A Study of the Class Imbalance Problem in Abusive Language Detection

Yaqi Zhang,[1] **Viktor Hangya**[2,3] and **Alexander Fraser**[1,2,3]
[1]School of Computation, Information and Technology, Technical University of Munich
[2]Center for Information and Language Processing, LMU Munich
[3]Munich Center for Machine Learning
yaqi.zhang@tum.de {hangyav, fraser}@cis.lmu.de

Abstract

Abusive language detection has drawn increasing interest in recent years. However, a less systematically explored obstacle is label imbalance, i.e., the amount of abusive data is much lower than non-abusive data, leading to performance issues. The aim of this work is to conduct a comprehensive comparative study of popular methods for addressing the class imbalance issue. We explore 10 well-known approaches on 8 datasets with distinct characteristics: binary or multi-class, moderately or largely imbalanced, focusing on various types of abuse, etc. Additionally, we propose two novel methods specialized for abuse detection: AbusiveLexiconAug and ExternalDataAug, which enrich the training data using abusive lexicons and external abusive datasets, respectively. We conclude that: 1) our AbusiveLexiconAug approach, random oversampling, and focal loss are the most versatile methods on various datasets; 2) focal loss tends to yield peak model performance; 3) oversampling and focal loss provide promising results for binary datasets and small multi-class sets, while undersampling and weighted cross-entropy are more suitable for large multi-class sets; 4) most methods are sensitive to hyperparameters, yet our suggested choice of hyperparameters provides a good starting point.

1 Introduction

The rapid expansion of social media platforms facilitates easy expression of opinions. However, the anonymity and lack of accountability can encourage speaking without inhibition, especially in an aggressive, offensive, or hateful way. To confront the surging amount of user-generated web content, we need effective automatic approaches to detect abusive content. Various systems and datasets have been introduced recently, such as for hate speech (de Gibert et al., 2018), offensive language (Davidson et al., 2017), cyberbully (Chen et al., 2012) and sexism (Samory et al., 2020) detection. Therefore, we consider abusive language as an umbrella term to refer to a wide range of improper content.

Since the majority of accessible online texts are not abusive, only a small portion of the data falls into the positive (abusive) classes, leading to imbalanced label distribution in the available resources. In some datasets, an abusive class may comprise only a few percent of all data, even as low as 4% as in the dataset released by Bretschneider et al. (2014). Class imbalance impedes learning and classification performance of machine learning algorithms, leading to over-classifying the majority classes. Previous approaches attempt to mitigate the issue with specific techniques, such as downsampling the majority and augmenting the minority class (Rizos et al., 2019), or adjusting the bias term of the output neurons (Pavlopoulos et al., 2020). However, there is an absence of comprehensive empirical studies that systematically compare different methods for the class imbalance problem for abusive language detection. Our work closes this research gap and provides insights and guidelines for selecting suitable methods for a given setup.

Existing methods for mitigating the class imbalance issue can generally be categorized into data-level, model-level and hybrid methods. Data-level methods focus on utilizing data resampling or augmentation (Chawla et al., 2002; Han et al., 2005; Liu et al., 2009a; Yen and Lee, 2009; Zhang and Li, 2014), model-level techniques adjust the classification model to increase the importance of the minority class (Lawrence et al., 1996; Phan and Yamamoto, 2020; Lin et al., 2020; Li et al., 2020), while hybrid methods combine both data- and model-level techniques (Chawla et al., 2003; Guo and Herna L., 2004; Zhou and Liu, 2006; Buda et al., 2018). As the main contribution of this project, we conducted an extensive study to examine the effectiveness of popular techniques in resolving the class imbalance issue, specifically

Proceedings of the The 8th Workshop on Online Abuse and Harms (WOAH), pages 38–51

in abusive language detection. We explored 8 binary and multi-class datasets with varying degrees of imbalance ratios and diverse definitions of abusive labels. Additionally, based on observations of existing methods, as a secondary contribution, we propose two task-specific methods and evaluated their efficacy: augmenting texts of the minority class 1) with synonym replacement of abusive terms (AbusiveLexiconAug) and 2) with external datasets (ExternalDataAug). Our results suggest that random oversampling, focal loss (Lin et al., 2020) and AbusiveLexiconAug are applicable to the widest range of datasets, with focal loss being the most promising method to achieve the best model performance, albeit requiring careful hyperparameter tuning. We analyzed different aspects of the tested methods and datasets to provide useful insights and guidelines for practitioners in the field.

2 Related Work

2.1 Abusive Language Detection

Various datasets and approaches have been proposed for detecting abusive language (de Gibert et al., 2018; Davidson et al., 2017; Chen et al., 2012, inter alia). In terms of model architectures, most approaches involve fine-tuning Transformer-based models, such as BERT (Devlin et al., 2019). Except for artificially balanced datasets, most corpora contain the non-abusive class as the majority of the samples. Previous work attempted to solve this problem with several methods, including random sampling (Rizos et al., 2019), data augmentation with synthetic samples (Steimel et al., 2019) or back-translation (Al-Azzawi et al., 2023), adjusting the bias term of output neurons (Pavlopoulos et al., 2020) or using weighted cross-entropy (Das et al., 2021). However, most work only tests a few methods to mitigate class imbalance, and there is a lack of a systematic comparison.

2.2 Class Imbalance

Since many machine learning tasks are affected by this problem, various approaches have been proposed to solve it. We can categorize these approaches into three groups: data-level, model-level and hybrid methods. We refer to (Krawczyk, 2016; Johnson and Khoshgoftaar, 2019; Kaur et al., 2019; Henning et al., 2023) for comprehensive surveys. Our primary objective in this study is to provide practical insights and guidance for researchers when confronting the class imbalance problem, specifically in the abusive language detection task.

2.2.1 Data-level Methods

The general idea is to preprocess the training data to reduce the imbalance among different classes. Popular methods include resampling and text augmentation. Resampling mainly involves manipulating the class distributions of the initial training sets. The most fundamental versions of the resampling strategy are random over- and undersampling, which involve making copies of minority and deleting majority samples to balance the class distribution. Experimental results in (Buda et al., 2018; Padurariu and Breaban, 2019) showed that random oversampling is the best method for addressing the imbalance issue in most circumstances. Liu et al. (2009b) showed that deleting some majority class samples can lead to a performance drop and proposed two methods, EasyEnsemble and BalanceCascade to mitigate this issue by combining multiple models trained on different subsets of the original data. Estabrooks et al. (2004) conducted comparative experiments with both resampling methods on medical image data, concluding that oversampling and undersampling can have equivalent performance, and there are no obvious optimal resampling ratios for either of the strategies. We also experimented with random over- and undersampling in our study.

