Toxic Comment Classification

Usama Zafar
Student ID: 2019380101
Dept. Computer Science and Technology
Tsinghua University
Beijing, China
sham19@mails.tsinghua.edu.cn

Abdalwhb Abdalwhab
Student ID: 2019380044
Dept. Computer Science and Technology
Tsinghua University
Beijing, China
Abdalwhab.Bakheet@gmail.com

Abstract

Flood of information is produced on a daily basis through the global internet usage arising from the online interactive communications among users. While this situation contributed significantly to quality of human life, unfortunately it involves enormous dangers, since online texts with high toxicity can cause personal attacks, harassment and bullying. This has triggered both industrial and academic community. Toxic comment classification has become an active research field with many recently proposed approaches. However, while these methods address some of the task's challenges other still remain unsolved and directions for further research are need. One such challenge being multilingual toxic comments. Classifying multilingual data is challenging and difficult. Not enough work has been carried out in this domain even though our current internet is a multilingual, multicultural and global place. In this work, we employ multiple models, such as pre-trained BERT, Distilled BERT, Bi-LSTM models to identify toxic comments among text data provided by Kaggle's "Jigsaw Multilingual Toxic Comment Classification Challenge". Finally, we create an ensemble of multiple well performing models and present our results. The reported accuracy of the toxic comment identification by our ensemble is certainly promising and provides a good insight into what to expect when dealing with multilingual data.

1. Introduction

Keeping online conversations constructive and inclusive is a crucial task for platform providers. It only takes one toxic comment to sour an entire conversation. Automatic classification of toxic comments, such as hate speech, threats, and insults, can help in keeping discussions fruitful. In addition, new regulations in certain European countries have been established enforcing to delete illegal content in less than 72 hours.[1]

Active research on the topic deals with common challenges of natural language processing, such as long-range dependencies or misspelled and idiosyncratic words. Proposed solutions include bidirectional recurrent neural networks with attention [23] and the use of pretrained word embedding [3]. However, many classifiers suffer from insufficient variance in methods and training data and therefore often tend to fail on the long tail of real world data [37]. For future research, it is essential to know which challenges are already addressed by state-of-the-art classifiers and what current solutions are still error-prone.

This project aims at tackling the Jigsaw Multilingual Toxic Comment Classification [1]. Where a toxic comment is defined as anything rude, disrespectful or otherwise likely to make someone leave a discussion. If these toxic contributions can be identified, then we can plan and execute polices like removing them or notifying their corresponding users. Eventually, this will help us taking a step towards having a safer, more collaborative internet.

2. Related Work

In this section we present some related work published in the past. Although it is not an easy job to tell what is toxic comment classification and what is not since, most NLP publications can be modified one way or the other to solve this particular problem.

Task definitions. Toxic comment classification is not clearly distinguishable from its related tasks. Besides looking at toxicity of online comments [34]; [12], related research includes the investigation of hate speech [3]; [4]; [7]; [10]; [13]; [28]; [30]; [31], online harassment [36]; [14], abusive language [20]; [22], cyberbullying [6]; [9]; [16]; [38] and offensive language [5]; [35]. Each field uses

[1] https://www.bbc.com/news/technology-42510868

different definitions for their classification, still similar methods can often be applied to different tasks. In our work we focus on toxic comment detection and show that the same method can effectively be applied to a hate speech detection task.

Multi-class approaches. Besides traditional binary classification tasks, related work considers different aspects of toxic language, such as "racism" [15]; [32]; [19] and "sexism" [33]; [17], or the severity of toxicity [7]; [29]. These tasks are framed as multi-class problems, where each sample is labeled with exactly one class out of a set of multiple classes. The great majority of related research considers only multi-class problems. This is remarkable, considering that in real-world scenarios toxic comment classification can often be seen as a multi-label problem, with user comments fulfilling different predefined criteria at the same time. We therefore investigate both a multi-label dataset containing six different forms of toxic language and a multi-class dataset containing three mutually exclusive classes of toxic language.

