Exploring Conditional Language Model Based Data Augmentation Approaches For Hate Speech Classification

Ashwin Geet d'Sa, Irina Illina, Dominique Fohr, Dietrich Klakow, Dana Ruiter

Exploring Conditional Language Model Based Data Augmentation Approaches For Hate Speech Classification

Ashwin Geet D'Sa[1], Irina Illina[1], Dominique Fohr[1],
Dietrich Klakow[2], and Dana Ruiter[2]

[1] Université de Lorraine, CNRS, Inria, LORIA
{ashwin-geet.dsa, irina.illina, dominique.fohr}@loria.fr
[2] Spoken Language System Group, Saarland University
{druiter, dietrich.klakow}@lsv.uni-saarland.de

Abstract. Deep Neural Network (DNN) based classifiers have gained increased attention in hate speech classification. However, the performance of DNN classifiers increases with quantity of available training data and in reality, hate speech datasets consist of only a small amount of labeled data. To counter this, Data Augmentation (DA) techniques are often used to increase the number of labeled samples and therefore, improve the classifier's performance. In this article, we explore augmentation of training samples using a conditional language model. Our approach uses a single class conditioned Generative Pre-Trained Transformer-2 (GPT-2) language model for DA, avoiding the need for multiple class specific GPT-2 models. We study the effect of increasing the quantity of the augmented data and show that adding a few hundred samples significantly improves the classifier's performance. Furthermore, we evaluate the effect of filtering the generated data used for DA. Our approach demonstrates up to 7.3% and up to 25.0% of relative improvements in macro-averaged F1 on two widely used hate speech corpora.

Keywords: Natural language processing · Hate speech classification · Data augmentation.

1 Introduction

Increased usage of social media has led to a rise in online hate speech. Hate speech is an anti-social behavior, against a social group based on ethnicity, nationality, religion, gender, etc. [7]. It induces a feeling of threat, violence, and fear to the targeted group or individual. Manual tagging of such comments on social media is time-consuming and very expensive. Hence, Natural Language Processing (NLP) and classification techniques can help moderators identify hate speech.

The research interest towards hate speech classification has increased [3, 17, 5, 14]. The performance of the commonly used neural network classifiers depends on the amount of training data that is available, and unfortunately most of the hate speech datasets have only a small amount of labeled data to train the classifier.

Various Data Augmentation (DA) approaches have been explored in literature to train better performing text classification or representation models. One group of approaches includes replication of samples by performing minor modifications such as addition, deletion, swapping of words, and synonym replacement [24]. Some approaches in this group replicate samples through word replacements based on embeddings of the word and its surrounding context [23, 16, 26]. Other group of approaches have explored translation and back-translation [20, 22], auto-regressive language models [1], and auto-encoders [13].

Similar DA techniques have been explored in the domain of hate speech classification. One group of approaches replicate samples by replacing similar words, based on pre-trained embeddings and cosine distance [19]. Word replacement based on features from ConceptNet and Wikidata knowledge graphs were explored in [21]. Approaches based on text transformation using back-translation are explored in [2]. Approaches based on sample generation using Long short-term memory (LSTM) and GPT-2 [18] are explored in [19, 27].

Given the significant improvements in the classification performance using the language generation based DA methods, we follow the approach by Wullach et al. [27]. The goal of this article is the experimental study of behavior of data augmentation approach in [27]. However, the contributions of this article comes with two key differences. (a) We fine-tune a single class conditioned GPT-2 language model [15], as opposed to class specific fine-tuned GPT-2 models in [27]. (b) We attempt three class classification of hate, abuse, and normal speech, which is known to be a relatively complex task due to overlap between hate speech and abusive speech [6, 10]. Additionally, we also explore the effect of the quantity and the quality of the generated data required to improve the classification performance.

To summarize, the contributions of this article are:

- Generation of training samples using conditional language model for DA in multi-class classification of hate speech.
- Analysis of how classification performance varies depending on the quantity of the additional samples.
- Study on how filtering the generated samples affects the performance.