Text augmentation includes methods for increasing the diversity of training texts without explicitly collecting new data (Feng et al., 2021; Bayer et al., 2022). Representative strategies can be categorized into three parts: rule-based, instance interpolation-based and model-based. Rule- and model-based methods are mainly implemented with text replacement, deletion, and insertion operations, while interpolation-based approaches combine two real samples to synthesize a new one. Rizos et al. (2019) proposed three techniques, including synonym replacement, to reduce the degree of class imbalance in abusive datasets and achieved significant F_1 improvements on a selection of neural architectures. In our study, we compare the effectiveness of the text augmentation method implemented by token-level synonym replacement based on different replacing strategies. We also proposed two innovative augmentation methods with abusive lexicons and external abusive texts.

2.2.2 Model-level Methods

To address the negative influence of the imbalance in the original training data, adjustments can be made to the classification models. There are two main approaches: threshold-moving and loss function modifications. Threshold-moving (also known as thresholding or post-scaling) is applied only during inference time by moving the classification threshold toward minority classes so that they are more likely to be predicted. Among the different variants (Lawrence et al., 1996; Zhou and Liu, 2006; Tian et al., 2020), one of the most basic versions is to compensate for prior class probabilities (Richard and Lippmann, 1991). Due to no hyper-parameter tuning requirements, we test this method in our work.

The widely used cross-entropy (CE) loss grants equal importance to each class without taking their numbers of samples into account. A simple modification of the CE loss is to add a class weight coefficient so that all classes make the same contribution to the weight optimization (Phan and Yamamoto, 2020). Lin et al. (2020) further pointed out that the hard, misclassified samples are suppressed by easy-to-classify samples during training and presented focal loss (FL) to increase the importance of misclassified samples. Li et al. (2020) held the view that the CE loss is accuracy-oriented and thus not optimal for improving the F_1 scores for the classification of imbalanced datasets. They introduced the dice coefficient as the harmonic mean of precision and recall to minimize the gap between the training objective and the evaluation metrics. In our study, we mainly focus on the weighted cross-entropy loss and the focal loss.

2.3 Hybrid Methods

It is also possible to combine multiple types of methods. Based on the observations that oversampling and undersampling are both useful to some degree, Estabrooks et al. (2004) designed a combination scheme to jointly employ results from multiple oversampling and undersampling classifiers. Buda et al. (2018) found that thresholding worked well together with oversampling for image data. Inspired by their work, we experimented with the combination of over- and undersampling.

3 Methods

In this section, we first provide a formal definition of the label imbalance problem, followed by a discussion of the methods that were investigated in our work. With our method selection, our aim is to focus on approaches that are widely used and easy to implement in real-world applications. In this way, we expect our conclusions to be practical and valuable to practitioners.

Given an abusive dataset of N text samples denoted as $\boldsymbol{D} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ and a set of labels C, where $y_i \in C$ indicates whether a sample x_i is non-abusive or belongs to a certain subtype of abusive language (offensive, sexist, etc).[1] We denote N_c as the number of samples in a class $c \in C$. Due to the existence of more non-abusive speech than abusive speech on the Internet, we have an uneven distribution of N_c among different classes. We define the imbalance ratio ρ, as the ratio between the maximum number and the minimum number of texts among all the classes: $\rho = N_{c_{\text{max}}}/N_{c_{\text{min}}}$, with $c_{\text{max}} = \arg\max_{c \in C} N_c$, $c_{\text{min}} = \arg\min_{c \in C} N_c$.

3.1 Data-Level Methods

3.1.1 Random Sampling

With random sampling methods, we attempt to adjust our training set such that a certain class is distributed against other classes with a desired imbalance ratio (ρ') for re-sampled data.

Random Oversampling (ROS) In ROS we randomly pick a text from the minority classes and duplicate it to achieve the desired imbalance ratio. After applying ROS, a class c will be represented with $N'_c = \frac{N_{c_{\text{max}}}}{\rho'}$ examples, if $N_c < N'_c$.

Random Undersampling (RUS) Contrary to ROS, we randomly delete certain numbers of texts from a majority class to obtain an expected distribution among classes. After RUS, a class c is expected to only contain $N'_c = N_{c_{\text{min}}} \cdot \rho'$ examples, if $N_c > N'_c$.

Hybrid Sampling (Combi RS) We also combine ROS and RUS to filter texts from majority and duplicate minority classes to obtain an balanced distribution with $\rho' = 1$. To this end, we first choose a resampling percentage p. A resampled dataset with $|C|$ classes will have a total number of $N' = p \cdot N$ samples, with $N'_c = \frac{N'}{|C|}$ samples in class c. Then, we randomly selected N'_c samples from each class with replacement. In view of the choice of $p = \frac{|C| \cdot N_{c_{\text{min}}}}{N}$ resulting in all the classes

[1] We focused on single-label classification in this work.

undersampled to $N_{c_{\text{min}}}$ samples, and $p = \frac{|C| \cdot N_{c_{\text{max}}}}{N}$ leading all the classes to be oversampled to $N_{c_{\text{max}}}$ samples, we tuned the resampling percentage p within the range of $\in (\frac{|C| \cdot N_{c_{\text{min}}}}{N}, \frac{|C| \cdot N_{c_{\text{max}}}}{N})$.

3.1.2 Text Augmentation

Instead of simply duplicating samples as in ROS, we augment texts from the minority class by replacing words with their synonyms to obtain an expected imbalance ratio. We test a technique based on contextual embeddings for word replacement:

BERTAug Similarly to random oversampling, we randomly pick texts from the minority classes to achieve the desired imbalance ratio. However, instead of simply duplicating the selected samples, we use them to generate new samples by replacing some of the words in them. To this end, we randomly mask aug_p percentage of the words in a given input and feed the surrounding tokens to HateBERT[2] (Caselli et al., 2021) to find the top_k most suitable replacements at each masked position. New samples are generated by randomly sampling a token for each masked position from the top_k candidates. We tune the values of aug_p, top_k and ρ'.

3.2 Model-Level Methods

Threshold-Moving (TM) Adjusting the threshold of the decision boundary allows us to prioritize the underrepresented classes. An effective approach that works well for various tasks is to compensate for the imbalance with the prior probability of the classes (Buda et al., 2018). Instead of adjusting the actual decision threshold, we adjust class probabilities at inference time as:

$$\tilde{p}(y_i = c|x_i) = \frac{p(y_i = c|x_i)}{p(y_i = c)}, \quad (1)$$

where $p(y_i = c) = \frac{N_c}{N}$. We do not use the development nor the test set to tune the adjustment.

