CAMsterdam at SemEval-2019 Task 6: Neural and graph-based feature extraction for the identification of offensive tweets

Guy Aglionby†, Christopher Davis†, Pushkar Mishra‡, Andrew Caines†, Helen Yannakoudakis†, Marek Rei†, Ekaterina Shutova* & Paula Buttery†
† Department of Computer Science & Technology, University of Cambridge, U.K.
{ga384,ccd38,apc38,hy260,mr472,pjb48}@cam.ac.uk
‡ Facebook AI, London, U.K.
pushkarmishra@fb.com
* Institute for Logic, Language and Computation, University of Amsterdam, Netherlands
e.shutova@uva.nl

Abstract

We describe the CAMsterdam team entry to the SemEval-2019 Shared Task 6 on offensive language identification in Twitter data. Our proposed model learns to extract textual features using a multi-layer recurrent network, and then performs text classification using gradient-boosted decision trees (GBDT). A self-attention architecture enables the model to focus on the most relevant areas in the text. We additionally learn globally optimised embeddings for hashtags using node2vec, which are given as additional tweet features to the GBDT classifier. Our best model obtains 78.79% macro F1-score on detecting offensive language (subtask A), 66.32% on categorising offence types (targeted/untargeted; subtask B), and 55.36% on identifying the target of offence (subtask C).

1 Introduction

The SemEval-2019 shared task 6 ('OffensEval') involved three sub-parts: the classification of tweets as offensive or not (subtask A), classifying whether they are targeted insults or not (subtask B), and finally whether the targeted insults are aimed at an individual, group or otherwise (subtask C). Further details may be found in the shared task report (Zampieri et al., 2019b). Here we describe CAMsterdam's competition entry.

In recent years, there has been a growing interest in the automatic detection of offensive opinions expressed in online texts, including those posted in discussion forums, news article comment sections, and social networks. Such detection is not straightforwardly a matter of identifying texts containing obscene words (Malmasi and Zampieri, 2018); offensiveness often arises from the context, current affairs, world knowledge, the use of acronyms and slang, and the identity of the authors and audience. Therefore the task is a challenging one, but one with real world impact: if measures can be taken to identify and curtail trolling, the toxicity of the internet can to some extent be reduced. There is evidence that online harassment is connected with oppression, violence and suicide (Dinakar et al., 2011; Sood et al., 2012; Wulczyn et al., 2017), and there may moreover be reasons for concern about the perpetrator's wellbeing along with that of the victims (Cheng et al., 2017).

Our approach to the task extends the work of Mishra et al. (2018b), who extract features from tweets using an RNN for subsequent use in a gradient-boosted decision tree (GBDT) (Ke et al., 2017). Firstly, we experiment with changes to the RNN, including the use of self-attention (Rei and Søgaard, 2019) and ELMo embeddings (Peters et al., 2018). Secondly, we add additional features to the GBDT, including globally-optimised hashtag embeddings learned from a graph of tweet contents using node2vec (Grover and Leskovec, 2016). We show that this method of learning distributional information about hashtags improves performance over just learning their embeddings within a RNN.

2 Related Work

There has been much work characterising offensive online discourse including hate speech and cyberbullying (Warner and Hirschberg, 2012; Kwok and Wang, 2013; Xu et al., 2013; Waseem et al., 2017; Ribeiro et al., 2018). This work also includes creating datasets for training and evaluating detection models, for example the Hate Speech Twitter Annotations and Wikipedia Comments Corpora (Waseem and Hovy, 2016; Davidson et al., 2017; Wulczyn et al., 2017). Most work has been conducted on English data – tweets in particular – with some extensions to other domains (e.g. hacking forums (Caines et al., 2018))

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 556–563
Minneapolis, Minnesota, USA, June 6–7, 2019.

and other languages (e.g. Arabic (Mubarak et al., 2017), Chinese (Su et al., 2017), Slovene (Fišer et al., 2017)).

Automated detection approaches have drawn on traditional document classification methods for spam detection and sentiment analysis, and tend to use lexical and syntactic features (Nobata et al., 2016; Li et al., 2017; Bourgonje et al., 2018). Machine learning techniques range from logistic regression (Cheng et al., 2015) to support vector machines (Yin et al., 2009) to neural networks (Gambäck and Sikdar, 2017).

