UNDER REVIEW

RESCIENCE C

Replication / ML Reproducibility Challenge 2022

[Re] Exploring the Role of Grammar and Word Choice in Bias Toward African American English (AAE) in Hate Speech Classification

Anonymous[1, ID]
[1] Anonymous Institution

Edited by
(Editor)

Reviewed by
(Reviewer 1)
(Reviewer 2)

Received
03 February 2023

Published
–

DOI
–

Reproducibility Summary

Scope of Reproducibility – We aim to reproduce a result from the paper titled above [1]. Our study is restricted specifically to the claim that the use of swear words impacts hate speech classification of AAE text. We were able to broadly validate the claim of the paper, however, the magnitude of the effect was very much dependent on the word replacement strategy, which was somewhat ambiguous in the original paper.

Methodology – The authors' code is not available. Therefore, we reproduce the experiments by following the methodology described in the paper. We train BERT models from TensorFlow Hub [2] to classify hate speech using the DWMW17[3] and FDCL18[4] Twitter datasets. Then, we compile a dictionary of swear words and replacement words with comparable meaning, and we use this to create "censored" versions of samples in Blodgett et al.'s[5] AAE Twitter dataset. Using the BERT models, we evaluate the hate speech classification of the original data and the censored data. Our experiments are conducted on an open-access research testbed, Chameleon [6], and we make available both our code and instructions for reproducing the result on the shared facility.

Results – Our results are consistent with the claim that the censored text (without swear words) is less often classified as hate speech, offensive, or abusive than the text with swear words. However, we find the replacement of the words is very sensitive to the word replacement dictionary being used.

What was easy – The authors used three well known datasets which were quite easy to obtain. They also used well-known widely available models like BERT and word2vec.

What was difficult – Some of the details of training the BERT and word2vec models were not fully specified. Also, we were not able to exactly re-create a comparable dictionary of swear words and their replacement terms of similar meanings following their methodology.

Communication with original authors – We reached out to the authors towards the end of the challenge, but were not able to communicate with them before the submission of this report. However, we hope to stay in touch with them throughout the review process.

Correspondence should be addressed to ()
The authors have declared that no competing interests exists.
Code is available at https://github.com/rescience-c/template.

1 Introduction

African American English (AAE) text is more likely to be classified as hate speech or other type of toxic speech by hate speech classification systems and sentiment analysis, according to prior research [3, 7, 8, 9]. This paper [1] seeks to determine how AAE vernacular and the classification of hate speech relate to one another. The research aims to understand how specific characteristics of AAE text may correlate the classification of AAE text as offensive, abusive and hate speech, and inform future bias-mitigation strategies. The paper considers two common characteristics of AAE speech for such classification. One, the use of swear words and two, certain grammatical patterns. In our paper, we focus on reproducing the results which are related to the first, which is the use of swear words in AAE text.
This report repeats the original paper's experiments and compares them with the reported results. We trained two BERT classifiers [2], once each on the DWMW17 [3] and FDCL18 [4] datasets. We then categorize tweets from each datasets as being hate speech or not. The same process is repeated on the Blodgett et al.'s AAE twitter dataset [5] for a similar categorization. Next we use different word2vec models to perform word replacement, to swap out swear words with words of similar meanings. Once done, we calculate the reduction in hate speech classification, which is a substantial amount. The original paper also checks if grammatical patterns have any impact on this hate speech classification, however, we have not validated this claim in our report.
In this reproducibility report, we attempt to re-create the experiments using the same datasets and similar models as that of the original paper. We were able to broadly validate the claim that AAE text with censored words is less likely to be classified as hate speech. However, the magnitude of this effect is very sensitive to the word replacement strategy used. The details of the author's word replacement strategy was not clearly specified and hence, we were not able to replicate the exact results of the original paper. We make all of our experiment code available for replication on Chameleon, which is an open access test bed. Other researchers can easily reproduce our experiment on the same environment.

2 Scope of reproducibility

The paper explores the role of grammar and word choice in bias toward AAE in hate speech classification. The authors consider two research questions (reproduced verbatim here from the original paper), and then based on their experiment results they made one claim for each research question:

- **RQ 1:** "How strongly does use of swear words or "offensive language" impact the hate speech classification of AAE text?" **Claim 1:** There is a considerable reduction in hate speech, offensive speech, and abusive speech when tweets are censored for "swear" words.

