Exploring Methods to Improve Implicit Hate Speech Detection with Data Augmentation

Wesley Chang
DATASCI 266: Natural Language Processing, Spring 2024
UC Berkeley School of Information
wesleychang@ischool.berkeley.edu

Abstract

Implicit hate speech detection is a difficult text classification task due to the dynamic nature of implicit messages, as well as the scarcity of human annotated data. In this paper, we explore the fine-tuning of a pre-trained BERT model (HateBERT) using data with heavy class imbalances on implicit speech nature. We explore the use of data augmentation techniques such as back-translation and easy data augmentation to generate synthetic examples for the fine-tuning task. We find that data augmentation shows consistent improvement to the macro F_1 score and provides a beneficial technique to navigate class imbalances.

1 Introduction

Implicit hate speech refers to coded and euphemistic ways to convey hate speech with ambiguous interpretation. Implicit and hateful messages are often communicated with seemingly innocuous or socially acceptable messages that obscure a much more malicious meaning (Kennedy et al.). This phenomenon is prolific on social media, where such language is used to circumvent harmful language filters that target racist language. Due to frequent use of double entendres, unconventional use of language, and dynamic development of new meanings, the problem of identifying implicit hate speech remains a large challenge for online platforms that traditionally rely on identifying hate speech using explicit keywords and filters to remove harmful content (Govers et al., 2023).

In recent years, development in the NLP space has rapidly shifted towards the use of transformer-based models for all varieties of language tasks such as machine translation, summarization, question answering, and of interest to this study, text classification (Xiao and Zhu). The rise of transformers such as the GPT and BERT families of models provide large parameter (100 million+) models pre-trained models on massive amounts of text (Radford et al., 2018; Devlin et al., 2019) that are capable of capturing deeper contexts and meaning due to the multi-headed self-attention mechanism (Vaswani et al., 2023).

Despite the flexibility of transformers in NLP tasks, hate speech detection remains a challenge for researchers due to the scarcity of available datasets stemming from the difficulty of concretely identifying implicit messages at scale for even human annotators (Govers et al., 2023). Therefore, implicit hate speech detection often encompasses a secondary task of addressing heavy class imbalances (ElSherief et al., 2021). In this study, we explore the use of data augmentation techniques to bolster fine-tuning tasks on a pre-trained BERT model, HateBERT, employing Easy Data Augmentation (EDA) and back-translation to balance minority classes with synthetic examples.

2 Background

2.1 Data Augmentation

Data augmentation refers to a set of techniques to create new examples of training data in machine learning tasks without the collection of real data. It is widely used in the computer vision domain and has gained traction in the NLP space in recent years, with promising results for tasks that have relatively small training sizes (Feng et al., 2021). In NLP, data augmentation is commonly applied in low-resource tasks that lack robust examples in order to generate a larger dataset.

2.2 Easy Data Augmentation

EDA is a rule-based based approach (Shi et al., 2022) to NLP data augmentation that attempts to manipulate words in a sentence using a set of four operations: *synonym replacement*, *random insertion*, *random swap*, and *random deletion* (Wei and Zou, 2019). These operations are conducted on a subset of a given example of text with a randomized probability of selection, modifying text on a

word-level basis. EDA presents a conceptually simple approach to data augmentation with promising results comparable to an unaugmented dataset of similar size (Wei and Zou, 2019) despite the naive nature of modification.

2.3 Back-translation

Back-translation is a model-based approach (Shi et al., 2022) to NLP data augmentation that leverages a machine translation model to translate an example of text from the source language to an intermediary language and a second model to translate from the intermediary language back to the source language (Quteineh et al., 2020). Back-translation is an expensive data augmentation method in overhead, requiring the outputs of two models to generate new examples, but has potential to create semantically similar texts with fundamentally different structures (Sennrich et al., 2016).

2.4 HateBERT

HateBERT is an encoder-only model fine-tuned on the RAL-E dataset, a corpus of approximately 1.5 million English-language messages from the social media platform Reddit. HateBERT was retrained from the model `bert-uncased`[1], a 110 million parameter version of BERT that ignore letter case in the tokenization process. Results from Caselli et al. (2021) find that there is a consistent improvement over BERT for general hate speech detection, suggesting that HateBERT was able to learn hate speech representations effectively. For this study, we take the base HateBERT model without downstream task training and apply LoRA to train the sequence classification task.