Weighted Cross Entropy (Weighted CE) Instead of adjusting the prediction as in TM, weighted CE accounts for label imbalance during model training. The standard loss function for classification tasks is cross-entropy:

$$L_i = -\sum_{c \in C} \delta(y_i, c) \log p(y_i^* = c), \quad (2)$$

where y_i^* is the predicted class of sample $i \in \{1, \ldots, N\}$, and $\delta(\cdot, \cdot)$ is 1 in case of equal parameters and 0 otherwise. This form assigns the same importance to all the classes, meaning the contribution of each class to the loss is greatly affected by the number of samples, i.e., minority classes are suppressed when the imbalance ratio is large. To mitigate this issue, we leverage a weight for each class to balance their influence. The class weight α_c for a class c can be either a fixed number proportional to the training set distribution defined as $\frac{1}{N_c}$ or a hyperparameter to be tuned during training. We compared the performance of both settings. The weighted CE loss is thus defined as:

$$\tilde{L}_i = -\sum_{c \in C} \alpha_c \delta(y_i, c) \log p(y_i^* = c). \quad (3)$$

Focal Loss (FL) In contrast, FL aims to differentiate between *hard* and *easy* texts. Easy-to-classify samples may result in a low loss value, causing premature stopping, while hard samples are still not correctly classified. To address this issue, Lin et al. (2020) proposed FL by introducing a modulating term to the CE loss to make the loss function focus more on hard and misclassified samples. This is particularly beneficial for minority classes, which are usually harder to learn compared to the majority classes. With FL, the majority class is gradually down-weighted, so that the minority class can be further improved. FL is defined as:

$$FL_i = -\sum_{c \in C} \delta(y_i, c)(1 - p(y_i^* = c))^\gamma \log p(y_i^* = c), \quad (4)$$

where γ is a modulating factor. With $\gamma = 0$ focal loss degrades to the original CE loss. When $\gamma > 0$, misclassified samples with a small probability ($p(y_i^* = c)$) have a scaling factor near 1, and their losses remain unaffected. However, for well-classified samples with a probability close to 1, the scaling factor approaches 0 and the loss is down-weighted.

Weighted Focal Loss (Weighted FL) As proposed by Lin et al. (2020), we can apply an α-balanced focal loss in practice:

$$\tilde{FL}_i = -\sum_{c \in C} \alpha_c \delta(y_i, c)(1 - p(y_i^* = c))^\gamma \log p(y_i^* = c). \quad (5)$$

[2]We choose HateBERT over a plain pre-trained BERT model because it is a re-trained BERT model on a Reddit abusive dataset is the same domain what we are working on.

4 Our Methods

Although ROS and RUS improve class imbalance, ROS can lead to overfitting if samples are duplicated too many times, while RUS removes valuable information. Naive data augmentation methods try to enrich the training data with new information (words), however efficacy on abusive datasets is limited, since most of the randomly replaced words are not abusive. Considering these disadvantages, we propose two new abusive language detection-specific data augmentation methods.

ExternalDataAug Instead of simply duplicating samples as in ROS, we augment a certain class in the training data with texts from another abusive dataset bearing classes with analogous definitions. In this way, we can improve the distribution of the minority classes and provide more abusive information at the same time without sample duplication. For each minority label, we randomly choose a subset from one or more suitable datasets to reach a desired imbalance ratio ρ', as in ROS. For minority labels that do not have enough external data to augment, we use ROS to oversample them. We provide details of the combined datasets and classes in Appendix A.1.

AbusiveLexiconAug Since BERTAug chooses words to be replaced randomly, it fails to introduce new informative words regarding abusive classification. Therefore, we turn to an abusive lexicon, which we use to find abusive words to replace in the inputs, as well as to select replacements from. As the lexicon, we leverage a combination of the following existing lexicons: 1) *English swear words on Wiktionary*[3] with 60 words; 2) *English profanity on Wiktionary*[4] with 55 words; 3) Multilingual Offensive Lexicon (Vargas et al., 2021) with 610 terms; 4) Hate Speech Lexicon (Davidson et al., 2017) with 178 terms; 5) Lexicon of Abusive Words (Wiegand et al., 2018) with 2858 unique abusive words, resulting in a lexicon of 3331 distinct abusive terms. Given an input sample, we choose aug_p percentage of terms that are contained in the abusive lexicon, and look for their top_k most similar pairs in the lexicon based on the similarities of their FastText embeddings[5] (Bojanowski et al., 2017) using cosine similarity. To generate a new input text, we replace the selected words by sampling from their top_k pairs. We generate a new training dataset with a desired imbalance ratio ρ'.

[3] https://en.wiktionary.org/wiki/Category:English_swear_words

[4] https://en.wikipedia.org/wiki/Category:English_profanity

[5] We use FastText instead of BERT embeddings to find top_k replacements of a given word, since we have no context for lexicon entries which is needed for the latter model.

Dataset	#Texts	Label Distributions (%)		ρ	Source
Twitter-Hate-Speech	31,962	Non-Hate 93%	Hate 7%	13.3	Twitter
Civil-Comments	5,000	Non-Toxic 92%	Toxic 8%	11.5	Civil Comments
Gibert-2018	10,703	Non-Hate 89%	Hate 11%	7.9	Stormfront
US-Election-2020	3,000	Non-HoF 88%	HoF 12%	7.5	Twitter
CMSB	13,631	Non-Sexist 87%	Sexist 13%	6.5	Twitter
Founta-2018	46,452	Normal 72% Abusive 8%	Spam 16% Hateful 4%	20.3	Twitter
Davidson-2017	24,783	Offensive 77% Hate Speech 6%	Neither 17%	13.4	Twitter
AMI-2018	2,245	Discredit 51% Stereotype 14% Derailing 5%	Harassment 18% Dominance 12%	11.2	Twitter

Table 1: Statistics of the used datasets. The column ρ contains the imbalance ratios. HoF stands for hateful or offensive.

5 Experiments

5.1 Experimental Setup

As the basis of our classifiers, we used *bert-base-uncased* which we fine-tuned on the training set of the tested datasets using the following hyperparameters: number of epochs 10, learning rate 5×10^{-5} and weight decay 0.01. We test the mentioned label imbalance approaches by applying them in the fine-tuning phase (prediction phase in the case of TM), and compare them to the baseline using no such techniques. For implementation, we used the Huggingface library for modeling (Wolf et al., 2020) and the NLPAug (Ma, 2019) for text augmentation. All models were trained 3 times with different seeds. We used the macro F_1 score to compare the model performance with different methods, as it is frequently used for imbalanced datasets, including abusive language detection. We tuned hyperparameters on the validation sets. Trainer hyperparameters mentioned above were chosen based on the baseline model and the US Election-2020 dataset for simplicity. Only imbalance method specific hyperparameters, such as ρ or γ, were tuned for each approach, which we discuss below.