- **RQ 2:** "How do grammatical patterns of AAE tweets impact the hate speech classification of AAE text?" **Claim 2:** The classification of hate speech is independent of the four grammar subcategories that the authors examined.

In this report, we attempt to reproduce and validate only the first claim, which is to check on how words that are described as offensive according to Standard American English impact the hate speech classification on African American English text. To validate this claim, we trained a BERT classifier[2] on the DWMW17 [3] and FDCL18 [4] hate speech training datasets. Then, we generated censored and uncensored versions of a subset of the AAE text in the Blodgett AAE [5] dataset. We validate that the censored AAE text is less likely than the uncensored text to be classified as abusive, offensive, or hateful.

Since there was no original code provided, all the code that is written is our own, using the description in the paper as a guideline.

3 Methodology

To replicate the experiments in the original paper, we retrieve the same datasets as used by the original authors, train comparable BERT and word2vec models, generate word replacement dictionaries, and then we classify censored and uncensored text samples using those BERT models. Here, we elaborate more on each of those steps, including the challenges we encountered in trying to follow the authors' instructions.

3.1 Datasets

There were 3 main datasets used in this paper and for its verification.

- DWMW17 [3]
- FDCL18 [4]
- Blodgett [5]

The first two datasets are Twitter hate speech datasets that categorize tweets by the various hate speech terms. For DWMW17 [3], the categorization is done as "hate speech", "offensive speech" or "neither". For FDCL18 [4], the categorization is done as "Abusive", "Hate", "Spam" or "normal". These datasets are used to train the BERT models.
The Blodgett dataset [5] is being used as the AAE text source. We select a sample of AAE text from this dataset (the specific IDs of the text samples we used are available in the supplementary materials), generate censored versions of this text using different word replacement strategies to replace swear words, and compare their classification with and without censoring.
We would also like to mention that the Twitter API does not allow researchers to redistribute the text. Hence, it is not clear what data the authors used originally and if there are any missing samples in the dataset that we used for our experiment. This is a common and well-known problem with using Twitter data.

3.2 Models

The original paper uses two types of models:

- **Offensive, abusive, or hateful speech classification**: the authors train two BERT models, one on the DWMW17 [3] dataset and one on the FDCL18 [4] dataset, to classify text.
- **Word replacement**: the authors train a word2vec model on AAE texts in the Blodgett [5] dataset, to find replacement words for each swear word using the closest match by cosine similarity.

The authors did not specify hyperparameters or other further details related to model training. Since our primary goal is to validate the broad claim, and not necessarily specific numeric results, we did not do an extensive hyperparameter search. We used a BERT classifier from the Tensorflow Hub named "bert_multi_cased_L-12_H-768_A-12" [10], followed the process described in this article [11] to train it on each of two datasets[3, 4], and train each model for 30 epochs. In the paper [1], the authors mention that "First we split the dataset appropriately and train a BERT classifier on the data. Then we test the model and use the best performing model". However, since there were no specific instructions provided on how this was done exactly, we did not do the same for our reproducibility experiment either. Training the BERT Classifier for 30 epochs on each of the DWMW17 [3] and FDCL18 [4] achieves an accuracy of 98.5% and 97% accuracy, respectively. And the validation accuracy for DWMW17 [3] was 90.3% while it was 91.2% for FDCL18 [4].

For the word replacement model, the authors say they train a word2vec model on 50,000 AAE tweets from the Blodgett dataset (representing a subset of about 10% of the entire AAE tweet dataset). We sampled 50,000 AAE tweets from the same dataset (the Tweet IDs are available in the supplementary materials) and trained a word2vec model to find replacement words for the swear words in the original tweets. The authors in the paper have done this replacement using the word2vec model which was built on top of the Blodgett Dataset and the replacement words were found using the cosine similarity to other words as well as a list of manually created replacement words for swear words - the details of which were not provided. However, sometimes, we observed that the replacement words were also swear words. Hence, we used a pre-trained word2vec model on the Google News 300 dataset as well as censored words by replacing all the characters in the swear word with an asterisk symbol to create new replacement words (Ex: ass is transformed to ***).

3.3 Word replacement

For word replacement, the authors say they used LIWC2007 [12] and a hand curated list of swear words. However, they did not provide their word replacement dictionary to us. Hence, we used a list of swear words from Google News 300 [13] dictionary. We also tried out censoring of the swear words by replacing all the characters in the swear word by creating a new word of the same length but all the characters as asterisks.