2.5 LoRA

LoRA, or Low-Rank Adaptation of Large Language Models, was introduced in 2021 as a retraining method for transformer architectures that greatly reduces the number of parameters retrained to a small fraction of the model. In ideal conditions, LoRA has been found to reduce trainable parameters by a factor of 10,000 and reduce the GPU memory requirement by a factor of 3, while retaining model performance nearly on par with traditional retraining off all parameters for GPT models (Hu et al., 2021). This is accomplished by freezing pre-trained model weights and adding trainable rank decomposition matrices to each layer in the model, approximating parameter updates by only optimizing the injected matrices. Due to the minimal tradeoff in performance for significantly reduced compute requirements, LoRA was chosen to be used for the fine-tuning steps in this study.

[1] `https://huggingface.co/google-bert/bert-base-uncased`

3 Data

We found that many of the available datasets with annotations for implicit hate speech comprised of combinations of other datasets, thus reducing potential pool of datasets in consideration for this study. From this reduced set, we selected two corpora: Implicit Hate Corpus, compiled by ElSherief et al. (2021) and ISHate, compiled by Ocampo et al. (2023) due to a lack of data overlap in their data sources.

3.1 Implicit Hate Corpus - Dataset a

The Implicit Hate Corpus (IHC) contains a total of 21,480 Tweets methodically gathered from ideological hate clusters as identified by the Southern Poverty Law Center (ElSherief et al., 2021). These Tweets were then labeled by paid annotators on increasing dimensions of detail about the nature and presence of hate speech in each example. We take the subset of results containing implicit hate speech annotation for training and evaluation in this study.

3.2 ISHate - Dataset b

The ISHate dataset contains a total of 29,116 hand-annotated examples of English-language hate speech taken and enriched from 7 standardized datasets used in hate speech detection. The 7 source datasets contained annotations of comments/posts from forums and social media, as well as GPT3 generated messages generated through prompt engineering (Chiu et al., 2022).

3.3 Class Imbalance

Class imbalance exists between the three classes of interest `["Non-HS", "Implicit-HS", "Explicit-HS"]` as illustrated in Table 1. The IHC and ISHate data have been segmented into respective train, validation, and test splits and saved to separate files, following a 70%-15%-15% split. The class imbalance is preserved in the split files with a proportional sampling strategy.

Dataset	Non-HS	Explicit-HS	Implicit-HS	Total
a: IHC				
full	13,291	1,089	7,100	21,480
train	9,304	762	4,970	15,036
val	1,994	163	1,065	3,222
test	1,993	164	1,065	3,222
	62%	5%	33%	(%)
b: ISHate				
full	17,869	10,009	1,238	29,116
train	12,508	7,007	866	20,381
val	2,680	1,501	186	4,367
test	2,681	1,501	186	4,367
	61%	34%	4%	(%)

Table 1: Illustration of class imbalance in overall datasets, by count and percentage.

4 Approach

4.1 Evaluation Metrics

The experiments conducted in this study sought to minimize the tradeoff between precision and recall and therefore utilized the macro average F_1 score as the primary evaluation metric. Hate speech detection models aim to maintain this balance between precision and recall, seeking to identify as many hate speech examples without being overly judicious and retrieving many false positives (Govers et al., 2023). This choice of metric merges precision and recall by taking the harmonic mean of both scores, collapsing both metrics into one, making it a clear benchmark for this study. Precision and recall are retained in the results section for secondary analysis.

4.2 Data Preprocessing

For all experiments conducted in this study, a number of pre-processing techniques are applied on the raw text found in the IHC and ISHate corpora. First, personal identifiers and uniform resource locators (URLs) are stripped and replaced with the tokens `user` and `url`. Second, emoji icons have been replaced with a text representation using the `emoji` [2] module. Third, leading and extra whitespace have been stripped from each text example. Fourth, since the data will be trained and tested on the HateBERT model, derived from `bert-uncased`, all capitalized characters in each example were preemptively converted to their lower-case variants. Finally, as text IDs in the ISHate corpus for each example reflect the ID from their respective source datasets, these IDs have been replaced by a sequential index across the entire ISHate corpus for simpler reference in post-hoc analysis.