Shallow classification and neural networks. Toxic comment identification is a supervised classification task and approached by either methods including manual feature engineering [4]; [20];[32]; [7]; [21]; [18]; [27]; [26] or the use of (deep) neural networks [24]; [23]; [3]; [30]; [22]; [10]. While in the first case manually selected features are combined into input vectors and directly used for classification, neural network approaches are supposed to automatically learn abstract features above these input features. Neural network approaches appear to be more effective for learning [4], while feature-based approaches preserve some sort of explainability. We focus in this paper on baselines using deep neural networks (e.g. CNN and Bi-LSTM) and shallow learners, such as Logistic Regression approaches on word n-grams and character n-grams.

Ensemble learning. [4] studied advantages of ensembles of different classifiers. They combined results from three feature-based classifiers. Further the combination of results from Logistic Regression and a Neural Network has been studied [11]; [25]. Zimmerman et al. [39] investigated ensembling models with different hyper-parameters. We use a similar approach with a very powerful pretrained model BERT and its variations thereof.

3. Competition Description

It only takes one toxic comment to sour an online discussion. The Conversation AI team, a research initiative founded by Jigsaw and Google, builds technology to protect voices in conversation. A main area of focus is machine learning models that can identify toxicity in online conversations, where toxicity is defined as anything rude, disrespectful or otherwise likely to make someone leave a discussion. If these toxic contributions can be identified, we could have a safer, more collaborative internet.

In the previous 2018 Toxic Comment Classification Challenge, Kagglers built multi-headed models to recognize toxicity and several subtypes of toxicity. In 2019, in the Unintended Bias in Toxicity Classification Challenge, you worked to build toxicity models that operate fairly across a diverse range of conversations.

Jigsaw's API, Perspective, serves toxicity models and others in a growing set of languages (see our documentation for the full list). Over the past year, the field has seen impressive multilingual capabilities from the latest model innovations, including few- and zero-shot learning. We're excited to learn whether these results ”translate” (pun intended!) to toxicity classification. Your training data will be the English data provided for our previous two competitions and your test data will be Wikipedia talk page comments in several different languages.

As our computing resources and modeling capabilities grow, so does our potential to support healthy conversations across the globe. Develop strategies to build effective multilingual models and you'll help Conversation AI and the entire industry realize that potential.

4. Proposed Methods

Figure 1 shows our BERT-based[8] baseline model, we started by a specialized pre-trained multilingual model called m-BERT. We used the pretrained weights to initialize both the tokenizer and the model.

For the pre-processing, we pad all sentences to the same length and add a [CLS] token at the beginning (a special token used as a placeholder to get a vector embedding representing the whole sentence). Then attention mask for each sentence is generated to clarify which tokens represents real words and which are just padded junk. Then every sentence is passed through a 12 transformers layers each has a size of 768, and with 12 attention heads. Each layer (including the last layer) produces a vector embedding for each word, and another vector embedding representing the whole sentence and pass it to the next layer. In this implementations, we neglect all the words embedding and only use the 768 dimensional vector representing the sentence and use it for classification. We started with simple 2 Fully Connected layers (FCs) for classification. First one, use ReLU activation and the other uses Sigmoid function.

Although BERT is the best possible model known to us, the performance can still be improved by data pre-processing. So we decided to split the work along two branches, exploring better models, and improving pre-processing of available data while using BERT or BERT

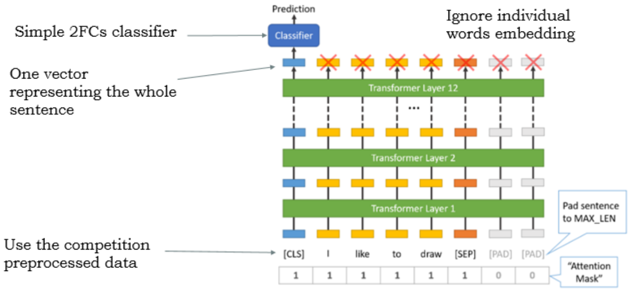

Figure 1. BERT-based Baseline Model.

like architecture. We tried bunch of other models such as:

- Bidirectional LSTMs with pretrained glove model for embedding
- Using the BERT model itself we also tried bunch of other things such as:
- Trying a more compact version of BERT (6 layers instead of 12)
- Use different pretrained BERT models
- Preprocess the data ourselves instead of using the competition pre-processed data
- Ensemble multiple predictions

Furthermore, to improve the obtained performance in the competition, we apply simple ensemble technique by taking the weighted average of our top models predictions. Figure 2 introduce the diagram for the ensemble technique.