The authors claim that, on manual inspection of a set of 50 tweets, 78% were successfully reworded (i.e. the censored tweet has comparable meaning). However, it is not clear what qualifies as "successful". During our experiment, we noticed that there were certain words in the replacement dictionaries that made no sense, sometimes the replacement words were also swear words, sometimes the words noted as swear words weren't actually swear words, and also, sometimes the swear words were just the asterisked versions of the offensive words.

Hence, we also considered two other word replacement strategies to get around these limitations:

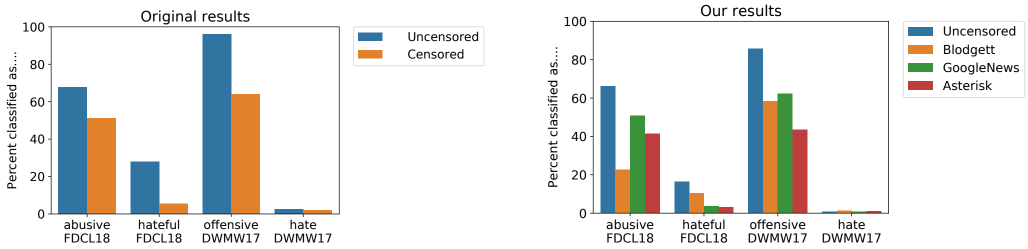
- Standard word2vec on Google News [13]: We used a word2vec model trained on the Google News 300 dictionary to get the censored tweets and ran our BERT classifier on these new tweets.
- We created our own asterisk dictionary, where the replacement word for the swear word was a word of same length with all its characters as asterisks.

3.4 Classifying censored and uncensored text

We prepared comparable experimental setup as there was no source code provided for the original experiment. We used instances on the Chameleon [6] cloud to run our experiments on and our whole code base can be found in the Google Colab notebooks uploaded as a part of the supplementary materials.

3.5 Computational requirements

We used Chameleon [6] cloud to provision all our resources required for the experiments. Chameleon [6] is an expansive, extensively adaptable experimental platform created to aid systems research in the computer sciences. We used Chameleon as users have complete control over the software stack, including root rights, kernel customisation, and



(a) Original results reported in [1].

(b) Our results.

Figure 1. Original results reported in [1], and our results after reproducing their experiments.

console access, thanks to Chameleon's support for bare metal reconfiguration systems. We ran our experiments using the RTX 6000 GPU. The training of our model took approximately 12h using the original approach.
For our entire experiment, approximately 7610 service units were used up. One Service Unit (SU) on Chameleon is equivalent to one hour of usage of one allocatable resource (physical hosts, network segments, or floating IPs). We have included instructions to provision resources on Chameleon in the supplementary materials so that our work can be reproduced easily.

4 Results

We were able to reproduce the claim that there is a significant reduction in hate speech classification of AAE tweets when the "swear" words are censored. However, the magnitude of the effect was different in our case, as compared to what was mentioned in the paper. Also, the authors reported that on manual inspection of 50 tweets, they observed a 78% success rate in removing all swear words and having identical meaning to the original tweet. This is not true for our investigation. We experimented with a few different dictionaries however, had difficulty in getting results as good as the ones that were reported.

4.1 Results reproducing original paper

Fig. 1a shows the results of the original paper, for both the uncensored tweets as well as the tweets with the swear words replaced. We reproduced the results which are mapped to the first claim of the paper. We provide stacked graphs in Fig. 1b showing the comparison of the models with each other, in terms of the percentage of hate speech classification when the speech is uncensored v/s when it is censored. We also tabulate examples of certain swear words and their replacement words with similar meanings according to the different dictionaries in table 1.

Result 1 — First we trained our word2vec model on the Blodgett dataset and found that that the results were almost comparable to the ones that were mentioned in the paper. However, we found certain words that were getting replaced with other words that did not have similar meanings. For example, the replacement word for "bitches" was getting replaced with the word "hoes" which in itself is an offensive word. Hence, we worked on two other word2vec models for word replacement.
We used a pre-trained word2vec model on the Google News 300 [13] as it has a better and more exhaustive list of swear words. This gave us better results. Taking the last example again, the word "bitches" here got replaced with the word "girls" which is of comparable meaning and is not a swear word itself.