We draw on the work by Mishra and colleagues, who used a character-based recurrent neural network to form contextual word representations of out-of-vocabulary words (Mishra et al., 2018b), and moreover employed graph-based author embeddings to represent group behaviour within social networks, significantly improving abuse detection (Mishra et al., 2018a). In this shared task, we do not have access to author information, but instead adapt the approach by building a graph of the tokens which occur in the training data, a method described in further detail in Section 4.4.

3 Data

The OffensEval shared task uses the Offensive Language Identification Dataset (OLID) (Zampieri et al., 2019a), which hierarchically labels tweets according to whether or not they are offensive, whether any offence is targeted, and if so targeted at whom: an individual, a group or otherwise. The three subtasks in this shared task correspond to predicting labels at each level of granularity. The data is structured to allow this: all tweets presented in subtask B are guaranteed to be offensive, and all of those in subtask C are targeted.

Tweets were collected by using the Twitter API to search for terms that are frequently associated with offensive behaviour. These included political keywords, as political content may attract a disproportionate amount of offensive comments. The dataset is evenly split between tweets sourced from these keywords and non-political ones. The authors additionally found that an effective strategy for gathering offensive tweets was to search for those flagged by Twitter's safe search feature. All tweets were anonymised by replacing usernames and URLs with placeholder tokens.

Each of the 14,100 collected tweets were manually annotated by at least two annotators; where the original two annotators disagreed on a tweet, it was further annotated until agreement reached 66%. Table 1 presents the number of tweets in each category.

A	B	C	Train	Test	Total
OFF	TIN	IND	2,407	100	2,507
OFF	TIN	OTH	395	35	430
OFF	TIN	GRP	1,074	78	1,152
OFF	UNT	—	524	27	551
NOT	—	—	8,840	620	9,460
All			13,240	860	14,100

Table 1: Count of tweets in each category of OLID (Zampieri et al., 2019a).

4 Methodology

In this section, we extend the model proposed by Mishra et al. (2018b) for offensive language classification. The architecture uses a 2-layer RNN, optimised using Adam (Kingma and Ba, 2015), to predict the class of a given tweet. The pre-softmax activation values from the output layer are given as input to a GBDT for final classification. Using the GBDT for classification was found to give better results compared with predicting from the RNN directly, and allows us to include additional features into the model. The RNN is initialised with pre-trained word embeddings which are fine-tuned during training. For previously unseen words, we follow Mishra et al. (2018b) in using a neural character-based compositional model to generate plausible embeddings of unseen words. This component is optimised to compose context-aware character embeddings into word-level embeddings that are similar to the pre-trained representations, trained on words for which the embeddings are available. This methodology is effective in generating reasonable quality embeddings in instances where words were deliberately obscured to evade detection.

Following common practice in named entity recognition (Sang and De Meulder, 2003), where fine-grained labels are used to improve performance on the sequence labeling task, we take advantage of the hierarchical labels available for each tweet. For subtasks A and B we train a model to predict all cascading labels, and sum the probabilities of labels under the relevant class to make a final prediction. For example, for subtask A the

model is trained to predict between 5 classes: not-offensive (NOT), offensive but not targeted (UNT), targeted towards an individual (IND), towards a group (GRP), and towards any other target (OTH). We classify a tweet as offensive if the cumulative probability mass for UNT, IND, GRP, and OTH is greater than NOT.

We also introduce several architectural extensions to the Mishra et al. (2018b) model. Firstly, we augment the core RNN with ELMo embeddings and a self-attention mechanism. Secondly, we add both the post-softmax output from the RNN as well as graph-based representations of tweets as input features to the GBDT classifier. We provide details of each extension in the following sections.

For each subtask, we experiment with combinations of the above and additionally tune the RNN type (between LSTM (Hochreiter and Schmidhuber, 1997) and GRU (Cho et al., 2014)), dimension, and batch size, whether to use character n-grams ($n \in [1, 4]$), and, when used, the size of self-attention layers. We also run experiments using the unmodified model to find which pre-trained embeddings give the best performance. We compare publicly available embeddings trained using Word2Vec (Mikolov et al., 2013), FastText (Mikolov et al., 2018), and GLoVe (Pennington et al., 2014).

4.1 ELMo

We use embeddings generated from ELMo concatenated with pre-trained word embeddings as input to the RNN. ELMo generates embeddings on a character level, so does not share the same out-of-vocabulary issue as pre-trained embeddings and is always able to generate a word representation. We used the largest pre-trained model available online[1], and learn a weighted linear combination of its three layers.

