NUIG at SemEval-2020 Task 12: Pseudo labelling for offensive content classification

Shardul Suryawanshi, Mihael Arcan, Paul Buitelaar
Insight SFI Research Centre for Data Analytics
Data Science Institute, National University of Ireland Galway
shardul.suryawanshi@insight-centre.org

Abstract

This work addresses the classification problem defined by sub-task A (English only) of the OffensEval 2020 challenge. We used a semi-supervised approach to classify given tweets into an offensive (OFF) or not-offensive (NOT) class. As the `OffensEval 2020` dataset is loosely labelled with confidence scores given by unsupervised models, we used last year's offensive language identification dataset (`OLID`) to label the `OffensEval 2020` dataset. Our approach uses a pseudo-labelling method to annotate the current dataset. We trained four text classifiers on the `OLID` dataset and the classifier with the highest macro-averaged F1-score has been used to pseudo label the `OffensEval 2020` dataset. The same model which performed best amongst four text classifiers on `OLID` dataset has been trained on the combined dataset of `OLID` and pseudo labelled `OffensEval 2020`. We evaluated the classifiers with precision, recall and macro-averaged F1-score as the primary evaluation metric on the `OLID` and `OffensEval 2020` datasets.

1 Introduction

We are generating data at a rate which has never been seen before[1], while social media plays a big role in our everyday life. Most people with smartphones have a social media account and an active social media life. However, due to anonymity and freedom of expression, social media has become a breeding ground for offensive content. This content can be deemed as offensive if it intends to demean a person or a group of people due to their ethnicity, sexual orientation or by a personal attack[2]. This raises the need for an evaluation system that could automatically detect such offensive text.

Sub-task A (English only) in the OffensEval 2020 (Zampieri et al., 2020) provided the `OffensEval 2020` dataset (Zampieri et al., 2019a) of English tweets, which has been labelled with average confidence and standard deviation of the confidence scores, e.g. "@USER His ass need to stay up" has an average confidence of 0.833496 while standard deviation is 0.140625. Rosenthal et al. (2020) used an unsupervised learning approach to produce these confidence scores. Since the system is evaluated on its label prediction performance (offensive *OFF*, not-offensive *NOT*) and not on these scores, it was up to the participant to decide how to best map the scores to labels. We trained different text classifiers on the `OLID` (Zampieri et al., 2019a) dataset to predict the labels of the `OffensEval 2020`. This pseudo-labelling approach avoids the efforts involved in the manual adjustment of threshold values of average confidence (*Avg conf*) and average standard deviation (*Avg std*).

2 Related work

There has been significant research on aggression detection, hate speech detection (Ranjan et al., 2016; Rani et al., 2020; Jose et al., 2020) and cyberbullying (Schmidt and Wiegand, 2017). Based on this

[1] https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/#75d1769a60ba

[2] https://dictionary.cambridge.org/us/dictionary/english/hate-speech

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 1598–1604
Barcelona, Spain (Online), December 12, 2020.

Parameters	Stats
The number of tweets/samples	9,075,418
Average word count per tweet	15.64
Average functional words count per tweet	**5.61**
Average Hashtags per tweet	0.08

Table 1: Data statistics of the OffensEval 2020 dataset

survey different types of features have been employed by previous works including surface, word generalization, sentiment-based (Chakravarthi et al., 2020a; Chakravarthi et al., 2020b), lexical, code-mixed (Priyadharshini et al., 2020), linguistic, knowledge-based and multimodal information features (Suryawanshi et al., 2020a; Suryawanshi et al., 2020b) as well. Traditional machine learning (ML) approaches such as support vector machines (SVM) by Perelló et al. (2019) can be trained on hate speech tweets by identifying n-grams features which could be improved further by combining word embedding with sentiment features. Research by Kebriaei et al. (2019) shows how a convolutional neural network (CNN) shows higher macro averaged F1-score than traditional ML approaches such as SVM, random forest (RF) and naive Bayes (NB). (Rajendran et al., 2019) uses an ensemble of classifiers to classify the offensive text in an imbalanced dataset by using models with Synthetic Minority Over-sampling technique (SMOTE). Singh and Chand (2019) uses sequence to sequence models combined with long short term memory (LSTM) network, gated recurrent unit (GRU) and Bidirectional LSTM (BiLSTM) to classify a given tweet into an offensive (OFF) or not-offensive (NOT) class.