Table 1. Comparison of replacement words as generated by the different dictionaries

Word	Replacement word according to Dictionary 1 (Google News 300)	Replacement word according to Dictionary 2 (Blodgett)	Replacement word according to Dictionary 3 (Asterisk)
ass	butt	ahh	***
nigga	boy	boy	*****
bitches	girls	hoes	*******
hell	h_*	usual	****
fucker	f_**ker	hismain_concerned	******
shit	sh_*_t	shyt	****
WTF	OMFG	<Not available in Blodgett>	***
PISSED	DAMMIT	<Not available in Blodgett>	******
dick	d_*_ck	neck	****

Table 2. Results from the Original Paper on Blodgett Dataset

	DWMW17 - Hate	FDCL18 - Hate	DWMW17 - Offensive	FDCL18 - Abusive
Original Sample	2.46	27.89	96.18	67.77
Edited Sample (Asterisk)	1.93	5.496	64.07	51.04
Difference (Original v/s Asterisk)	-0.53	-22.394	-32.11	-16.73

Finally, we just wanted to see the effect masking all the characters of the swear words with the asterisk character in the replacement word, has on the classification of hate speech. The table above has examples of replacement words that were given by all the three different word2vec models.
As can be observed from the 1, the Google 300 News word2vec model to generate a replacement dictionary makes the most sense as compared to Blodgett dictionary that has been used in the paper. However, since the results weren't exactly similar, we also tried out an asterisk dictionary created by us, to just replace the swear words with the asterisked out word replacements. Ex: ass becomes ***, as opposed to ahh in the case of Blodgett or butt in the case of Google News 300 (see table 1 above).
Table 3 contains the results of our experiment while the table 2 contains the results of the paper. We can observe that the results of our experiment are almost similar in magnitude as that of the author and hence, helps to validate the claim of the paper and does show that there is a notable decrease in hate speech classification of AAE text, once the word replacement happens for ”swear words”.

4.2 Results beyond original paper

Apart from the original experiment mentioned in the paper, there were two additional experiments we did in the word replacement part of the activity. For the word replacement, apart from using the word2vec model on only the Blodgett dataset as specified in the paper, we also created a model on the:

1. Google News 300 dataset.
2. New words created by replacing the swear words with words of same length but all characters as asterisks.

Additional Result 1 — On running the model on the AAE text of the Blodgett dataset, we obtained the we obtained the results as mentioned in the table 3.

Additional Result 2 — On running the model on the AAE text of the Google News 300 word2vec dataset, we obtained the results as mentioned in the table 4.

Table 3. Results from the Our Experiment on Blodgett Dataset

	DWMW17 - Hate	FDCL18 - Hate	DWMW17 - Offensive	FDCL18 - Abusive
Original Sample	0.95	16.52	85.86	66.22
Edited Sample (Blodgett)	1.00	10.60	56.64	22.18
Difference (Original v/s Blodgett)	0.05	-5.92	-29.22	-44.04

Table 4. Results for the Google News 300 dataset

	DWMW17 - Hate	FDCL18 - Hate	DWMW17 - Offensive	FDCL18 - Abusive
Original Sample	0.95	16.52	85.86	66.22
Edited Sample (Google)	0.82	3.81	62.36	51.00
Difference (Original v/s Google)	-0.13	-12.71	-23.5	-15.22

Additional Result 3 — On running the model on the AAE text of the swears words replaced with words of same length and all asterisk symbols, we obtained the results as mentioned in the table 5.

5 Discussion

To summarise, our experiments do validate the high-level claim made in the paper[1] regarding their research question 1: How strongly does use of swear words or "offensive language" impact the hate speech classification of AAE text? We observe that the results are qualitatively similar to what has been said in the paper. However, we find that the magnitude of the effect is highly dependent on the details of the word replacement strategy, which was somewhat ambiguous in the original paper. We feel that the experiment could've been replicated better if we had access to the actual word replacement dictionary that the authors used for their experiment.
Also, since the time this paper was published, to when we replicated this paper, due to the nature of the Twitter data, there is a high possibility that some tweets were removed from the dataset. Hence, when we retrieve tweets using the Twitter API, we get a different dataset as opposed to what the authors must have received originally. This might also be a reason in some of the minor differences in the results we observed.

