f(\mathbf{w} \cdot \mathbf{x}_{i:i+h-1} + b). \quad (2)$$

We denote $b \in \mathbb{R}$ as the bias term and f as a RELU activation function defined as $f(x) = x^+ = \max(0, x)$, where x in the input to the neuron (Glorot et al., 2011) . The convolution layer applies the filter to each possible window of words in the sentence $\{\mathbf{x}_{1:h}, \mathbf{x}_{2:h+1}, \ldots, \mathbf{x}_{n-h+1:n}\}$ to produce a *feature map*

$$\mathbf{c} = [c_1, c_2, \ldots, c_{n-h+1}], \quad (3)$$

with $\mathbf{c} \in \mathbb{R}^{n-h+1}$. Then, we apply a max-over-time pooling operation over the feature map and take the maximum value $\hat{c} = \max\{\mathbf{c}\}$ as the feature corresponding to this particular filter. The goal is to capture the essential feature (the highest feature value) for the feature maps. The pooling scheme allows us to deal with variable sentence lengths.

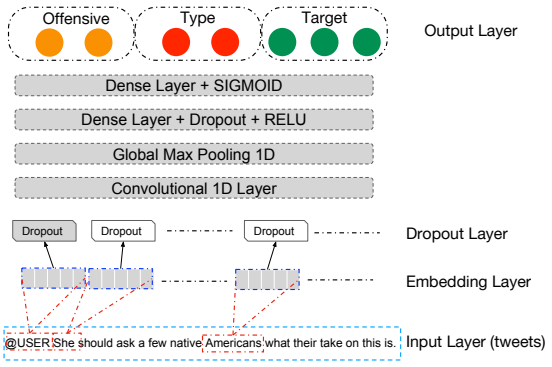

Figure 1: The CNN architecture used to identify offensive tweets using binary output layer.

We have described the process by which we extract *one* feature from *one* filter. The model uses multiple filters to obtain multiple features. These features feed a fully connected layer with a RELU activation function, and finally a sigmoid layer that outputs the probability distribution over labels.

For regularization, we employ a dropout layer with rate $r = 0.2$, constrained on l_2-norms of the weight vectors. We apply the dropout after the embeddings and the penultimate layer. Dropout prevents co-adaptation of hidden units by randomly dropping out (i.e., set to zero) a proportion of p of the hidden units during forward-backpropagation. That is, given the penultimate layer $\mathbf{z} = [\hat{c}_1, \ldots, \hat{c}_m]$ (note that here we have m filters), instead of using

$$y = \mathbf{w} \cdot \mathbf{z} + b \tag{4}$$

for output unit y in forward propagation, dropout uses

$$y = \mathbf{w} \cdot (\mathbf{z} \circ \mathbf{r}) + b, \tag{5}$$

where $\circ$ is the element-wise multiplication operator and $\mathbf{r} \in \mathbb{R}^m$ is a *masking* vector of Bernoulli random variables with probability p of being 1. Gradients are backpropagated only through the unmasked units. At test time, the learned weight vectors are scaled by p such that $\hat{\mathbf{w}} = p\mathbf{w}$, and $\hat{\mathbf{w}}$ is used (without dropout) to score unseen sentences. We additionally constrain l_2-norms of the weight vectors by rescaling $\mathbf{w}$ to have $||\mathbf{w}||_2 = s$ whenever $||\mathbf{w}||_2 > s$ after a gradient descent step.

4 Experiments

In this section, we describe the experimental settings and the results for the subtask A: identifying offensive tweets. We use the data provided in the shared task OffensEval described in Zampieri et al. (2019a). Table 2 describe the label distribution for each of the subtasks. We use the F_1 score as an evaluation metric for the models, and it is the official ranking metric for the shared task is macro-averaged F1.

Subtask A	tweets
NOT	8840
OFF	4400

Table 2: Distribution of the labels in the training dataset.