2 Data Augmentation

In this section, we describe our approach for DA using the GPT-2 model to generate new training samples.

2.1 Conditional Language Modeling

A typical language modeling task involves learning the joint probability distribution of a sequence [4]. Given the vocabulary V containing a fixed set of distinct tokens, a sequence of n tokens $z = (z_1, z_2, ..., z_n)$ where $z_i \in V$, the joint probability distribution of the sequence is given as:

$$p(z) = \prod_{i=1}^{n} p(z_i | z_{<i}) \tag{1}$$

Given a dataset containing m samples $D = \{z^1, z^2, ..., z^m\}$, a neural language model learns the parameter set θ such that it reduces the negative log-likelihood:

$$L(D) = -\sum_{j=1}^{|D|} \log p(z_i^j | z_{<i}^j; \theta) \tag{2}$$

The language model can be trained with a conditional context c, extending equation (1) to:

$$p(z|c) = \prod_{i=1}^{n} p(z_i | c, z_{<i}) \tag{3}$$

Likewise, equation (2) extends to:

$$L(D) = -\sum_{j=1}^{|D|} \log p(z_i^j | c^j, z_{<i}^j; \theta) \tag{4}$$

Given a conditional context c, the learned parameter set θ can be used to sample l tokens and generate a new sequence $\hat{z}$ using $p(\hat{z}_t | c, \hat{z}_{<t}; \theta)$, where $t = \{1, 2, .., l\}$.

2.2 Proposed Methodology

Figure 1 shows the block diagram of our approach. We fine-tune a single pre-trained GPT-2 model for the given datasets (see §3.1) using conditional language modeling objective. We then use the fine-tuned GPT-2 model to generate a large number of samples for each class. We filter the samples using a Bidirectional Encoder Representations from Transformers (BERT) [8] model that has been fine-tuned on the original training set. Top-N samples sorted by the BERT model are augmented to the original training set to train a Convolutional-Gated Recurrent Unit (C-GRU) based classifier.

GPT-2 Fine-Tuning and Data Generation: We fine-tune a GPT-2 model on the original training set by conditioning it on the class labels. To achieve this, we prepend the class label of the sample as a conditional context. For example, a 'normal' class sentence such as *"a cat is sitting on the mat"* is transformed to *"**normal** a cat is sitting on the mat"* before using it as input to fine-tune GPT-2 model.

Filtering the Generated Sequences: Sometimes, the generated content does not match the target class. Thus, we adopt a technique similar in [27] to filter the generated samples by fine-tuning the BERT model for the multi-class classification. In order to avoid a bias induced by imbalanced class sample size in the BERT classifier, we downsample the classes to have an equal amount of samples in each class. The samples generated by the fine-tuned GPT-2 model are then passed through the fine-tuned BERT model in order to sort them according to the score given by the BERT model, finally retaining only the top-N for DA.

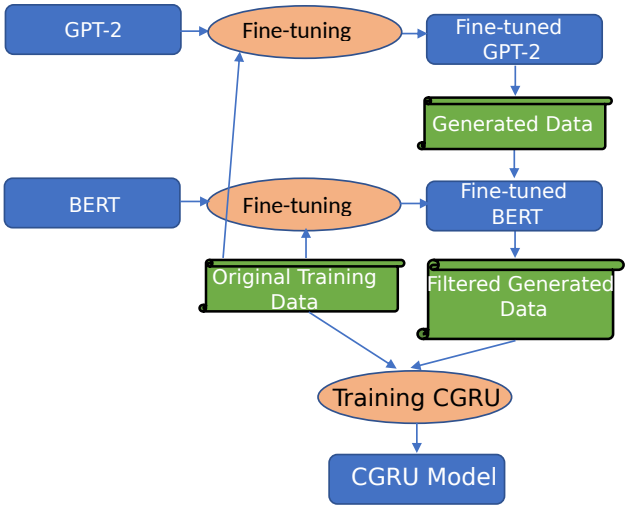

Fig. 1. Block diagram for training an improved classifier with DA.