Hybrid approaches which combine a recurrent neural network (RNN) and a CNN have been proven to be better at text classification. Rhanoui et al. (2019) has designed such an approach for sentiment analysis. In their research, they combined multiple outputs of a convolutional filter to form a vector representation of text, which was then fed to a BiLSTM. Dynamic meta-embedding (*DME*) and Contextualised DME (*CDME*) introduced by Kiela et al. (2018) has shown significant improvement in a variety of natural language processing tasks such as natural language inference, sentiment classification and image-caption retrieval. Pre-trained Bidirectional Encoder Representations from Transformer (*BERT*) (Devlin et al., 2019) based models (*BERT*) have performed exceptionally well in many natural language processing tasks such as text classification, natural language generation and machine translation. We designed our experiments based on *CNN + BiLSTM*, *CDME*, *DME* and *BERT*.

3 Methodology

In this section, we are giving insights about the methodology followed to process and pseudo-label the data in Subsection 3.1 and Subsection 3.2 respectively.

3.1 Data Statistics and Pre-processing Steps

The data statistics in Table 1 show that on average, a high number of functional words[3] per tweet are present in the dataset. We decided not to remove them in pre-processing as they might add valuable information to the tweet. Inspired by the OffensEval 2019 (Zampieri et al., 2019b) submissions, we have pre-processed the data as follows:

I Instead of converting each token to a lower case, we kept the true case of the tokens. This has been done deliberately as people tend to use capital casing to emphasise their opinion.

II Emoticons play a vital role in conveying the sentiment behind the text. We leveraged this property of emoticons by translating all them to text using the *unidecode* library[4]. E.g. "@USER His ass need to stay up ☺☺" has been translated into "'@USER His ass need to stay up FACE WITH TEARS OF JOY FACE WITH TEARS OF JOY"'

[3]https://www.nltk.org

[4]https://pypi.org/project/Unidecode/

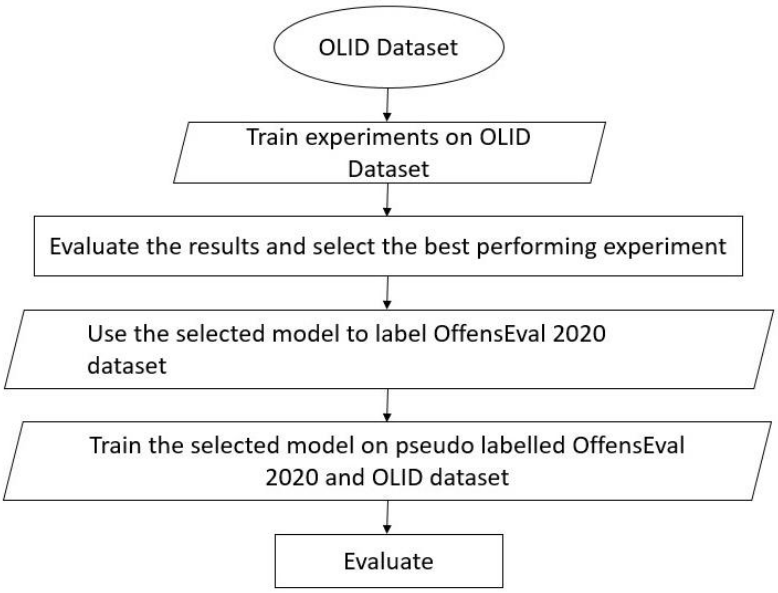

Figure 1: Pseudo-labelling and training source.

III We converted hashtags into words by removing the hash symbol.

IV For the text classifiers other than those using *BERT*, the maximum length of the tweet has been restricted to 64 word tokens. If the tweet exceeds then we truncated it or else we padded zeros to the left to match the length.