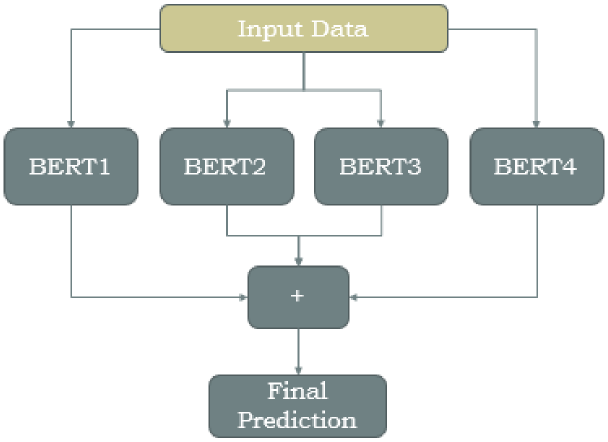

Figure 2. Ensemble couple of models.

5. Experiments

5.1. Dataset

We based our work on the competition dataset. The primary data for the competition consists of comments classified as toxic or non-toxic (0 and 1). The dataset is divided into train, validation and test splits. The train set's comments are entirely in english and come either from Civil Comments or Wikipedia talk page edits. The validation and test data's comments are composed of multiple non-English languages[1]. They also, include the data from their other competition "Jigsaw Unintended Bias in Toxicity Classification"[2] as an auxiliary dataset, but the dataset has 6 classifications for comments labels instead of just two.

The organizers of the competition provided the dataset in two format; raw text files, and pre-processed data ready for use by BERT models, but they used a small maximum sequence length of only 128 tokens.

5.2. Evaluation

To evaluate a model, depending on the experiment, we may train it on the primary training dataset, auxiliary dataset and validation dataset, but we never let the model train on the test dataset. In fact, the labels for test dataset of the competition are not provided, so the only way to get the score of your model is by submitting the predictions to the competition website. So, that is what we did.

Submissions to the competition are evaluated on area under the receiver operating characteristic (ROC) curve (AUC) between the predicted probability and the observed target.

5.3. Baseline m-BERT

For the first experiment, we tried to train our baseline model (the m-BERT-based model) for 5 epochs on the main training dataset and evaluate it on the text dataset.

After this baseline experiment, we decided to evaluate a more compact version of BERT called DistilBERT with special pretrained weights on multilingual data called distilbert-base-multilingual-cased. This model has only 6 hidden layers instead o the 12 layers by m-BERT.

We only used one FC layer with one neuron and sigmoid activation function to do the classification. With the memory limitations on Kaggle environment we adopted this smaller model to be able to add on top of it our own ideas.

Then we investigated improving this model in two directions, one that utilizes the auxiliary dataset for training with early stopping, while the second one only uses the primary dataset.

5.4. Utilizing The Auxiliary Dataset

For this experiment, we tried to utilize the auxiliary dataset by converting its labels to be in the range 0,1 by applying a threshold of 0.5 (the recommended threshold from [2] for mapping their labels to toxic and non-toxic). Then we merged this data with the primarily train dataset process it by ourselves with a maximum sequence length of 100 tokens.

We then trained the model for 10 epochs on the combined training data, and 20 epochs on the validation data while early stopping utilized to stop training whenever the performance on the validation data is not improved after one epoch.

5.5. No Auxiliary Dataset

Without utilizing the auxiliary dataset, with maximum sequence length of 192. Training on the primary dataset for 3 epochs and on the validation dataset for another 3 epochs.

5.6. Improved Classifier

Based on previous experiment in this experiment, we increased the maximum sequence length to 120 token (which is still less than the maximum sequence length used in Ep.2.B because processing of the auxiliary dataset requires a lot of memory), and improved the classifier by adding two layers before the final classification layer. The first one, is FC layer with 256 neuron, and ReLU activation function. While the second is a dropout layer with dropout probability of 0.2.

5.7. Train on Primary and Half The Auxiliary Datasets Separately

In this experiment we tried to utilize the available memory more efficiently by explicitly releasing any unneeded object. This allowed us to process longer sequences without truncating it. In fact, we changed the maximum sequence length to 250.