Hate Speech Classifier: As presented in [28], the C-GRU based architecture is a powerful hate speech classifier. This model is faster to train and requires smaller computational power since it has fewer model parameters in comparison to the transformer based BERT model. Thus, as adopted in [27] we choose a similar architecture for our hate speech classification. With the C-GRU based architecture, the input sequence is first passed through convolutional layers followed by the GRU layer.

3 Experimental Setup

This section describes the datasets, text pre-processing, and the choice of hyper-parameters for the models.

3.1 Data Description

Table 1. Statistics of Founta and Davidson datasets.

Dataset	#Samples	Normal	Abusive	Hateful
Founta	86.9K	63%	31%	6%
Davidson	24.7K	17%	77%	6%

For the multi-class classification of hate speech, we chose two widely used hate speech datasets containing tweets sampled from Twitter, one by Founta et al. [12] and the other by Davidson et al. [6]. Here onwards, referred to as 'Founta' and 'Davidson'. Each dataset is randomly split into three sets, 'training', 'validation', and 'test', containing 60%, 20%, and 20% respectively.

Founta dataset is collected by boosted random sampling of data from Twitter. The dataset is annotated into four classes, named, 'normal', 'abusive', 'hateful', and 'spam'. In our study, we do not use the samples from the 'spam' class and this reduces the number of samples in the dataset from 100K to 86.9K.

Davidson dataset is collected by sampling the tweets based on keywords from the hatebase lexicon. The dataset is annotated into three classes 'hate speech', 'offensive language', and 'neither'. Since the definition of the class labels used by Founta et al. [12] was similar to Davidson et al. [6], in this article, we have referred to these classes as 'hateful', 'abusive', and 'normal' respectively.

A summary of the two corpora is available in Table 1. As indicated, 'hateful' tweets are the minority in both datasets.

3.2 Data Preprocessing

We removed all numbers and special characters except '.', ',', '!', '?', and *apostrophe*, and repeated occurrences of the same special character are changed to a single one. Twitter user handles are changed to '@USER'. The '#' symbol in the hashtag is removed, and the multi-word hashtags are split based on the presence of uppercase characters in the hashtags. For example, '#leaveThisPlace' is changed to 'leave This Place'. Finally, the data is converted to lowercase.

3.3 Model Parameters

Our model parameters are adopted from [27]. We use the implementation of huggingface's transformers API [25] to fine-tune the 'GPT-2 large' model.[3] The final generative model is chosen based on the lowest loss computed on validation set after each epoch. The class label is used as a prompt text to the fine-tuned GPT-2 model to generate samples for each specific class. Overall, we generate 600K samples for each class label.

To fine-tune the BERT model, we used the pre-trained 'BERT-base-uncased' model trained on the English corpus. We fine-tuned two BERT models, one on the training set of Founta, another on the training set of Davidson. The generated data is sorted according to the softmax score obtained by the fine-tuned BERT model.

For the C-GRU classifier, words occurring less than three times are considered as out-of-vocabulary words, and are replaced with a '⟨UNK⟩' token. For both BERT models and the C-GRU models, at the end of each epoch, the macro-averaged F1 measure is evaluated on the validation set to choose the best models. The best models are then used to sort the generated samples or for the classification.

4 Results and Discussion

We report mean and standard deviation of test set in percentage macro-averaged F1 evaluated over five separate runs. Each run uses a C-GRU classifier with a different random weight initialisation. The 95% confidence interval on macro-averaged F1 obtained using paired bootstrap [9, 11] is ±1.6 and ±2.8 for the Founta and Davidson test sets, respectively.