4.2 Self-attention

The model proposed by Mishra et al. (2018b) uses the last hidden state of the RNN as the feature representation for each tweet; instead, we propose the use of a self-attention mechanism to learn a weighted combination of all intermediate hidden states (Rei and Søgaard, 2019). The weights $\hat{a}_i$ for each hidden state h_i are learned by passing h_i through two dense layers with $\tanh$ activation, and a further 1-dimensional dense layer. The final dense layer has either sigmoid or exponential activation, corresponding to soft or sharp attention respectively. The weights are normalised to sum to 1, yielding final attention values $\widetilde{a}_i$, which are used to obtain the final sentential representation $s = \sum_i \widetilde{a}_i h_i$. The RNN is then trained using categorical cross-entropy on s passed through a final $\tanh$ layer.

4.3 RNN Prediction

This modification includes the post-softmax output of the RNN as an additional input feature to the decision tree.

4.4 node2vec

We make use of node2vec to learn low-dimensional continuous representations of hashtags used in tweets on the basis of whole-tweet contexts. We first represent every token (including all hashtags) and each tweet as nodes in a graph, with edges formed between tweets and the tokens they contain. node2vec first follows a tunable sampling strategy to perform random walks from each node, generating directed acyclic graphs with a maximum out degree of 1 (i.e. a sequence of nodes). It then applies the SkipGram model (Mikolov et al., 2013) to learn a representation of each node based on its neighbours in the sampled sequences. Specifically, given a graph with nodes V, node2vec maximises the log probability: $\sum_{v \in V} log(P(N_s(v)|v))$, where $N_s(v)$ is the set of neighboring nodes for node v generated from a sampling strategy s.

We train these node2vec representations on two data sets: the OLID training data, and our own scrape of Twitter using `rtweet` (Kearney, 2018). We collect this additional data by searching for each of the 24 hashtags which appear at least 10 times in the training set, with at least 1 in 4 occurrences in tweets labelled offensive. Intuitively, these common and frequently offensive hashtags are a more reliable signal of offensiveness than less frequent hashtags. It remains to be seen whether collecting more tweets with all hashtags in OLID would help, but the strict rate limits on the Twitter API meant that we ran out of time to explore this.

We trained 200-dimensional embeddings on a random sample of $10,000$ of the resulting tweets. To represent each tweet we sum the embeddings of each hashtag present, and normalised the re-

[1] https://allennlp.org/elmo

System	F1 (macro)
Vanilla model	0.710
Vanilla model + ELMo	0.742
Vanilla model + ELMo + self attention	0.764
Vanilla model + ELMo + self attention + char. ngrams	0.763
Vanilla model + ELMo + self attention + node2vec	0.764
Vanilla model + ELMo + self attention + char. ngrams + node2vec	**0.767**

Table 2: Ablation test for features, with results reported on our held-out development set for subtask A.

sulting vector to unit length. These vectors, or a 200-dimensional 0-vector for tweets containing no hashtags with trained embeddings, were then concatenated with the RNN features (either from self-attention where it was used, or the last hidden state if not) prior to being input into the GBDT.

5 Results

In this section, we present a sample of results obtained during model selection, and results on each of the official subtask test sets. Model selection is carried out by evaluating each model on a consistent 90% training and 10% validation split of the provided training data. Before carrying out model selection, we ran an unmodified version of Mishra et al. (2018b)'s model on subtask A and found that 300-dimensional FastText embeddings trained on Common Crawl gave the best performance[2].

We submitted three models for each of the three subtasks. We submit models that differ in two ways. The first is the amount of data they are trained on. Models labelled ALL-DATA are trained on all of the provided data, while models tagged TRAIN-SPLIT are trained on just the 90% training split, but have a known performance via their results on the development set. It is beneficial to know this as there is a large amount of variance in model results due to stochasticity in the training process. The second way in which the models differ is designed to handle this variance by ensembling three models via majority vote. Such submissions are labelled with ENSEMBLE, while those only using a single model are labelled BEST.

In all three subtasks we find that the best performing system is that which ensembles three identical models trained on the entire training set.

5.1 Subtask A

Subtask A concerns classifying a tweet as OFF (offensive) or NOT (see Section 3). We experiment with adaptations of the model from Mishra et al. (2018b) to perform a 5-WAY classification between all categories, and select the most effective feature combination for each subtask. We experiment with features mentioned in Section 4: ELMo, self-attention, character n-grams, and node2vec. Results from ablation studies are presented in Table 2.