We started, by training on the primary training data for 5 epochs, then process around half of the auxiliary data and train on it for 4 epochs. Lastly, the model is trained on the validation data for 3 epochs before it is used for evaluation.

5.8. Train on Primary and Auxiliary Datasets Separately

We increased the maximum sequence length to 250 and same setup as Exp.4.A, but after training on half of the auxiliary dataset, we process and train on the other half before training on validation dataset.

5.9. Ensemble Multiple Predictions

In this experiment, we take the average of predictions from our top performing models that were trained on both primary and secondary datasets.

5.10. Other Failed Experiments

We tried to implement a Bidirectional Long-Short-Term Memory (BiLSTM) based model with pretrained glove model for embedding, but the experiment failed couple of times. Some of the failures were due to the law speed internet connection and the fact that we have to use VPN to be able to connect to Kaggle website and run our experiments. In addition, the time limit for the project did not allow us to investigate in all the directions we want, so we focused on BERT-based models.

Implementation	ROC Score
Baseline m-BERT	0.8236
Utilize the auxiliary dataset	0.8613
No auxiliary dataset	0.8653
Improved classifier	0.8782
Primary/$\frac{1}{2}$auxiliary separately	0.8835
Primary/auxiliary separately	0.8668
Ensemble multiple predictions	0.8851
Pre-Trained Transformer [HuggingFace] v2	0.8782
Pre-Trained Transformer [HuggingFace] v3	0.8835
Ensemble [m-BERT + Pre-trained] v1	0.9269
Ensemble [m-BERT + Pre-trained] v2	0.9315
Ensemble [m-BERT + Pre-trained] v3	0.9318

Table 1. Brief summary of our submissions

Furthermore, we tried to use the Off-shelf BertForSequenceClassification to tackle the classification problem but failed for the same above reasons.

6. Results

A brief summary of our submissions is presented in Table1. As expected we managed to improve the performance as we improve the model and pre-processing. Except for Exp.5.A where training on the whole auxiliary dataset did not improve the performance, which we think is because the model starts to overfit on it, while it is too different from the training dataset.

Despite the fact that, we are currently ranked as 944 out of 1436 teams, our performance is not very far from the current top team (0.9318 compared to 0.9556).

7. Conclusion

In this work, multiple BERT-based model were successfully implemented to identify toxic comments among text data provided by Kaggle's "Jigsaw Multilingual Toxic Comment Classification Challenge". Finally, we create an ensemble of multiple well performing models and present our results. The reported accuracy of the toxic comment identification by our ensemble is certainly promising and provides a good insight into what to expect when dealing with multilingual data.

For future work, we may try to use the individual words embedding generated by BERT, and possibly pass them through a couple of Bi-LSTM layers. Furthermore, we may incorporate position embedding to enhance the performance.

References

[1] Jigsaw multilingual toxic comment classification. https://www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification. Accessed: 2020-06-13.

[2] Jigsaw unintended bias in toxicity classification. https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification/data. Accessed: 2020-06-13.

[3] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma. Deep learning for hate speech detection in tweets. volume abs/1706.00188, 2017.

[4] P. Burnap and M. L. Williams. Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. volume 7, pages 223–242, 2015.

[5] Y. Chen, Y. Zhou, S. Zhu, and H. Xu. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust, PASSAT 2012, and 2012 International Confernece on Social Computing, SocialCom 2012, Amsterdam, Netherlands, September 3-5, 2012*, pages 71–80. IEEE Computer Society, 2012.

[6] M. Dadvar, D. Trieschnigg, R. Ordelman, and F. de Jong. Improving cyberbullying detection with user context. In P. Serdyukov, P. Braslavski, S. O. Kuznetsov, J. Kamps, S. M. Rüger, E. Agichtein, I. Segalovich, and E. Yilmaz, editors, *Advances in Information Retrieval - 35th European Conference on IR Research, ECIR 2013, Moscow, Russia, March 24-27, 2013. Proceedings*, volume 7814 of *Lecture Notes in Computer Science*, pages 693–696. Springer, 2013.

[7] T. Davidson, D. Warmsley, M. W. Macy, and I. Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017, Montréal, Québec, Canada, May 15-18, 2017*, pages 512–515. AAAI Press, 2017.