4.3 Data Augmentation

Data augmentation techniques were implemented using the `nlpaug`[3] library to create new training examples. To balance the dataset, we redistributed the class labels to a 33%-33%-33% ratio across all three labels for the training splits. Majority class examples were downsampled to fit this ratio, while minority class examples were augmented to create new examples until the 33% ratio was reached. As data augmentation is intended to create new examples with similar meanings, the class label of the source example is propagated forward to augmented examples.

4.3.1 EDA

We utilize all four EDA operations to augment the training data by drawing from a uniform distribution of discrete values $[0, 3]$, mapped to each operation and selecting an operation to apply accordingly. The selected operation is then applied to examples drawn from the minority class with replacement until the threshold ratio is reached. The augmentation selection strategy is consistent across all four operations: 30% of words in a provided example are randomly drawn to be augmented.

4.3.2 Back-translation

The back-translation task relies on German as the intermediary language, using the `wmt19-en-de`[4]

[2] https://github.com/carpedm20/emoji

[3] https://github.com/makcedward/nlpaug

[4] https://huggingface.co/facebook/wmt19-en-de

Version	Non-HS	Explicit-HS	Implicit-HS	Total
a: IHC				
Baseline Training	9,304	762	4,790	14,856
	62%	5%	33%	(%)
Augmented	4,790	4,790	4,790	14,370
	33%	33%	33%	(%)
b: ISHate				
Baseline Training	12,508	7,007	866	20,381
	61%	34%	4%	(%)
Augmented	7,007	7,007	7,007	21,021
	33%	33%	33%	(%)

Table 2: Redistributed label class ratios, by count and percentage.

and `wmt19-de-en`[5] models to make the corresponding translations, based on methodology established by Ng et al. (2019). These augmented examples were generated by applying back-translation on oversampled examples from the minority class, filling in the difference needed to meet the threshold ratio.

4.4 Modeling

All experiments in this study utilize a sequence classification approach to predict three outputs: `["Non-HS", "Implicit-HS", "Explicit-HS"]`. Model inputs comprise of the text from the IHC and ISHate corpora, varying training data based on augmentation technique employed. This text is then processed using the HateBERT tokenizer in preparation for both the training and evaluation tasks. LoRA is then used to train the downstream binary classification task, of which the best checkpoint is saved based on performance in the F_1 score of the validation set. Validation and test sets are left untouched, allowing for direct comparisons between experiments.

4.5 Fine-Tuning: LoRA

LoRA was applied on top of the HateBERT model using the PEFT[6] library for each experiment. Hyperparameters were set as follows: task type as sequence classification, rank to 8, alpha to 16, and dropout to 0.1. Each model was trained for 10 epochs and the final model was selected according to the highest F_1 score obtained.

[5] https://huggingface.co/facebook/wmt19-de-en
[6] https://github.com/huggingface/peft

Experiment	Training Data
1. Baseline	unaugmented
2. EDA	augmented with EDA
3. Back-translation	augmented with back-translation

Table 3: Primary Grouping of Experiments Conducted

5 Models

All models aim to replicate exactly the tokenization and architecture used in HateBERT model, differing in only the training data used. Base hyperparameters were preserved from HateBERT, with the only departure being batch size, increased to 20 from 16. We can group our models into three experiments based on training data used (Table 3), with each experiment yielding two models for a total of six models trained in this study.

6 Results and Analysis

Between the baseline and augmented models, there is a general trend of improvement in our overall metric, macro F_1. IHC models show an 84% improvement in macro F_1 from model (1a) to (2a) and 76% from models (1a) to (2a). ISHate models show a less drastic change, but still exhibit an upward trend from Experiment 1 to Experiment 3. Baseline models for both IHC and ISHate both retrieved zero instances of their minority class in their respective test sets, as shown by the zero-valued class F_1 in models (1a) and (1b). The increase in macro F_1 score is generally attributable to improvements in predicting the minority class.