3.2 Pseudo-labelling

We used the `OLID` dataset as a reference dataset for labelling the `OffensEval 2020` dataset. The `OLID` dataset is relatively small with 14,100 observations in comparison with the `OffensEval 2020` dataset, which has 9,075,418 observations. We performed four experiments (explained in Section 4) on the `OLID` dataset. The performance of these experiments has been evaluated with macro-averaged F1-score and the approach with the highest score was chosen to label the `OffensEval 2020` dataset. This same model has been trained on the combined dataset of `OLID` and `OffensEval 2020` datasets. Figure 1 illustrates this approach. As both of the datasets are composed of English tweets, the experiments used to model one dataset can yield a similar result for the other dataset. The experiments take less time to execute on the `OLID` dataset because of its smaller size when compared with the `OffensEval 2020` dataset. Hence experiments performed on the `OLID` dataset, can similarly predict their results on the `OffensEval 2020` dataset. This approach saves a lot of time by avoiding training on a bigger dataset.

4 Experiments

We outline the experiments performed on the `OLID` dataset and the combined dataset `OffensEval 2020` + `OLID` in Subsection 4.1 and Subsection 4.2 respectively.

4.1 Training on `OLID` dataset

The experiments listed below are performed on the `OLID` dataset to select the classifier which can be used to label the `OffensEval 2020` dataset.

CNN+BiLSTM: In this architecture, we extracted the abstract features from the text vector using a one dimensional (1D) convolution network and 1D maxpooling, which later has been fed to the BiLSTM to form a vector representation of the tweet. This vectorised text represents the word with its long and short term context in a vector space. Abstract and prominent features captured by the CNN have been contextualised with time steps with the RNN. We used pre-trained GloVe 50d Twitter crawled embeddings (Pennington et al., 2014).

DME and CDME: We used *DME* and *CDME* as an ensemble of embeddings to study if it gives better results with the OLID dataset. This architecture takes advantage of both Word2vec (Mikolov et al., 2013) and FastText (Bojanowski et al., 2017) embeddings to learn the vector representation of the word. The Word2vec and FastText embedding of the same words, later on, are projected on the embedding matrix space. These projected vectors are later concatenated. Weights of each embedding have been learned as a hyperparameter using the self-attention model. Unlike *DME*, *CDME* has BiLSTM incorporated in its architecture.

BERT: In this experiment, we are using a *BERT* based model pre-trained on Wikipedia. We used *BERT-BASE uncased* architecture with 12 layers, 768 hidden units, 12 attention heads, which has 110M parameters. The Simple Transformer library[5] has been used to implement this architecture. This pre-trained transformer with stacked encoders learns the vector representation of the tweet irrespective of its length. Each encoder layer applies self-attention to form a vector representation and passes it to the next layer with a feed-forward network. We expect this representation to stand a better chance in identifying the OFF or NOT tweet.

We performed all experiments with the same hyperparameter settings, i.e., 20 epochs, binary cross-entropy, adam optimiser (learning rate = 0.001) with sigmoid activation for the classification layer and Relu as an activation function for the intermediate layers. The number of instances of offensive text is higher when compared with instances of not-offensive text. This leads to class imbalance. To address this issue, we used class weights. These weights are adjusted inversely proportional to the training data class frequencies. The dataset has been split into a train (80%), test (10%) and validation (10%) set.

4.2 Training on OffensEval 2020 + OLID dataset

BERT based model was found to be the best performing model on the OLID dataset and has therefore been used to label the OffensEval 2020 dataset. Later, the same *BERT* model has been trained on the combined data of the pseudo labelled OffensEval 2020 and OLID datasets with two epochs by keeping the rest of the parameters the same as mentioned in Section 4.1. To avoid overfitting, we re-trained the *BERT* model on the combined dataset without prior knowledge of hyperparameter settings tuned on the OLID dataset.

5 Results

We have divided this section into two subsections. In the Subsection 5.1, we report on results achieved on the OLID dataset. Subsection 5.2 discusses the performance of *BERT* on the OffensEval dataset.

5.1 Results on OLID dataset

As per the results shared in Table 2, the macro-averaged F1-score with *BERT* on the OLID dataset is the highest, i.e. 0.76. The precision, recall and F1-score for the experiments other than *BERT* shows that they failed significantly to correctly identify the offensive text. Even after using class weights, *CDME*, *DME*, *CNN + BiLSTM* fail to give good results. The poor macro-averaged F1-score achieved by these experiments eliminate them as a choice to pseudo-label the OffensEval 2020 dataset.