5.2 Datasets

We utilized multiple English datasets. Since some Twitter datasets had to be downloaded using tweet-IDs, the number of samples and the distribution of classes may differ from the original due to unavailability. Considering the main focus of our project is

Macro F_1 (%)	Binary Datasets					Multi-Class Datasets			Avg.	#+
	Twitter-Hate-Speech	Civil-Comments	Gibert-2018	US Election-2020	CMSB	Founta-2018	Davidson-2017	AMI-2018		
Baseline	87.21±0.55	75.99±0.46	76.89±0.70	75.62±1.53	84.36±0.53	62.70±0.91	74.70±0.59	54.65±2.35	74.02	
ROS	87.65±0.28	75.85±2.60	77.25±0.82	76.23±1.59	84.83±0.41	63.98±0.25	75.64±0.46	55.70±1.68	74.64	7/8
RUS	87.16±0.29	73.97±2.66	75.72±0.62	77.00±1.73	84.33±0.62	**64.38**±1.68	**76.57**±0.19	54.46±1.45	74.20	3/8
Combi RS	87.10±0.70	74.15±2.78	77.21±0.70	74.87±2.87	84.84±0.31	62.46±0.50	74.94±0.74	53.62±1.28	73.65	3/8
BERTAug	87.49±0.52	75.88±1.43	75.74±1.04	74.22±0.26	84.85±0.50	63.37±0.77	75.19±0.34	54.62±2.25	73.92	4/8
TM	86.18±1.10	75.27±2.12	77.11±0.97	**77.06**±2.34	**84.91**±1.12	61.90±0.35	74.33±0.91	53.83±0.79	73.82	3/8
Weighted CE	87.39±0.38	73.55±0.54	75.62±1.03	77.02±0.99	84.19±0.38	64.33±1.36	75.48±0.18	55.35±3.37	74.12	4/8
FL	**88.01**±0.63	76.75±0.91	**77.45**±0.34	74.44±2.76	84.72±0.49	63.55±0.50	74.74±0.86	**56.44**±0.76	74.51	7/8
Weighted FL	87.36±0.67	73.45±3.04	76.39±0.96	74.73±2.25	84.84±1.18	64.22±0.98	75.52±0.52	55.54±3.82	74.01	5/8
ExternalDataAug	87.16±0.45	**76.77**±3.04	75.85±0.45	-	84.59±0.58	64.20±0.82	73.71±0.50	-	-	3/6
AbusiveLexiconAug	87.36±0.54	75.67±0.96	77.25±0.22	73.81±0.24	84.59±0.46	63.51±0.45	76.05±0.06	55.61±1.31	74.23	6/8

Table 2: Macro F_1 scores (%) and standard deviation ($\pm$) of the tested methods on different datasets. We present the average performance in column *Avg.*, while column *#+* indicates the number of improved datasets compared to the baseline. For each column, the best scores in each method type (data-level, method-level, and our novel methods) are underlined and the highest overall scores are in bold. Systems achieving worse performance than the baseline are in gray. A $-$ indicates that the method is not applicable.

to analyze the effectiveness of various methods for label imbalance, we do not perform any preprocessing steps but rely only on the subword tokenizer of the used models. We perform a 60/20/20 random split on each dataset for training, validation, and testing, if the original dataset is not split for testing.

We experiment with 8 datasets, including 5 binary and 3 multi-class datasets, covering various types of abusive language, such as hate speech, offensive language, sexism, etc., as well as various sources from microblogging platforms (Twitter) to forums (Stormfront, Civil Comments). We refer to Table 1 for the list of used datasets and their statistics, such as label imbalance ratios. Dataset specifics are presented in Appendix A.

6 Results and Analysis

Our main results are presented in Table 2. In general, there is no single method that achieves the best performance on the majority of the datasets. Random oversampling (ROS), focal loss (FL) and our AbusiveLexiconAug method achieve better results than the baseline on most of the datasets. On binary datasets, model-level methods appear to be more effective than data-level methods, while for multi-class sets both methods exhibit comparable performance. On Civil-Comments, we found degraded performance with almost all the methods. We thus did a further investigation of this dataset in Section 6.1.

Our Proposed Methods AbusiveLexiconAug method shows promising improvements over existing methods, particularly when compared to BERTAug. It enriches the abusive information in the training set leading to these improvements. We anticipate further improvements in case a larger lexicon is available. Conversely, ExternalDataAug did not demonstrate sufficient efficacy, except on a limited number of datasets. We attribute this to potential dataset shifts being the main cause. Even though for each augmented dataset, we selected datasets with the most similar label definitions (as shown Table 6), it still introduces texts that are out-of-domain. To achieve further improvements, only external data from the same platform or domain should be utilized.

Data-Level Methods We found that for almost all the binary and small multi-class sets (AMI-2018), oversampling performs better than undersampling. However, on larger multi-class datasets (Founta-2018 and Davidson-2017), undersampling has better performance. The hybrid resampling method, Combi RS, tends to yield worse performance than over- and undersampling.

Model-Level Methods Other than focal loss being the most universally effective method in dealing with the class imbalance issue, we found that threshold-moving, which does not require hyperparameter tuning, is also quite effective on most binary datasets while achieving no improvements on multi-class datasets. On the contrary, weighted CE (with tuned class weights, as detailed in Appendix C) shows better performance on the multi-class sets compared to the binary sets. Weighted FL yields slightly better results on 4 out of 8 datasets when compared to FL.