4.1 Improvements with Data Augmentation

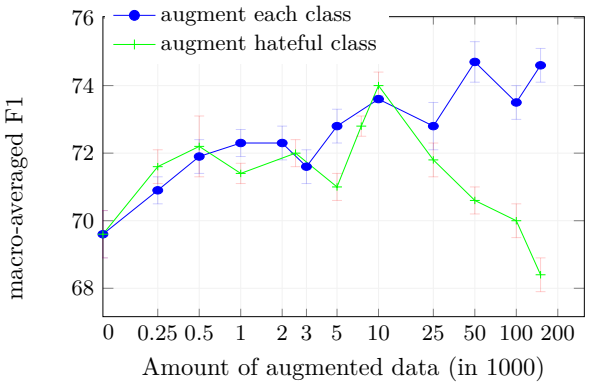

Fig. 2. Macro-averaged F1 on Founta test set using DA. The classifier is trained using DA with increasing amounts of generated data (X-Axis).

Figure 2 and Figure 3 show the macro-average F1 by varying the amount of augmented data for the Founta and Davidson datasets respectively. In these experiments, the generated data is combined with the original training data to train the classifier. We have explored two strategies, (a) augmenting each class with an equal amount of data ; (b) augmenting data only in the 'hateful' class, because the

[3] https://huggingface.co/gpt2-large

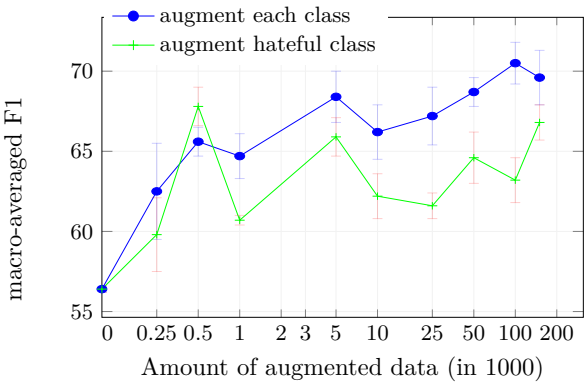

Fig. 3. Macro-averaged F1 on Davidson test set using DA. The classifier is trained using DA with increasing amounts of generated data (X-Axis).

number of samples in 'hateful' class is very small. Baseline macro-averaged F1 obtained using the C-GRU classifier without DA is 69.6 ± 0.7 for the Founta dataset and 56.5 ± 0.3 for the Davidson dataset.

Figure 2 and Figure 3 show that DA improves the classifier performance for the 'augment each class' and gives up to 7.3% of relative improvement for the Founta test set and up to 25.0% for the Davidson test set. We observe performance gains even with few hundred samples augmented with the original training set, however, the performance gain reduces as the amount of additional augmented data increases. We would like to highlight that our implementation of the non class conditional GPT-2 model based augmentation [27] resulted in similar results. Thus, we have achieved comparable performance by using three times lesser parameters to augment training data by using a class conditioned GPT-2 model.

In the 'augment hateful class' case, we observe a relative improvement to the classification performance by up to 6.2% for the Founta test set and up to 20.2% for the Davidson test set. After adding data we initially observed improvements in macro-average F1, however, as the amount of augmenting data increased, the macro-average F1 declined. An analysis of the confusion matrices revealed that the reduction in the performance is due to the classifier getting biased and predicting the 'normal' class samples incorrectly as 'hateful'. As we increase the data added only to the 'hateful' class, the model's prior probability of predicting the data as the 'hateful' class also increases.

4.2 Quality of Augmented Data

Table 2. Macro-averaged F1 for classifier trained using only the generated data.

Amount of generated data used for each class	**Founta test set**	**Davidson test set**
Baseline (no DA)	69.6 ± 0.7	56.5 ± 0.2
5K	60.2 ± 1.2	48.4 ± 1.8
10K	60.5 ± 1.6	56.0 ± 2.0
25K	64.1 ± 1.1	56.2 ± 3.7
50K	**64.6 ± 0.8**	62.6 ± 1.1
100K	64.0 ± 0.8	**67.2 ± 1.0**
150K	63.0 ± 0.2	**67.2 ± 0.8**

Table 2 shows the classification performance of the C-GRU that was trained with only the generated data. For both datasets, we observe an increase in the classifier's performance as the amount of generated data used to train the classifier is increased. For the Davidson dataset, we note that the performance is higher than the baseline when more than 50K generated samples are used for training. These results show that the generated data can be efficiently used for DA since it characterises the original training data and its classes.