4.1 Embeddings

We evaluate the CNN model with several word embeddings such as: **a)** Random Uniform initialized, **b)** Word2Vec (Mikolov et al., 2013), and **c)** FastText (Mikolov et al., 2018).

During training, we fine-tune the embedding layer for each type of embeddings. Table 3 shows that FastText embeddings provide the best results, and we use it in further experiments.

Embedding	Precision	Recall	F_1
Random	71.69	69.47	70.23
Word2Vec	70.67	70.15	70.38
FastText	**71.76**	**71.97**	**71.86**

Table 3: Evaluation of different embeddings.

4.2 Models

We compare the CNN model against baseline models such.

Logistic Regression (LR) with *liblinear* solver and *class weight* to account for the imbalance of the labels.

FastText as a simple and efficient baseline for text classification, and often on par with deep learning classifiers regarding the accuracy but orders of magnitude faster for training and evaluation (Joulin et al., 2016).

LSTM in its vanilla implementation (Tang et al., 2015), with one LSTM layer after the Embeddings Layer.

Bi-LSTM implements a bi-directional LSTM architecture (Zhou et al., 2016).

Table 4 shows the performance of cross-validation data for the proposed CNN model and the baseline models. For evaluation purposes, we split the training dataset in 80% for training subset and 20% testing subset. We use k fold cross validation ($k = 10$) on the training subset. The CNN model outperforms other models in detecting Offensive Tweets and the overall Macro F_1 but detecting Not Offensive tweets works better with Bi-LSTM. We found that the non-neural LR model outperforms neural models such as FastText and LSTM. Bi-LSTM and CNN performance

	Not Offensive			Offensive			Macro		
Model	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
LR	81.80	75.90	78.74	58.27	66.59	62.15	70.03	71.24	70.45
CNN	81.33	80.50	80.91	62.18	63.44	**62.81**	71.76	71.97	**71.86**
LSTM	77.73	78.97	78.34	57.03	55.23	56.11	67.38	67.10	67.23
Bi-LSTM	80.68	81.92	**81.30**	63.11	61.19	62.14	71.90	71.56	71.72
FastText	77.87	79.02	78.44	57.24	55.57	56.39	67.56	67.30	67.42

Table 4: Benchmark of supervised learning models. CNN yields the best performance based on the metric F_1.

are on pair, and further evaluation of the hyper-parameters (e.g., number of layers/neurons, activation functions) is required to determine which of them performs better.

Table 5 show the results in the shared task evaluated on the testing dataset for subtask A. We include a random baseline generated by assigning the same labels for all instances. For example, "All OFF" in sub-task A represents the performance of a system that labels everything as offensive. The CNN model outperforms by a large margin the random baseline.

System	F_1 (macro)	Accuracy
All NOT baseline	0.4189	0.7209
All OFF baseline	0.2182	0.2790
CNN	**0.7591**	**0.8105**

Table 5: Results for Sub-task A using CNN model compared to simple baseline.

Figure 2 shows the confusion matrix for the results with our CNN model. Due to the imbalance in the labels, the False Negatives in the results affects by a large margin the F_1 macro score.

5 Conclusion

In this paper, we proposed a neural model based on Convolutional Neural Networks to identify and categorize offensive tweets on social media. The model outperforms baseline models and other Sequential Models such as LSTM and Bi-LSTM. The reason CNN perform better than sequential models could be due to the noisy and unstructured form of the tweets.

In future work, we plan to use several variations of CNN such as multi-channel and multi-view architectures. Also, we will use recent advances in learning representations based on deep contextualized embeddings such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018).