6.1 Analysis

Sampling Ratio In ROS and RUS, a sampling ratio (ρ') has to be chosen. Figure 1 presents the

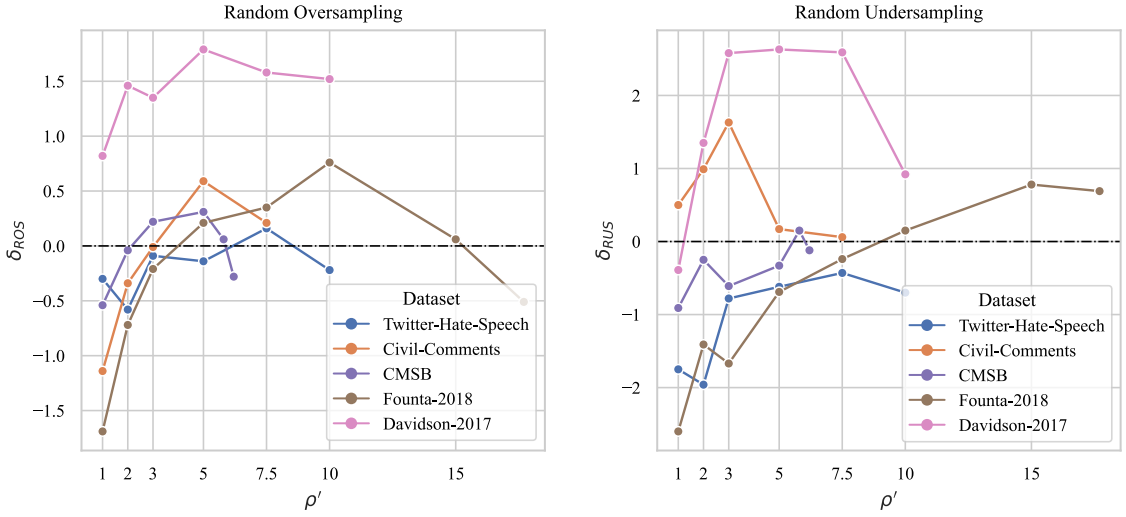


Figure 1: Macro F_1 scores of models with ROS/RUS with varying imbalance ratio ρ'. The y-axis δ_{ROS} = Macro F1$_{ROS}$ − Macro F1$_{Baseline}$ for a certain dataset, the same goes with RUS.

Dataset	#Texts	ρ	$\frac{\rho}{2}$	Actual best ρ'	
				ROS	RUS
Founta-2018	46,452	20.3	$\geq$ 10.2	10.0	15.0
Twitter-Hate-Speech	31,962	13.3	$\geq$ 6.6	7.5	7.5
Davidson-2017	24,783	13.4	$\geq$ 6.7	5.0	5.0
CMSB	13,631	6.5	$\geq$ 3.3	5.0	5.8
Gibert-2018	10,703	7.9	$\approx$ 4.0	3.0	5.0
Civil-Comments	5,000	11.5	$\leq$ 5.8	5.0	3.0
US-Election-2020	3,000	7.5	$\leq$ 3.8	2.0	6.1
AMI-2018	2,245	11.2	$\leq$ 5.6	3.0	3.0

Table 3: The best ρ' of ROS and RUS. A good starting point for ρ' is $\frac{\rho}{2}$, while the best value tends to be $\leq$, $\approx$ or $\geq$ based on the dataset size (threshold at 10,000). Exceptions are in red.

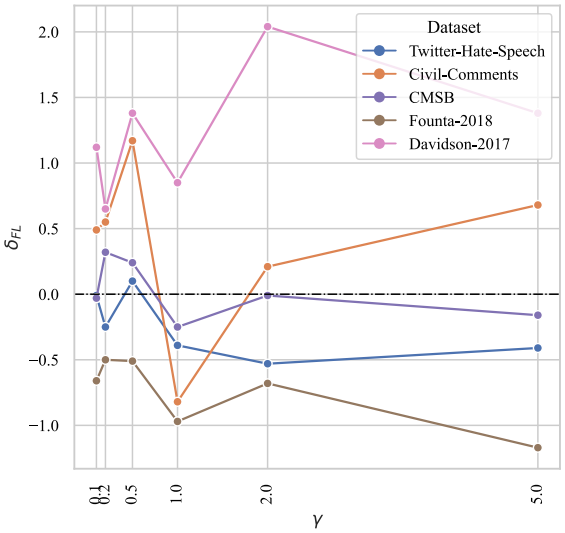


Figure 2: Model performance when employing Focal Loss with different γ to train the models. The y-axis δ_{FL} = Macro F1$_{FL}$ − Macro F1$_{Baseline}$ for a certain dataset.

model performance when applying different ρ' values. In the case of ROS when the value is close to 1, examples are duplicated too many times, leading to overfitting. Further analysis in Appendix B shows that on small datasets it is less likely to overfit than on large datasets. A large target ρ' close to the original imbalance ratio of a certain dataset is also not effective enough for improving performance. Similarly for RUS, we found that in most of our datasets, when ρ' is close to 1, i.e., perfect balance in the training set, too many samples are discarded and much information is lost, which leads to lower performance. Furthermore, we observed a decrease in F_1 scores when ρ' surpasses a certain threshold. There is a sweet spot for both methods, where the imbalance ratio is not too high to harm performance, but there aren't too many duplicates for the model to overfit (ROS), and it obtains enough information from the original training set (RUS) to classify the samples well.

According to our experiments, we found a general rule to estimate a good ρ' is to halve the original imbalance ratio of a certain dataset (Table 3). Further tuning of the value should be done around this half point to find the best value. However, our results indicate that for datasets of size at least 10 000, the best value is slightly higher (which means a lower amount of copied/deleted data), while for smaller datasets it tends to be lower than the half-point mark.

Tuning Focal Loss γ decides how much focus is put on the misclassified samples and the extent to

	Davidson-2017			
	Macro F1	Hate Speech	Offensive	Neither
Baseline	74.70	40.46	**94.54**	89.10
ROS	75.64	43.64	93.96	89.34
BertAug	75.19	41.84	94.51	89.22
AbusiveLexiconAug	**76.05**	**44.10**	94.49	**89.55**

Table 4: Macro and class-wise F1 scores when applying ROS, BertAug and AbusiveLexiconAug.

Macro F_1 (%)	Civil-Comments			
	#Texts=5k		#Texts=20k	#Texts=40k
	$\rho = 11.5$	$\rho = 7.5$	$\rho = 11.5$	$\rho = 11.5$
Baseline	75.99±0.46	78.95±2.16	79.19±1.24	79.22±0.55
ROS	75.85±2.60	80.30±0.76	79.07±1.56	79.65±0.85
RUS	73.97±2.66	81.46±1.53	78.73±1.70	79.21±0.35
TM	75.27±2.12	79.47±0.36	77.66±1.14	77.73±0.25
FL	76.75±0.91	79.05±0.42	78.83±1.31	78.85±0.82
ExternalDataAug	76.77±3.04	79.57±0.50	77.88±0.65	78.85±0.54
AbusiveLexiconAug	75.67±0.96	78.98±0.94	78.09±0.62	79.11±0.44

Table 5: Macro F_1 scores (%) and standard deviation ($\pm$) of the tested methods on variants of the Civil-Comments dataset. Systems achieving worse performance than the baseline are in gray. Standard deviations > 2 are marked in red, while the ones > 1.5 are in orange.

which well-classified samples are ignored. As seen in Figure 2, we found that smaller values of $\gamma \in \{0.1, 0.2, 0.5\}$ perform the best on the evaluated datasets, with 0.2 achieving the peak performance in most of the cases. We also further analyzed how the abusive class performance changes as γ increases in Appendix D.