Furthermore, we analysed the quality of generated samples by using it as test samples for the model trained using only the original training set. We trained the C-GRU model on Founta training set and

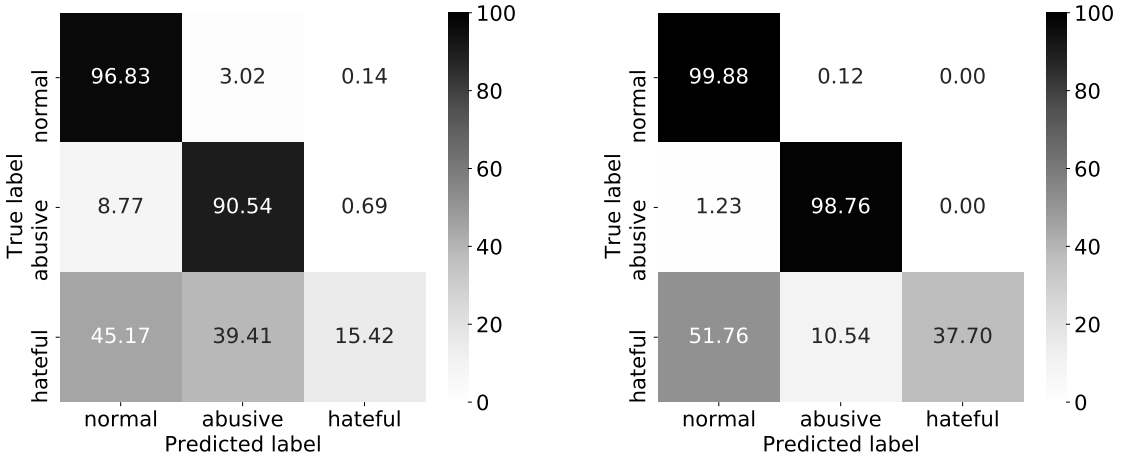

Fig. 4. (a) Confusion matrix obtained on Founta test set. (b) Confusion matrix obtained on generated samples.

compared confusion matrices obtained from the Founta test set and the top 50K samples generated from each class. The confusion matrices are shown in Figure 4. We observe that the classification performance on the generated set is much better than classification on the test set, implying that generated data is similar to the original training set. Further, we tried to improve the filtering technique by fine-tuning the BERT model using the data from both the original training set and generated set. Our preliminary experiments did not show any improvement in the final classification results.

4.3 Influence of Filtering the Samples

Table 3. Comparison of classification performance on Founta and Davidson test sets by augmenting N randomly sampled data versus top-N filtered by BERT.

	Founta test set		**Davidson test set**	
Amount of generated data used for each class	Random Sampling	Top-N scored by BERT	Random Sampling	Top-N scored by BERT
Baseline (no DA)	69.6 ± 0.7		56.4 ± 0.2	
5K	70.7 ± 0.3	72.8 ± 0.5	63.5 ± 0.4	68.4 ± 1.6
25K	70.9 ± 0.4	72.8 ± 0.7	68.5 ± 0.2	67.2 ± 1.8
50K	71.0 ± 0.6	74.7 ± 0.6	68.5 ± 0.4	68.7 ± 0.9

Table 3 shows the effect of using a fine-tuned BERT model for filtering the samples generated by GPT-2 for DA on the Founta dataset and the Davidson dataset. Here, we randomly choose N generated samples and compared them against the top-N samples sorted by the fine-tuned BERT model. Choosing the samples filtered by the fine-tuned BERT model gave a relative improvement of up to 5.2% for the Founta dataset and up to 7.7% for the Davidson dataset over the randomly chosen samples for augmentation.