The IHC models appear to greatly benefit from the class ratio redistribution and augmentation strategy, owing to improvements in predictions of the

Experiment	F_1*	F_1 Not HS	F_1 Explicit	F_1 Implicit	Precision*	Recall*
a: IHC						
1a. Baseline	0.25	0.76	0.00	0.00	0.21	0.33
2a. EDA	0.46	0.71	0.10	0.56	0.56	0.47
3a. Back-translation	0.44	0.72	0.03	0.57	0.54	0.46
b: ISHate						
1b. Baseline	0.55	0.88	0.78	0.00	0.54	0.56
2b. EDA	0.56	0.85	0.75	0.07	0.62	0.56
3b. Back-translation	0.58	0.85	0.75	0.13	0.62	0.58

Table 4: Experiment Results, * columns represent macro averages across all label classes.

minority class. The IHC baseline model (1a) failed to identify any instances outside the dominant class, "Non-HS", returning a zero-valued F_1 score for those classes. Inspection of test set predictions also confirmed this observation. We reason that this is a combination of two factors. First, even with 33% of training examples, the model was unable to learn to distinguish implicit hate speech. Second, explicit hate speech is the minority class in the IHC dataset, forming a scarce 5% of all observations. Because of the small presence of explicit hate speech labels in this data, it is reasonable that the model learned to avoid predicting this class during fine-tuning, despite the HateBERT model being originally trained to classify this label. It's clear that augmenting on the oversampled minority class improved the retrieval here. With data augmentation, the middle class F_1 (implicit hate speech) rose sharply from 0.00 to 0.56-0.57 for both Experiments 2 and 3 (models 2a, 2b).

The ISHate models exhibit much less improvement through data augmentation compared with the IHC models, but already demonstrated a higher baseline F_1 score of 0.55. We observe that F_1 scores for the dominant non hate speech class and the middle explicit hate speech class both slightly decay, but F_1 of the minority class, implicit hate speech, reaches 0.13 in Experiment 3 (3b). Overall, the difference in magnitude between improvement in the implicit class and decay in the explicit and non hate speech classes still nets an increase in the macro F_1 score, from 0.55 in model (1b) to 0.58 in model (3b).

7 Conclusions

This project showed the utility of redistributing class ratios in unbalanced datasets through its consistent improvements, confirming the findings of Wei and Zou (2019) and Sennrich et al. (2016). As the task of implicit hate speech detection suffers from data sparsity and non-trivial annotation, data augmentation is particularly of interest in this domain for researchers seeking to tackle this issue. Further research building on the results of this study can look to more sophisticated data augmentation methods such as sentence-level augmentation using context-level embeddings or abstractive summarization to investigate their potential (Shi et al., 2022).

References

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2021. HateBERT: Retraining BERT for Abusive Language Detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, Online. Association for Computational Linguistics.

Ke-Li Chiu, Annie Collins, and Rohan Alexander. 2022. Detecting Hate Speech with GPT-3. ArXiv:2103.12407 [cs].

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv:1810.04805 [cs].

Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. 2021. Latent Hatred: A Benchmark for Understanding Implicit Hate Speech. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 345–363, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A Survey of Data Augmentation Approaches for NLP. ArXiv:2105.03075 [cs].

Jarod Govers, Philip Feldman, Aaron Dant, and Panos Patros. 2023. Down the Rabbit Hole: Detecting Online Extremism, Radicalisation, and Politicised Hate Speech. *ACM Computing Surveys*, 55(14s):1–35.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. ArXiv:2106.09685 [cs].

Brendan Kennedy, Drew Kogon, Kris Coombs, Joe Hoover, Christina Park, Gwenyth Portillo-Wightman, Aida Mostafazadeh, Mohammad Atari, and Morteza Dehghani. A Typology and Coding Manual for the Study of Hate-based Rhetoric.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 News Translation Task Submission. ArXiv:1907.06616 [cs].

Nicolas Ocampo, Ekaterina Sviridova, Elena Cabrio, and Serena Villata. 2023. An In-depth Analysis of Implicit and Subtle Hate Speech Messages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1997–2013, Dubrovnik, Croatia. Association for Computational Linguistics.

Husam Quteineh, Spyridon Samothrakis, and Richard Sutcliffe. 2020. Textual Data Augmentation for Efficient Active Learning on Tiny Datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7400–7410, Online. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. ArXiv:1511.06709 [cs].

Yiwen Shi, Taha ValizadehAslani, Jing Wang, Ping Ren, Yi Zhang, Meng Hu, Liang Zhao, and Hualou Liang. 2022. Improving Imbalanced Learning by Pre-finetuning with Data Augmentation.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. ArXiv:1706.03762 [cs].

Jason Wei and Kai Zou. 2019. 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)*, pages 6381–6387, Hong Kong, China. Association for Computational Linguistics.

Tong Xiao and Jingbo Zhu. Introduction to Transformers: an NLP Perspective.