We found that the best performing model used features extracted from a RNN that used ELMo embeddings in addition to FastText and compositional character-based word embeddings, with sharp self-attention over a GRU with 256 hidden units trained using a batch size of 64. These features were used in a GBDT alongside the 10,000 most frequently occurring character n-grams, and node2vec representations of the tweets.

Table 3 shows our results for subtask A on the test data. All three submissions use the model architecture and hyperparameters described above.

System	F1	Accuracy
All NOT baseline	0.419	0.721
All OFF baseline	0.218	0.279
TRAIN-SPLIT-BEST	0.776	0.835
ALL-DATA-ENSEMBLE	**0.788**	**0.847**
ALL-DATA-BEST	0.769	0.835

Table 3: Accuracy and macro F1 results on the official subtask A test set. All three models have the same hyperparameters.

5.2 Subtask B

Subtask B involves a binary classification of whether a tweet is untargeted (UNT) or targeted (TIN). Following Subtask A, we maintain a finer grained classifier using a 4-WAY classification (TIN, IND, GRP, OTH), where we classify a tweet as targeted if the probability for TIN is less than the sum of probabilities for the 3 other labels.

We re-ran feature selection experiments to op-

[2]https://fasttext.cc/docs/en/english-vectors.html

System	F1	Accuracy
All TIN baseline	0.470	0.888
All UNT baseline	0.101	0.113
TRAIN-SPLIT-BEST	0.577	0.717
ALL-DATA-ENSEMBLE	**0.663**	**0.904**
TRAIN-SPLIT-ENSEMBLE	0.657	0.900

Table 4: Accuracy and macro F1 results on the official subtask B test set.

timise for this task. Development experiments showed that the use of character n-grams does not improve performance on this subtask, LSTM performs better than a GRU, and that reducing the RNN dimension to 64 and training batch size to 32 is beneficial. These smaller hyperparameter values are likely more suitable due to the smaller amount of available training data. We found that training node2vec using the provided training data, rather than the scraped dataset, gave better representations, with F1 scores on our held-out development set of 0.635 for OLID data and 0.618 for the extra tweets we obtained from Twitter's API (section 4.4).

Results on the test set are presented in Table 4, where we once again find that the ensemble of classifiers trained on all of the data performs best.

5.3 Subtask C

Subtask C involves classifying the target of an offensive tweet as either an individual, group, or other. As this is the last subtask, only classification between these three labels is possible: there are no finer-grained labels that can be trained on. We find that the best performing model is the same as that in subtask B, except that a GRU is used and the softmax from the RNN is included in the GBDT. Results on the test data are presented in Table 5.

System	F1	Accuracy
All GRP baseline	0.179	0.366
All IND baseline	0.213	0.470
All OTH baseline	0.094	0.164
ALL-DATA-ENSEMBLE	**0.554**	0.704
TRAIN-SPLIT-ENSEMBLE	0.544	**0.709**
TRAIN-SPLIT-BEST	0.534	0.695

Table 5: Accuracy and macro F1 results on the official subtask C test set.

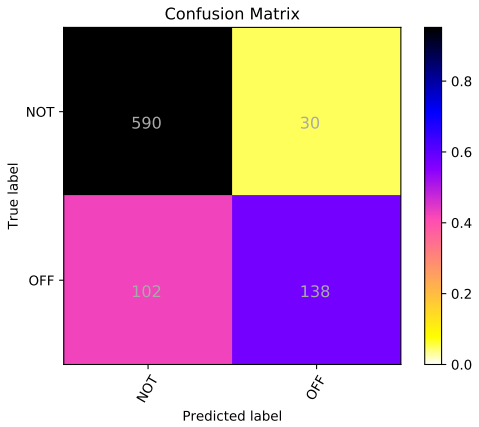

Figure 1: Subtask A, ALL-DATA-ENSEMBLE model.

6 Discussion

In all subtasks, our best performing submission was an ensemble of three identical models, independently trained on all of the training data. Ensembling helps to account for the high variance observed during model training, which occurred despite fixing random seeds.

Across all subtasks we find the inclusion of node2vec features to be helpful. These features offer contextualised representations of hashtags in terms of the tokens they appear with across the corpus, suggesting that features that share information between tweets are useful in addition to those derived from each individually.