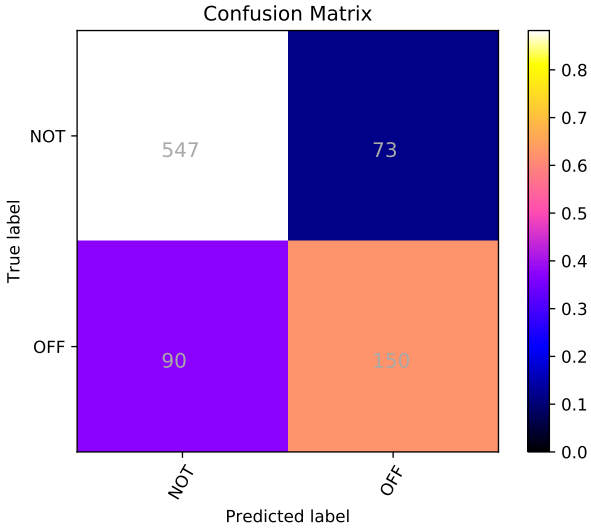

Figure 2: Confusion matrix for Sub-task A, JTML CodaLab CNN model.

References

Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. 2015. Antisocial behavior in online discussion communities. In *Icwsm*, pages 61–70.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated Hate Speech Detection and the Problem of Offensive Language. In *Proceedings of ICWSM*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mai ElSherief, Vivek Kulkarni, Dana Nguyen, William Yang Wang, and Elizabeth Belding. 2018. Hate Lingo: A Target-based Linguistic Analysis of Hate Speech in Social Media. *arXiv preprint arXiv:1804.04257*.

Darja Fišer, Tomaž Erjavec, and Nikola Ljubešić. 2017. Legal Framework, Dataset and Annotation Schema for Socially Unacceptable On-line Discourse Practices in Slovene. In *Proceedings of the Workshop Workshop on Abusive Language Online (ALW)*, Vancouver, Canada.

Paula Fortuna and Sérgio Nunes. 2018. A Survey on Automatic Detection of Hate Speech in Text. *ACM Computing Surveys (CSUR)*, 51(4):85.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Shervin Malmasi and Marcos Zampieri. 2017. Detecting Hate Speech in Social Media. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP)*, pages 467–472.

Shervin Malmasi and Marcos Zampieri. 2018. Challenges in Discriminating Profanity from Hate Speech. *Journal of Experimental & Theoretical Artificial Intelligence*, 30:1–16.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. Scoring, term weighting and the vector space model. *Introduction to information retrieval*, 100:2–4.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Stephanie Pieschl, Christina Kuhlmann, and Torsten Porsch. 2015. Beware of publicity! perceived distress of negative cyber incidents and implications for defining cyberbullying. *Journal of School Violence*, 14(1):111–132.

Haji Mohammad Saleem, Kelly P. Dillon, Susan Benesch, and Derek Ruths. 2017. A web of hate: Tackling hateful speech in online social spaces. *CoRR*, abs/1709.10159.

Anna Schmidt and Michael Wiegand. 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. Association for Computational Linguistics*, pages 1–10, Valencia, Spain.

Huei-Po Su, Chen-Jie Huang, Hao-Tsung Chang, and Chuan-Jie Lin. 2017. Rephrasing Profanity in Chinese Text. In *Proceedings of the Workshop Workshop on Abusive Language Online (ALW)*, Vancouver, Canada.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1422–1432.

Marilyn A Walker, Jean E Fox Tree, Pranav Anand, Rob Abbott, and Joseph King. 2012. A corpus for research on deliberation and debate. In *LREC*, pages 812–817.

Zeerak Waseem, Thomas Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding Abuse: A Typology of Abusive Language Detection Subtasks. In *Proceedings of the First Workshop on Abusive Language Online*.

Zeerak Waseem and Dirk Hovy. 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.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language. In *Proceedings of GermEval*.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1391–1399. International World Wide Web Conferences Steering Committee.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of NAACL*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval)*.

Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network. In *Lecture Notes in Computer Science*. Springer Verlag.

Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu. 2016. Text classification improved by integrating bidirectional lstm with two-dimensional max pooling. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3485–3495.