In weighted FL, the best results on binary sets are obtained with larger γ compared to FL. Additionally, the class weight of the abusive class (which is always the minority class in binary sets) in the best setting of WFL is slightly smaller than the best choice in WCE. This is logical as the weights of the easy-to-classify classes are already reduced with γ, thus it does not need to put as much importance on the minority classes as in WCE, and vice versa. Note however, that WFL is the best among WCE, FL and WFL only in the case of CMSB dataset. In the case of multi-class sets, the same class weights perform the best for both WCE and WFL for all three datasets. While a larger γ (compared to FL) on Founta-2018 and AMI-2018 sets puts WFL in between of WCE and FL, a smaller γ in Davidson-2017 allows WFL to be the best among all model-based methods.

Augmentation with Abusive Lexicon vs. Bert As introduced in Section 3, ROS randomly duplicates samples and BertAug replaces random words in a sample, both do not enrich abusive information in the training data. In contrast, our AbusiveLexiconAug (Section 4) augments samples specifically with abusive terms. As shown in Table 2, BertAug did not achieve better results than ROS, but AbusiveLexiconAug yielded some improvements. Table 4 presents a comparison between the model performance when applying ROS, BertAug and our new method AbusiveLexiconAug. F1 scores for the minority abusive class (*Hate Speech*) are greatly improved with the abusive lexicon. This indicates that our strategy to focus on the abusive terms of a text and augment them is quite effective in providing models with more information about various abusive categories. In terms of hyperparameters, we find that it is better to use a value of $aup_p = 0.1$ in the case of BertAug, while a value between $aug_p = 0.1$ or 0.3 works best for AbusiveLexiconAug.

Challenges with Small Datasets As analyzed in Figure 4a, a substantial standard deviation of the results of models with different seeds is observed in several datasets: Civil-Comments, US-Election-2020, AMI-2018. These datasets are all of a relatively small scale with a total number of texts $N \leq 5,000$ (Table 1). Our advice when dealing with small datasets is to conduct more experiments with different seeds to acquire unbiased results since they are highly sensitive to any changes in the models.

A particularly challenging dataset is the Civil-Comment (CC), as most of our methods did not achieve better results than the baseline on it. We thus explore the potential causes in terms of data sizes and imbalance ratios. As mentioned, we used a 5k-sample subset with an imbalance ratio $\rho = 11.5$ of the full dataset in our main experiments. For comparative experiments, we resampled another 5k-sample set with an imbalance ratio $\rho = 7.5$, and 20k-/40k-sample sets with an imbalance ratio $\rho = 11.5$. We then conducted experiments with our best methods and methods with which the main CC results (Table 2) have large standard deviations. Results are shown in Table 5. As the results show, with larger data sizes or with a smaller imbalance ratio, the standard deviations are reduced (Figure 4b). Interestingly, we see a substantial performance improvement on the subset with a smaller imbalance ratio $\rho = 7.5$, in comparison to the setups with considerably increased data sizes (#Texts=20k/40k). These findings further support our suggestion above that more rigorous experimentation is needed in the case of small and/or

largely imbalanced datasets.

7 Conclusions and Final Suggestions

In this study, we investigated four data-level and four model-level strategies for addressing the class imbalance problem in abusive language detection. As secondary contributions, we proposed two novel methods, ExternalDataAug and AbusiveLexiconAug, to compensate for the limitations of existing methods. We evaluated the effectiveness of these methods across a diverse set of datasets. Our experiments demonstrated that AbusiveLexiconAug and focal loss consistently delivered strong performance over all datasets. However, no single method emerged as the clear winner out of the tested methods and experimented methods did not significantly boost model performance. Thus, we outline our key findings for practitioners seeking the most suitable solution for their specific task:

1. Random oversampling, focal loss and AbusiveLexiconAug are the safe first choices for various abusive datasets. However, tuning their parameters is suggested. Further options also include a combination of these methods.

2. Focal loss is the most effective model-level approach. Weighted focal loss is likely to further improve performance with proper weights. For multi-class datasets, weighted cross-entropy loss is also a good choice.

3. In terms of augmentation methods, using synonym augmentation with an abusive lexicon (our AbusiveLexiconAug) brings an overall enhancement to the model performance compared to methods that replace randomly chosen words.

4. Random undersampling, can achieve high performance, but only if a large training dataset is available, with some exceptions.

5. Datasets with a small number of training samples ($N \leq 5,000$) are extremely sensitive. In this situation, we suggest a rigorous search for the best method and parameters, starting with focal loss, or AbusiveLexiconAug to add more information to the training set.

8 Limitations

Although we tested on 8 datasets, we only included English corpora in this study. We believe that our findings are valid for other languages as well, however we leave such experiments for future work. Similarly, we selected the most popular approaches from data-level, model-level and hybrid approaches, but we were not able to test all previously proposed methods. In future work, we are interested in approaches tailored specifically for the abusive language detection task. Out of practical values, we experimented only on BERT with a classification head, but it's also worth exploring other classifiers in the future work.

Acknowledgements

We thank the anonymous reviewers for their helpful feedback. The work was funded by the European Research Council (ERC; grant agreement No. 640550) and by the German Research Foundation (DFG; grants FR 2829/4-1 and FR 2829/7-1).

References

Sana Al-Azzawi, György Kovács, Filip Nilsson, Tosin Adewumi, and Marcus Liwicki. 2023. NLP-LTU at SemEval-2023 task 10: The impact of data augmentation and semi-supervised learning techniques on text classification performance on an imbalanced dataset. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1421–1427.

Markus Bayer, Marc-André Kaufhold, and Christian Reuter. 2022. A survey on data augmentation for text classification. *ACM Comput. Surv.*, 55(7).

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Uwe Bretschneider, Thomas W. Wöhner, and Ralf Peters. 2014. Detecting online harassment in social networks. In *International Conference on Interaction Sciences*.

Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259.

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.

Nitesh Chawla, Aleksandar Lazarevic, Lawrence Hall, and Kevin Bowyer. 2003. Smoteboost: Improving

prediction of the minority class in boosting. In *Proceedings of the 7th European conference on principles and practice of knowledge discovery in database*, pages 107–119.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80.

Mithun Das, Somnath Banerjee, and Punyajoy Saha. 2021. Abusive and threatening language detection in urdu using boosting based and bert based models: A comparative approach. In *Fire*.

Thomas Davidson, Dana Warmsley, Michael W. Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *International Conference on Web and Social Media*.

Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate speech dataset from a white supremacy forum. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 11–20.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186.