5.2 Results on OffensEval 2020 dataset

The results in Table 3 show that *BERT* achieved a macro-averaged F1-score of 0.89 on the imbalanced test set (# of NOT: 2,807, # of OFF: 1,080). Nevertheless, *BERT* still manages to achieve a better precision (NOT: 0.97, OFF: 0.78) and recall (NOT: 0.90, OFF: 0.93) for both classes. Figure 2 shows that *BERT* correctly identified 1,005 OFF tweets out of 1,080 offensive tweets. Only 75 tweets are incorrectly labelled as NOT. The mislabelled tweets have offensive words which gave them a negative connotation. For example, "*One thing I hate most is a liar*" has been labelled as offensive. This tweet contains strong offensive words such as "*hate*" and "*liar*", but it is not offensive when the context of the whole tweet is taken into consideration. On the other hand, the tweet "*@USER I thought you magas refused to use Nike because they don't hate black people*" has been mislabelled as NOT. The word "*hate*" in the context of

[5]https://github.com/ThilinaRajapakse/simpletransformers

	CNN + BiLSTM				BERT			
	Precision	Recall	F1-score	count	Precision	Recall	F1-score	count
NOT	0.67	0.94	0.78	890	0.84	0.84	0.84	890
OFF	0.32	0.06	0.10	434	0.68	0.67	0.67	434
macro	0.50	0.50	**0.44**	1,324	0.76	0.76	**0.76**	1,324
weighted	0.56	0.65	0.56	1,324	0.79	0.79	0.79	1,324
	DME				**CDME**			
NOT	0.67	1.00	0.80	890	0.67	0.94	0.79	890
OFF	0.00	0.00	0.00	434	0.37	0.07	0.12	434
macro	0.34	0.50	**0.40**	1,324	0.52	0.51	**0.45**	1,324
weighted	0.45	0.67	0.54	1,324	0.58	0.66	0.57	1,324

Table 2: Precision, recall, F1-score and count for CNN + BiLSTM, BERT, DME, CDME with OLID dataset.

	BERT			
	Precision	Recall	F1-score	count
NOT	0.97	0.90	0.93	2,807
OFF	0.78	0.93	0.85	1,080
macro	0.88	0.92	**0.89**	3,887
weighted	0.92	0.91	0.91	3,887

Table 3: Precision, recall, F1-score and count for BERT trained OffensEval 2020 dataset.

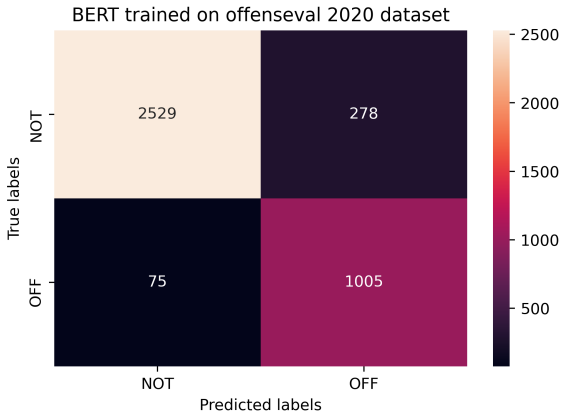

Figure 2: Confusion matrix for BERT trained on OffensEval 2020 dataset.

"*Don't*" is positive but as it is targeting a specific group makes this tweet offensive. High precision, recall and F1-score in Table 2 and Table 3 show that *BERT* generalises better on both the datasets.

6 Conclusion

With an macro-averaged F1-score of 0.89 using *BERT* on the `OffensEval 2020` test set, we showed that pseudo-labelling can help to create a model trained on a loosely labelled dataset. The process of training the model on the small dataset, before training on a similar, yet bigger dataset, is more time efficient compared to a manual annotation. Manual adjustments of the thresholds is further not straightforward and it is rather difficult to analyze larger datasets manually. Pseudo-labelling solves this issue by automatically annotating the data. Nevertheless, pseudo-labelling is just one approach with which this issue of unlabelled data could be solved. K-nearest neighbour (KNN) with two clusters can be used to label the data as well. For our future work, we will concentrate on feature engineering, focusing mainly on the Uniform Resource Locators (URLs). As URLs in the tweets were removed while processing the data, we think that they could be useful if they are normalised and considered as a token.