We observe that performance drops from subtask A to C. This could be due to the decreasing amounts of training data, from 13,240 instances in Subtask A, to 4,400 in subtask B and 3,876 in subtask C. Very small amounts of data are available for two classes in particular – untargeted offence (UNT) with only 524 training instances, and offence targeted at those other than individuals and groups (OTH) with 395.

As seen in Figures 1 and 2, our model achieves high recall for the NOT class (0.952) in subtask A and for TIN (0.986) in subtask B, but low recall for the other classes OFF (0.575) and UNT (0.259). Figure 3 shows that in subtask C we perform worst on the OTH label, with a low recall of 0.086. In all cases, the model shows weakest performance on the classes for which we have least training data. Therefore, we expect that model performance would improve given more training instances of the minority classes.

Furthermore, in subtask C, the definition of the 'other' class is less clear-cut than the other two cat-

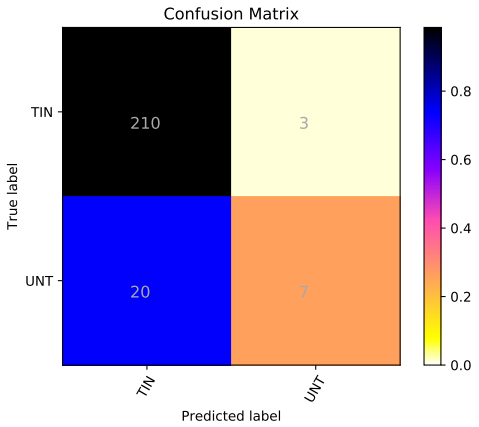

Figure 2: Subtask B, ALL-DATA-ENSEMBLE model.

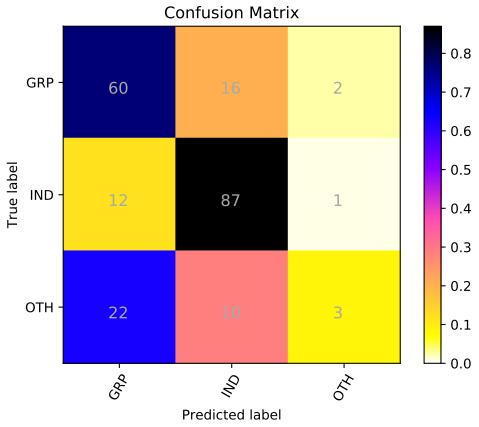

Figure 3: Subtask C, ALL-DATA-ENSEMBLE model.

egories GRP and IND, including abstract concepts such as events or issues, and serving as a catch-all for targeted insults against anything other than specific people or groups of people with a common characteristic. A manual inspection of the data suggests that a large amount of the OTH data includes politically-motivated insults, though similar language also appears in the two other categories, which may make classification harder.

7 Conclusion

The CAMsterdam team attempted the OffensEval tasks taking inspiration from the approach of Mishra and colleagues (2018b), feeding the pre-softmax activation layer of an RNN into a GBDT to classify tweets into one of the applicable fine-grained classes for each subtask. The probabilities of the fine-grained classes were summed to obtain a probability for the desired class: for instance, in subtask A, we summed the probabilities of UNT, IND, GRP and OTH, and compared this sum with the probability of NOT to classify a tweet as offensive or not.

We extended the work of Mishra et al. by using ELMo embeddings as additional input to the RNN, and incorporating a self-attention mechanism following Rei and Søgaard (2019). We also used node2vec to train graph-based representations of hashtags, using both tweets from the OLID training set and new data obtained from the Twitter API featuring hashtags frequently found in the offensive subset. We focus on hashtags on the intuition that they are employed by users to reach those interested in similar topics, and are thus indicative of tweet content. Their use encodes this useful information directly, which we show to be useful for classification. We take into account the fact that hashtags are used in many positions in a tweet by constructing the graph based on co-occurrence across the whole tweet, rather than only within a small window as other embedding methods do.

During development, we found that our best performing models were those formed from an ensemble of three models trained in an identical fashion, thereby smoothing random variation in the training process. The results of the test phase show that our model performed in line with expectations set during development, with F1-scores which decrease from subtask A to C, and lowest precision and recall on the minority classes.