Andrew Estabrooks, Taeho Jo, and Nathalie Japkowicz. 2004. A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence*, 20.

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. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988.

Antigoni Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. *Proceedings of the International AAAI Conference on Web and Social Media*, 12(1).

Lara Grimminger and Roman Klinger. 2021. Hate towards the political opponent: A Twitter corpus study of the 2020 US elections on the basis of offensive speech and stance detection. In *Proceedings of the Eleventh Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 171–180, Online.

Hongyu Guo and Viktor Herna L. 2004. Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach. *SIGKDD Explor.*, 6:30–39.

Hui Han, Wenyuan Wang, and Binghuan Mao. 2005. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing*.

Sophie Henning, William Beluch, Alexander Fraser, and Annemarie Friedrich. 2023. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 523–540.

Justin Johnson and Taghi Khoshgoftaar. 2019. Survey on deep learning with class imbalance. *Journal of Big Data*, 6:27.

Harsurinder Kaur, Husanbir Singh Pannu, and Avleen Kaur Malhi. 2019. A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Comput. Surv.*, 52(4).

B. Krawczyk. 2016. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5:221–232.

Steve Lawrence, Ian Burns, Andrew D. Back, Ah Chung Tsoi, and C. Lee Giles. 1996. Neural network classification and prior class probabilities. In *Neural Networks*.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice loss for data-imbalanced NLP tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 465–476, Online.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2020. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327.

Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. 2009a. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.

Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. 2009b. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Cristian Padurariu and Mihaela Elena Breaban. 2019. Dealing with data imbalance in text classification. *Procedia Computer Science*, 159:736–745. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 23rd International Conference KES2019.

John Pavlopoulos, Jeffrey Scott Sorensen, Lucas Dixon, Nithum Thain, and Ion Androutsopoulos. 2020. Toxicity detection: Does context really matter? *ArXiv*, abs/2006.00998.

Trong Huy Phan and Kazuma Yamamoto. 2020. Resolving class imbalance in object detection with weighted cross entropy losses. *ArXiv*, abs/2006.01413.

Michael D. Richard and Richard Lippmann. 1991. Neural network classifiers estimate bayesian a posteriori probabilities. *Neural Computation*, 3:461–483.

Georgios Rizos, Konstantin Hemker, and Björn Schuller. 2019. 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*, CIKM '19, page 991–1000. Association for Computing Machinery.

Mattia Samory, Indira Sen, Julian Kohne, Fabian Flöck, and Claudia Wagner. 2020. "Call me sexist, but..." : Revisiting Sexism Detection Using Psychological Scales and Adversarial Samples. In *International Conference on Web and Social Media*.

Kenneth Steimel, Daniel Dakota, Yue Chen, and Sandra Kübler. 2019. Investigating multilingual abusive language detection: A cautionary tale. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1151–1160, Varna, Bulgaria. INCOMA Ltd.

Junjiao Tian, Yen-Cheng Liu, Nathaniel Glaser, Yen-Chang Hsu, and Zsolt Kira. 2020. Posterior recalibration for imbalanced datasets. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20.

Francielle Vargas, Fabiana Rodrigues de Góes, Isabelle Carvalho, Fabrício Benevenuto, and Thiago Pardo. 2021. Contextual-lexicon approach for abusive language detection. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1438–1447, Held Online.

Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. 2018. Inducing a lexicon of abusive words – a feature-based approach. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1046–1056.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Show-Jane Yen and Yue-Shi Lee. 2009. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, 36(3):5718–5727.

Huaxiang Zhang and Mingfang Li. 2014. Rwo-sampling: A random walk over-sampling approach to imbalanced data classification. *Inf. Fusion*, 20:99–116.

Zhi-Hua Zhou and Xu-Ying Liu. 2006. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering*, 18(1):63–77.

A Datasets

We provide further information about the used datasets below. Dataset statistics are presented in Table 1. Additionally, we discuss the combined datasets in our ExternalDataAug method at the end of this section.

Twitter Hate Speech Dataset (Twitter-Hate-Speech) was constructed for a practice problem on Analytics Vidya[6] for better detection and moderation of hate speech on Twitter. We only used the labeled training set, since the test set is not available.

Kaggle Toxic Comment Classification Challenge (Civil-Comments) is a multi-label dataset[7] used to identify and classify various types of toxic online comments. We utilized the toxic score in the dataset to obtain binary data. We randomly sampled a subset of 5,000 due to limited computational resources.

Stormfront Hate Speech Dataset (Gibert-2018) is a hate speech dataset collected from the Stormfront white supremacist forum by de Gibert et al. (2018). We used only *hate* and *no hate* labels.

[6]Although the dataset is named sentiment analysis, it is about hate speech detection. https://datahack.analyticsvidhya.com/contest/practice-problem-twitter-sentiment-analysis

[7]https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data

Category	Dataset & Label
Hate	Twitter-Hate-Speech: Hate Gibert-2018: Hate Founta-2018: Hateful Davidson-2017: Hate Speech
Sexism	CMSB: Sexist AMI-2018: All 5 labels
Toxic	Civil-Comments: Toxic Founta-2018: Abusive Davidson-2017: Offensive
Non-Hate	Twitter-Hate-Speech: Non-Hate Gibert-2018: Non-Hate Davidson-2017: Neither

Table 6: Categories of labels from our datasets. A dataset and its specified class is used to augment the listed class of another dataset in the same cell.

Hate Speech in US 2020 Elections (US-Election-2020) is a binary set of tweets collected by Grimminger and Klinger (2021) during the US 2020 Election to examine whether supporters of Biden and Trump communicate in a hateful and offensive manner.

Sexism Detection (CMSB) is a binary dataset created by Samory et al. (2020), combining four existing datasets to detect subtle and diverse expressions of sexism.

Hate and Abusive Speech on Twitter (Founta-2018) is a fine-grained dataset by Founta et al. (2018) to study four types of abusive behavior on Twitter.

Hate Speech and Offensive Language on Twitter (Davidson-2017) is collected by Davidson et al. (2017) to better differentiate between serious hate speech and commonplace offensive language. We used its fine-grained labels.

Evalita 2018 Task on Automatic Misogyny Identification (AMI-2018) is a dataset for misogyny identification and categorization. We used its imbalanced find-grained set to categorize 5 misogynous behaviors.

A.1 ExternalDataAug

As discussed in Section 4, instead of simple oversampling, we augment the minority classes of a given training dataset with texts from external datasets. To find a suitable augmentation source for each label in our data, we examined the definitions of all the labels and grouped them into 4 categories as presented in Table 6. Classes from a specific dataset within the same category is thus used as augmentation sources for each other.