Acknowledgments

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2, co-funded by the European Regional Development Fund, as well as by the H2020 project Prêt-à-LLOD under Grant Agreement number 825182.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bharathi Raja Chakravarthi, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John P McCrae. 2020a. A sentiment analysis dataset for code-mixed Malayalam-English. In *Proceedings of the 1st Joint Workshop of SLTU (Spoken Language Technologies for Under-resourced languages) and CCURL (Collaboration and Computing for Under-Resourced Languages) (SLTU-CCURL 2020)*, Marseille, France, May. European Language Resources Association (ELRA).

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John P McCrae. 2020b. Corpus creation for sentiment analysis in code-mixed Tamil-English text. In *Proceedings of the 1st Joint Workshop of SLTU (Spoken Language Technologies for Under-resourced languages) and CCURL (Collaboration and Computing for Under-Resourced Languages) (SLTU-CCURL 2020)*, Marseille, France, May. European Language Resources Association (ELRA).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota.

Navya Jose, Bharathi Raja Chakravarthi, Shardul Suryawanshi, Elizabeth Sherly, and John P. McCrae. 2020. A survey of current datasets for code-switching research. In *2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS)*.

Emad Kebriaei, Samaneh Karimi, Nazanin Sabri, and Azadeh Shakery. 2019. Emad at SemEval-2019 task 6: Offensive language identification using traditional machine learning and deep learning approaches. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 600–603, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Douwe Kiela, Changhan Wang, and Kyunghyun Cho. 2018. Dynamic meta-embeddings for improved sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1466–1477, Brussels, Belgium, October-November. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Carlos Perelló, David Tomás, Alberto Garcia-Garcia, Jose Garcia-Rodriguez, and Jose Camacho-Collados. 2019. UA at SemEval-2019 task 5: Setting a strong linear baseline for hate speech detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 508–513, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Mani Vegupatti, and John P. McCrae. 2020. Named entity recognition for code-mixed Indian corpus using meta embedding. In *2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS)*.

Arun Rajendran, Chiyu Zhang, and Muhammad Abdul-Mageed. 2019. UBC-NLP at SemEval-2019 task 6: Ensemble learning of offensive content with enhanced training data. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 775–781, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Priya Rani, Shardul Suryawanshi, Koustava Goswami, Bharathi Raja Chakravarthi, Theodorus Fransen, and John P McCrae. 2020. A comparative study of different state-of-the-art hate speech detection methods for Hindi-English code-mixed data. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, Marseille, France, May. European Language Resources Association (ELRA).

P. Ranjan, B. Raja, R. Priyadharshini, and R. C. Balabantaray. 2016. A comparative study on code-mixed data of Indian social media vs formal text. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 608–611, Dec.

Maryem Rhanoui, Mounia Mikram, Siham Yousfi, and Soukaina Barzali. 2019. A cnn-bilstm model for document-level sentiment analysis. *Machine Learning and Knowledge Extraction*, 1:832–847, 07.

Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Marcos Zampieri, and Preslav Nakov. 2020. A Large-Scale Weakly Supervised Dataset for Offensive Language Identification. In *arxiv*.

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*, pages 1–10, Valencia, Spain, April. Association for Computational Linguistics.

Pardeep Singh and Satish Chand. 2019. Pardeep at SemEval-2019 task 6: Identifying and categorizing offensive language in social media using deep learning. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 727–734, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Shardul Suryawanshi, Bharathi Raja Chakravarthi, Mihael Arcan, and Paul Buitelaar. 2020a. Multimodal meme dataset (MultiOFF) for identifying offensive content in image and text. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, Marseille, France, May. European Language Resources Association (ELRA).

Shardul Suryawanshi, Bharathi Raja Chakravarthi, Pranav Verma, Mihael Arcan, John P McCrae, and Paul Buitelaar. 2020b. A dataset for troll classification of Tamil memes. In *Proceedings of the 5th Workshop on Indian Language Data Resource and Evaluation (WILDRE-5)*, Marseille, France, May. European Language Resources Association (ELRA).

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 the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1415–1420.

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*, pages 75–86.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of SemEval*.