In the future, we will seek to address the imbalance in the training data, inspect the tweets further to analyse the linguistic differences between targeted and untargeted insults, group- and individual-targeted insults and so on. Further architectural changes include collecting more instances of hashtags frequently found in offensive tweets as extra unsupervised data, and we can seek to include author embeddings, a technique found to greatly improve the performance of Mishra et al's system (Mishra et al., 2018a). Finally, we would aim to evaluate our model on other offensive text classification datasets, to discover how well the design generalizes beyond OLID.

Acknowledgements

The 2nd author is supported by the EPSRC, U.K. The 4th, 5th, 6th and 8th authors are members of the ALTA Institute, supported by Cambridge Assessment, University of Cambridge. We thank the NVIDIA Corporation for the donation of the Titan GPU used in this research.

References

Peter Bourgonje, Julian Moreno-Schneider, Ankit Srivastava, and Georg Rehm. 2018. Automatic classification of abusive language and personal attacks in various forms of online communication. In *Language Technologies for the Challenges of the Digital Age*. Springer International Publishing.

Andrew Caines, Sergio Pastrana, Alice Hutchings, and Paula Buttery. 2018. Aggressive language in an online hacking forum. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*. Association for Computational Linguistics.

Justin Cheng, Michael Bernstein, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. 2017. Anyone can become a troll: Causes of trolling behavior in online discussions. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*.

Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. 2015. Antisocial behavior in online discussion communities. In *The 9th International AAAI Conference on Web and Social Media (ICWSM)*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN EncoderDecoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

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

Karthik Dinakar, Roi Reichart, and Henry Lieberman. 2011. Modeling the detection of textual cyberbullying. In *Fifth International AAAI Conference on Weblogs and Social Media*.

Darja Fišer, Tomaž Erjavec, and Nikola Ljubešić. 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*.

Björn Gambäck and Utpal Kumar Sikdar. 2017. Using convolutional neural networks to classify hate-speech. In *Proceedings of the First Workshop on Abusive Language Online*.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Sepp Hochreiter and Jrgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3146–3154. Curran Associates, Inc.

Michael W. Kearney. 2018. *rtweet: Collecting Twitter Data*. R package version 0.6.7.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: a Method for Stochastic Optimization. In *International Conference on Learning Representations*, pages 1–13.

Irene Kwok and Yuzhou Wang. 2013. Locate the hate: Detecting tweets against blacks. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*.

Tai Ching Li, Joobin Gharibshah, Evangelos E. Papalexakis, and Michalis Faloutsos. 2017. TrollSpot: Detecting misbehavior in commenting platforms. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*. 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)*.

Pushkar Mishra, Marco Del Tredici, Helen Yannakoudakis, and Ekaterina Shutova. 2018a. Author profiling for abuse detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Pushkar Mishra, Helen Yannakoudakis, and Ekaterina Shutova. 2018b. Neural character-based composition models for abuse detection. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, Brussels, Belgium. Association for Computational Linguistics.

Hamdy Mubarak, Kareem Darwish, and Walid Magdy. 2017. Abusive language detection on Arabic social media. In *Proceedings of the First Workshop on Abusive Language Online*.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*.

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.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Marek Rei and Anders Søgaard. 2019. Jointly Learning to Label Sentences and Tokens. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019)*, Honolulu, USA.

Manoel Horta Ribeiro, Pedro H. Calais, Yuri A. Santos, and Wagner Meira J Virgílio A. F. Almeid and. 2018. "Like sheep among wolves": Characterizing hateful users on Twitter. In *Proceedings of WSDM workshop on Misinformation and Misbehavior Mining on the Web (MIS2)*.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Sara Owsley Sood, Elizabeth F. Churchill, and Judd Antin. 2012. Automatic identification of personal insults on social news sites. *Journal of the American Society for Information Science and Technology*, 63:270–285.

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.

William Warner and Julia Hirschberg. 2012. Detecting hate speech on the World Wide Web. In *Proceedings of the Second Workshop on Language in Social Media*.

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 Langauge 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*.

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*.

Jun-Ming Xu, Benjamin Burchfiel, Xiaojin Zhu, and Amy Bellmore. 2013. An examination of regret in bullying tweets. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Dawei Yin, Zhenzhen Xue, Liangjie Hong, Brian D. Davison, April Kontostathis, and Lynne Edwards. 2009. Detection of harassment on Web 2.0. In *Proceedings of the Content Analysis in the WEB 2.0 (CAW2.0) Workshop at WWW2009*.

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)*.