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

[9] K. Dinakar, R. W. Picard, and H. Lieberman. Common sense reasoning for detection, prevention, and mitigation of cyberbullying (extended abstract). In Q. Yang and M. J. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 4168–4172. AAAI Press, 2015.

[10] B. Gambäck and U. K. Sikdar. Using convolutional neural networks to classify hate-speech. In *Proceedings of the First Workshop on Abusive Language Online*, pages 85–90, Vancouver, BC, Canada, Aug. 2017. Association for Computational Linguistics.

[11] L. Gao and R. Huang. Detecting online hate speech using context aware models. In R. Mitkov and G. Angelova, editors, *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, Varna, Bulgaria, September 2 - 8, 2017*, pages 260–266. INCOMA Ltd., 2017.

[12] S. V. Georgakopoulos, S. K. Tasoulis, A. G. Vrahatis, and V. P. Plagianakos. Convolutional neural networks for toxic comment classification. In *Proceedings of the 10th Hellenic Conference on Artificial Intelligence, SETN 2018, Patras, Greece, July 09-12, 2018*, pages 35:1–35:6. ACM, 2018.

[13] N. D. Gitari, Z. Zu-ping, Z. Zhang, H. Damien, and J. Long. A lexicon-based approach for hate speech detection. In *MUE 2015*, 2015.

[14] J. Golbeck, Z. Ashktorab, R. O. Banjo, A. Berlinger, S. Bhagwan, C. Buntain, P. Cheakalos, A. A. Geller, Q. Gergory, R. K. Gnanasekaran, R. R. Gunasekaran, K. M. Hoffman, J. Hottle, V. Jienjitlert, S. Khare, R. Lau, M. J. Martindale, S. Naik, H. L. Nixon, P. Ramachandran, K. M. Rogers, L. Rogers, M. S. Sarin, G. Shahane, J. Thanki, P. Vengataraman, Z. Wan, and D. M. Wu. A large labeled corpus for online harassment research. In P. Fox, D. L. McGuinness, L. Poirier, P. Boldi, and K. Kinder-Kurlanda, editors, *Proceedings of the 2017 ACM on Web Science Conference, WebSci 2017, Troy, NY, USA, June 25 - 28, 2017*, pages 229–233. ACM, 2017.

[15] E. Greevy and A. F. Smeaton. Classifying racist texts using a support vector machine. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, page 468–469, New York, NY, USA, 2004. Association for Computing Machinery.

[16] C. V. Hee, E. Lefever, B. Verhoeven, J. Mennes, B. Desmet, G. D. Pauw, W. Daelemans, and V. Hoste. Detection and fine-grained classification of cyberbullying events. In G. Angelova, K. Bontcheva, and R. Mitkov, editors, *Recent Advances in Natural Language Processing, RANLP 2015, 7-9 September, 2015, Hissar, Bulgaria*, pages 672–680. RANLP 2015 Organising Committee / ACL, 2015.

[17] A. Jha and R. Mamidi. When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 7–16, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics.

[18] G. Kennedy, A. McCollough, E. Dixon, A. Bastidas, J. Ryan, C. Loo, and S. Sahay. Technology solutions to combat online harassment. In *Proceedings of the First Workshop on Abusive Language Online*, pages 73–77, Vancouver, BC, Canada, Aug. 2017. Association for Computational Linguistics.

[19] I. Kwok and Y. Wang. Locate the hate: Detecting tweets against blacks. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, AAAI'13, page 1621–1622. AAAI Press, 2013.

[20] Y. Mehdad and J. R. Tetreault. Do characters abuse more than words? In *Proceedings of the SIGDIAL 2016 Conference, The 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 13-15 September 2016, Los Angeles, CA, USA*, pages 299–303. The Association for Computer Linguistics, 2016.

[21] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 145–153, Republic and

Canton of Geneva, CHE, 2016. International World Wide Web Conferences Steering Committee.

[22] J. H. Park and P. Fung. One-step and two-step classification for abusive language detection on twitter. In Z. Waseem, W. H. K. Chung, D. Hovy, and J. R. Tetreault, editors, *Proceedings of the First Workshop on Abusive Language Online, ALW@ACL 2017, Vancouver, BC, Canada, August 4, 2017*, pages 41–45. Association for Computational Linguistics, 2017.