Furthermore, to observe the influence of filtering, we analyze the samples generated by the GPT-2 model and filtered by BERT. Table 4, Table 5, and Table 6 present some representative examples of generated samples for the 'normal', 'abusive, and 'hateful' classes respectively. Founta dataset is used. We present the top-ranked and bottom-ranked generated samples in the data sorted by the BERT model. In Table 4, the bottom-ranked sentences are classified as 'abusive', in Table 5 as 'normal', and in Table 6 as 'abusive' or 'normal'. We can observe that the the bottom-ranked samples do not belong to the desired target class. This could be due to the fine-tuning of the class conditioned GPT-2 model on samples from all the three classes. The bottom-ranked samples were filtered out and not used to train the C-GRU classifier. This shows that BERT filtering performs a powerful selection of relevant samples from the generated data.

Table 4. Examples of high-scored and low-scored samples generated for 'normal' class by the GPT-2 model trained on Founta dataset, sorted by the BERT model.

Top-ranked generated samples
ive never seen such a beautiful and wonderfully supportive group of people. love you guys @user. looking forward to the next event!
ive been super thankful for this chance amp so glad to be a part of my generation. those in leadership need our collective leadership to be stronger.
thank you for the recent follow @user @user happy to connect have a great wednesday. need some inspiration? check out our cam...
Bottom-ranked generated samples
ive lived my entire life expecting to hear every f**king word said by people i know and trust, but instead only get, sh*ts not right man!
do re mi fa so f**king done with you girl @user - luv... ...finally done with you girl
'all of my girlfriends have cheated on me at some point in time 'oh god i hope so. its so f**ked up. and

Table 5. Examples of high-scored and low-scored samples generated for 'abusive' class by the GPT-2 model trained on Founta dataset, sorted by the BERT model.

Top-ranked generated samples
ive been so f**ked up in the head lately its scary fuck me out please @user
ik im still in the f**king stages rn like wtf
????? at the end of the day?????? that's bullsh*t can't be true
Bottom-ranked generated samples
ive been doing my bit to change the world, the thing im most passionate about is education. education is key. and it is a p... see
iphone easter egg hunt mobile version is out! hunt for! ppl have been having issues finding easter eggs. yours are here!
?????! that was years ago and i have nothing but respect for @user today! god bless you and everyone involved!??????!

5 Conclusion

In this article, we explored the use of Data Augmentation (DA) in hate speech classification. The DA is performed by generating samples from a GPT-2 model, as similar in [27]. However, we fine-tuned a GPT-2 model using the objective of conditional language modeling. Our experiments showed that augmenting a few hundred generated samples with the training set yield a significant gain in performance. Further, we showed a considerable amount of performance gain by augmenting data only to the 'hateful' class of the training set. Our experiments were validated using two widely used hate speech corpora. Additionally, we analyzed the quality of the generated data by evaluating classifiers trained only on the generated data, which showed that generated data is similar to training data. Finally, we investigated the influence of using fine-tuned BERT to filter the generated data and showed that using BERT-based filtering helps to choose pertinent samples for DA.

Acknowledgments

This work was funded by the M-PHASIS project supported by the French National Research Agency (ANR) and German National Research Agency (DFG) under contract ANR-18-FRAL-0005. Experiments presented in this article were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations. We thank Hayakawa Akira for his valuable comments and feedback and on this article.

References

1. Anaby-Tavor, A., Carmeli, B., Goldbraich, E., Kantor, A., Kour, G., Shlomov, S., Tepper, N., Zwerdling, N.: Do not have enough data? deep learning to the rescue! In: AAAI. pp. 7383–7390 (2020)
2. Aroyehun, S.T., Gelbukh, A.: Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling. In: Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018). pp. 90–97 (2018)

Table 6. Examples of high-scored and low-scored samples generated for 'hateful' class by the GPT-2 model trained on Founta dataset, sorted by the BERT model.