5.1 What was easy

The datasets used by the authors in the paper are easily obtainable. We were able to find the git repositories hosting this data and downloaded it to run our experiments. The steps to be followed in the experiment were well documented. Having the number of samples they used at each step of the experiment was a good reference point for us follow while validating the claim. Finally, they used models that are widely available. For instance, we used the TensorFlow implementation of the BERT [2] model as a reference example for our experiment.

Table 5. Results for the Asterisks word replacements

	DWMW17 - Hate	FDCL18 - Hate	DWMW17 - Offensive	FDCL18 - Abusive
Original Sample	0.95	16.52	85.86	66.22
Edited Sample (Asterisk)	1.066	3.27	43.63	41.611
Difference (Original v/s Asterisk)	0.116	-13.25	-42.23	-24.609

5.2 What was difficult

The most challenging part about reproduction of the paper was that the details of some experiments were not fully specified. There were certain parts of the paper that were ambiguous at times and we have listed the specifics below:
Datasets: The authors acknowledge that many tweets in the original dataset that could not be accessed by them. This is a well known issue with Twitter datasets and we had issues around the same. Moreover, the authors state that they split the data "appropriately", however no clarity is provided around how they actually split the data into training and testing sets.
Models: While we agree that easily available models were used, there was still obscurity around how exactly the experiments were run by the authors. For instance, they specified that they test the model and choose the best performing model. However, there is no clarification provided around how this training was done and what was the criteria for choosing the "best performing model".
Word Replacement Dictionary: Another area where there details were missing is how the swear words to replacement words with the closest meaning of the swear word was constructed.

5.3 Communication with original authors

Towards the end of the challenge, we reached out to the authors asking them for help with the swear words to replacement words dictionary they had built, but couldn't get a response back in time for the submission. We however, hope to be in touch with the authors throughout the review process.

References

1. C. Harris, M. Halevy, A. Howard, A. Bruckman, and D. Yang. "Exploring the Role of Grammar and Word Choice in Bias Toward African American English (AAE) in Hate Speech Classification." In: **2022 ACM Conference on Fairness, Accountability, and Transparency**. FAccT '22. Seoul, Republic of Korea: Association for Computing Machinery, 2022, pp. 789–798.
2. TensorFlow. **bert_en_uncased_L-12_H-768_A-12**. https://tfhub.dev/tensorflow/bert_en_uncased_L-12_H-768_A-12/4.
3. T. Davidson, D. Bhattacharya, and I. Weber. In: **Racial Bias in Hate Speech and Abusive Language Detection Datasets**. Association for Computational Linguistics, 2019.
4. A.-M. Founta, C. Djouvas, D. Chatzakou, I. Leontiadis, J. Blackburn, G. Stringhini, A. Vakali, M. Sirivianos, and N. Kourtellis. **Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior**. 2018. arXiv: 1802.00393 [cs.SI].
5. S. L. Blodgett, J. Wei, and B. O'Connor. "Twitter Universal Dependency Parsing for African-American and Mainstream American English." In: **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 1415–1425.
6. K. Keahey, J. Anderson, Z. Zhen, P. Riteau, P. Ruth, D. Stanzione, M. Cevik, J. Colleran, H. S. Gunawi, C. Hammock, et al. "Lessons learned from the chameleon testbed." In: **Proceedings of the 2020 USENIX Conference on Usenix Annual Technical Conference**. 2020, pp. 219–233.
7. S. Kiritchenko and S. Mohammad. "Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems." In: **Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics** (2018).
8. M. Halevy, C. Harris, A. Bruckman, D. Yang, and A. Howard. "Mitigating Racial Biases in Toxic Language Detection with an Equity-Based Ensemble Framework." In: **Equity and Access in Algorithms, Mechanisms, and Optimization** (Oct. 2021).
9. M. Sap, D. Card, S. Gabriel, Y. Choi, and N. A. Smith. "The Risk of Racial Bias in Hate Speech Detection." In: **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 1668–1678.
10. TensorFlow. **Classify text with Bert**. https://www.tensorflow.org/text/tutorials/classify_text_with_bert.
11. **Bert Source Code**. https://tinyurl.com/bert-github-source-code.

12. LIWC 2007. http://www.gruberpeplab.com/teaching/psych231_fall2013/documents/231_Pennebaker2007.pdf.
13. FSE/word2vec-google-news-300 · hugging face.