[23] J. Pavlopoulos, P. Malakasiotis, and I. Androutsopoulos. Deeper attention to abusive user content moderation. pages 1125–1135, 01 2017.

[24] M. Ptaszynski, J. K. K. Eronen, and F. Masui. Learning deep on cyberbullying is always better than brute force. In R. Rzepka, J. Vallverdú, and A. Wlodarczyk, editors, *Proceedings of the Linguistic and Cognitive Approaches To Dialog Agents Workshop co-located with the 26th International Joint Conference on Artificial Intelligence, LaCATODA@IJCAI 2017, Melbourne, Australia, August 21, 2017*, volume 1926 of *CEUR Workshop Proceedings*, pages 3–10. CEUR-WS.org, 2017.

[25] J. Risch and R. Krestel. Aggression identification using deep learning and data augmentation. In R. Kumar, A. K. Ojha, M. Zampieri, and S. Malmasi, editors, *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying, TRAC@COLING 2018, Santa Fe, New Mexico, USA, August 25, 2018*, pages 150–158. Association for Computational Linguistics, 2018.

[26] D. Robinson, Z. Zhang, and J. Tepper. Hate speech detection on twitter: Feature engineering v.s. feature selection. In A. Gangemi, A. L. Gentile, A. G. Nuzzolese, S. Rudolph, M. Maleshkova, H. Paulheim, J. Z. Pan, and M. Alam, editors, *The Semantic Web: ESWC 2018 Satellite Events*, pages 46–49, Cham, 2018. Springer International Publishing.

[27] N. Safi Samghabadi, S. Maharjan, A. Sprague, R. Diaz-Sprague, and T. Solorio. Detecting nastiness in social media. In *Proceedings of the First Workshop on Abusive Language Online*, pages 63–72, Vancouver, BC, Canada, Aug. 2017. Association for Computational Linguistics.

[28] A. Schmidt and M. Wiegand. A survey on hate speech detection using natural language processing. In L. Ku and C. Li, editors, *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, SocialNLP@EACL 2017, Valencia, Spain, April 3, 2017*, pages 1–10. Association for Computational Linguistics, 2017.

[29] S. Sharma, S. Agrawal, and M. Shrivastava. Degree based classification of harmful speech using twitter data. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 106–112, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics.

[30] F. D. Vigna, A. Cimino, F. Dell'Orletta, M. Petrocchi, and M. Tesconi. Hate me, hate me not: Hate speech detection on facebook. In A. Armando, R. Baldoni, and R. Focardi, editors, *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17), Venice, Italy, January 17-20, 2017*, volume 1816 of *CEUR Workshop Proceedings*, pages 86–95. CEUR-WS.org, 2017.

[31] W. Warner and J. Hirschberg. Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26, Montréal, Canada, June 2012. Association for Computational Linguistics.

[32] Z. Waseem. Are you a racist or am I seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[33] Z. Waseem and D. Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June 2016. Association for Computational Linguistics.

[34] E. Wulczyn, N. Thain, and L. Dixon. Ex machina: Personal attacks seen at scale. volume abs/1610.08914, 2016.

[35] G. Xiang, B. Fan, L. Wang, J. I. Hong, and C. P. Rosé. Detecting offensive tweets via topical feature discovery over a large scale twitter corpus. In X. Chen, G. Lebanon, H. Wang, and M. J. Zaki, editors, *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 1980–1984. ACM, 2012.

[36] D. Yin, Z. Xue, L. Hong, B. Davison, A. Edwards, and L. Edwards. Detection of harassment on web 2.0. 01 2009.

[37] Z. Zhang and L. Luo. Hate speech detection: A solved problem? the challenging case of long tail on twitter. volume abs/1803.03662, 2018.

[38] H. Zhong, H. Li, A. C. Squicciarini, S. M. Rajtmajer, C. Griffin, D. J. Miller, and C. Caragea. Content-driven detection of cyberbullying on the instagram social network. In S. Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 3952–3958. IJCAI/AAAI Press, 2016.

[39] S. Zimmerman, U. Kruschwitz, and C. Fox. Improving hate speech detection with deep learning ensembles. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, and T. Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA), 2018.