Top-ranked generated samples
ik wicked if you call me a n*gga.
ikorchick - i hate a party that relies on the white male vote. ikorchick might not always agree, but he will
ik i hate babies all of them
Bottom-ranked generated samples
ive been drinking vj's for the last hours and my body is still f**ked up. im going to bed, f**king sleepy bum hoe. the
ive been tryna get an account on hitmontop since m s. just gonna wait till we get hitmontop x hitmontop and we
your support makes a big difference

3. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: Proceedings of the 26th International Conference on World Wide Web Companion. pp. 759–760 (2017)
4. Bengio, Y., Ducharme, R., Vincent, P., Janvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. **3**, 1137–1155 (2003)
5. Cao, R., Lee, R.K.W., Hoang, T.A.: Deephate: Hate speech detection via multi-faceted text representations. In: 12th ACM Conference on Web Science. pp. 11–20 (2020)
6. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language. In: Eleventh international AAAI conference on web and social media (2017)
7. Delgado, R., Stefancic, J.: Hate speech in cyberspace. Wake Forest L. Rev. **49**, 319 (2014)
8. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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). pp. 4171–4186 (2019)
9. Dror, R., Baumer, G., Shlomov, S., Reichart, R.: The hitchhiker's guide to testing statistical significance in natural language processing. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). vol. 1, p. 1383–1392 (2018)
10. D'Sa, A.G., Illina, I., Fohr, D.: Bert and fasttext embeddings for automatic detection of toxic speech. In: 2020 International Multi-Conference on: "Organization of Knowledge and Advanced Technologies" (OCTA). pp. 1–5 (2020)
11. Efron, B., Tibshirani, R.J.: An introduction to the bootstrap. CRC press (1994)
12. Founta, A.M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., Kourtellis, N.: Large scale crowdsourcing and characterization of twitter abusive behavior. In: Twelfth International AAAI Conference on Web and Social Media (2018)
13. Hu, Z., Yang, Z., Liang, X., Salakhutdinov, R., Xing, E.P.: Toward controlled generation of text. In: International Conference on Machine Learning. pp. 1587–1596. PMLR (2017)
14. Isaksen, V., Gambäck, B.: Using transfer-based language models to detect hateful and offensive language online. In: Proceedings of the Fourth Workshop on Online Abuse and Harms. pp. 16–27 (2020)
15. Keskar, N.S., McCann, B., Varshney, L.R., Xiong, C., Socher, R.: Ctrl: A conditional transformer language model for controllable generation. arXiv preprint arXiv:1909.05858 (2019)
16. Kobayashi, S.: Contextual augmentation: Data augmentation by words with paradigmatic relations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). pp. 452–457 (2018)
17. Malmasi, S., Zampieri, M.: Challenges in discriminating profanity from hate speech. Journal of Experimental & Theoretical Artificial Intelligence **30**(2), 187–202 (2018)
18. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019)
19. Rizos, G., Hemker, K., Schuller, B.: Augment to prevent: short-text data augmentation in deep learning for hate-speech classification. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. pp. 991–1000 (2019)
20. Sennrich, R., Haddow, B., Birch, A.: Improving neural machine translation models with monolingual data. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 86–96 (2016)
21. Sharifirad, S., Jafarpour, B., Matwin, S.: Boosting text classification performance on sexist tweets by text augmentation and text generation using a combination of knowledge graphs. In: Proceedings of the 2nd workshop on abusive language online (ALW2). pp. 107–114 (2018)
22. Shleifer, S.: Low resource text classification with ulmfit and backtranslation. arXiv preprint arXiv:1903.09244 (2019)
23. Wang, W.Y., Yang, D.: That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In: Proceedings of the 2015 conference on empirical methods in natural language processing. pp. 2557–2563 (2015)

24. Wei, J., Zou, K.: Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 6383–6389 (2019)
25. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45. Association for Computational Linguistics, Online (Oct 2020)
26. Wu, X., Lv, S., Zang, L., Han, J., Hu, S.: Conditional bert contextual augmentation. In: International Conference on Computational Science. pp. 84–95. Springer (2019)
27. Wullach, T., Amir, A., Einat, M.: Towards hate speech detection at large via deep generative modeling. IEEE Internet Computing pp. 1–1 (2020)
28. Zhang, Z., Robinson, D., Tepper, J.: Detecting hate speech on twitter using a convolution-gru based deep neural network. In: European semantic web conference. pp. 745–760. Springer (2018)