B Overfitting in ROS

We checked the performance correlation between the evaluation and training splits when using different target ρ' values. We observed that in the case of small datasets (US-Election-2020 and AMI-2018) the validation and train scores positively correlate. However, as shown in Figure 3, for large or highly imbalanced sets, when the performance on the training set improves with a smaller ρ' value, we see a reduction in validation scores, indicating overfitting. Nevertheless, overfitting has to be handled with care when applying ROS using a suitable imbalance ratio.

C Class Weights in Weighted CE

In weighted CE, class weights $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{|C|})$ determine how much importance we assign to each class. As discussed by Lin et al. (2020) and Li et al. (2020), $\boldsymbol{\alpha}$ can be either obtained directly from training set distributions or as a hyperparameter to tune. We thus would like to determine which option is better. Table 7 presents how the overall and label-wise macro F1 scores change when applying different $\boldsymbol{\alpha}$. We observe that a larger α_c increases the performance for a specific class, but after it surpasses a certain threshold, it harms both the overall and the performance on class c. To choose the best $\boldsymbol{\alpha}$, we conclude that although the class weights $(\frac{1}{N_c})$ from the training set on binary datasets do not guarantee the best model performance, they can ensure decent macro F1 scores. With a slight adjustment based on this, we can achieve the highest macro F1 scores. The same rule applies to multi-class datasets. We can see from table (b) that a class weight of 1.2 does not obtain the highest F1 score for the class *hate speech*. Rather, we need to consider other classes when assigning weights in multi-class sets. A slightly deviated version of the weights $(0.9, 0.1, 0.4)$, which increases and decreases the portions of certain classes in a minor way, while keeping the relative proportion of different classes, yields the best model performance.

D Focal Loss with Varying γ

Although focal loss brings improvements in the overall macro F1 scores on almost all of our datasets, we observed that some datasets are sensitive to varying γ and larger values do not guaran-

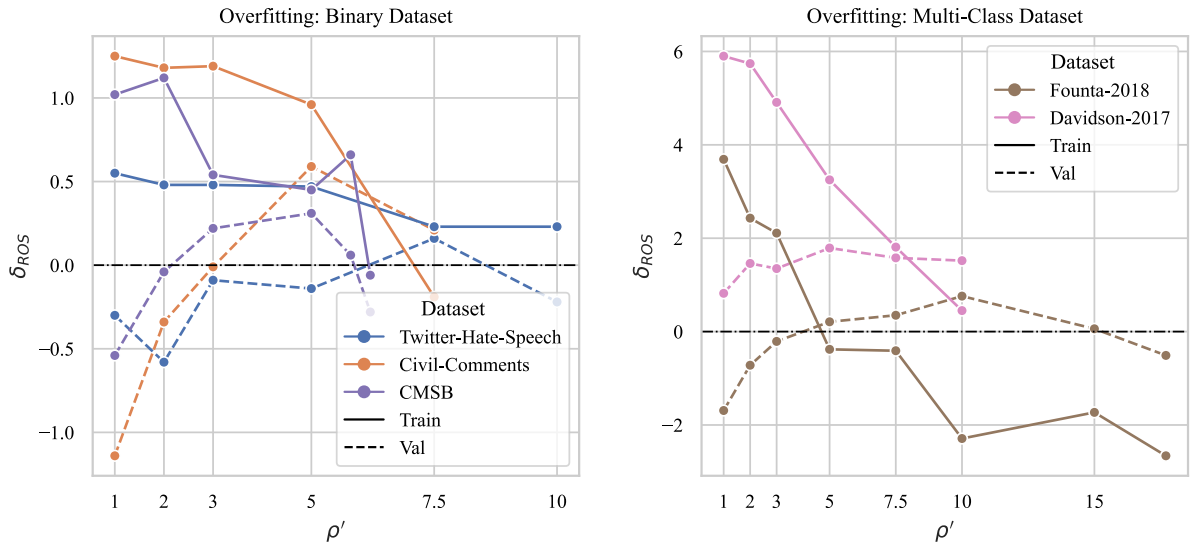


Figure 3: Correlation between training and validation performance when applying varying ρ' in ROS.

α_+	Twitter-Hate-Speech		
	Macro F1	Non-hate	Hate
0.1	86.94	**98.37**	75.51
0.25	87.21	98.31	76.10
0.75	**87.60**	98.33	**76.88**
0.9	87.48	98.31	76.65
0.930	87.42	98.25	76.59
0.99	82.73	97.02	68.44

α_+	Gibert-2018		
	Macro F1	Non-hate	Hate
0.1	77.69	**95.40**	59.97
0.25	78.43	95.27	61.60
0.75	78.98	94.88	63.09
0.888	**79.72**	95.39	**64.04**
0.9	79.10	94.84	63.37
0.99	76.09	93.45	58.73

α_+	US-Election-2020		
	Macro F1	Non-hate	Hate
0.1	57.94	94.42	21.45
0.25	80.44	**95.76**	65.13
0.75	79.75	95.43	64.08
0.878	79.47	95.47	63.47
0.9	**81.16**	95.63	**66.70**
0.99	32.50	30.71	34.29

(a) Results on binary datasets.

α	Davidson-2017			
	Macro F1	hate speech	offensive	neither
(0.1, 0.7, 0.9)	68.48	23.35	**94.19**	87.90
(0.5, 0.6, 0.1)	74.08	41.61	94.01	86.62
(0.9, 0.1, 0.3)	**76.57**	**47.52**	93.67	**88.51**
(1.2, 0.1, 0.4)	75.85	46.08	93.58	87.89

(b) Results on multi-class datasets.

Table 7: Macro and label-wise F1 scores on the validation set when applying varying α for Weighted CE Loss. Class weights α calculated from training sets are underlined. Best α (**bolded**) is selected based on the highest validation macro F1 scores.

tee a more significant punishment on non-abusive class, nor a greater improvement on the abusive classes that were not well classified. In Table 8 we present a comparison of two kinds of datasets when applying different γ. In table (a), we observe that as γ increases, initially both datasets achieve improved macro F1 scores, and then despite some decrease, the overall and label-wise scores do not vary significantly. On the contrary, there is a significant change (degradation) in model performance when γ increases on datasets presented in table (b). In general, we found that small datasets (US-Election-2020, AMI-2018) tend to be sensitive to varying values of γ.

E Standard Deviation

We provide a statistical analysis of the standard deviation of macro F1 scores in our experiments. From the box plots in Figure 4a, we can see that three datasets with $N \leq 5,000$ (Civil-Comments, US-Election-2020, and AMI-2018) have abnormal standard deviations with medians larger than 1.0 and relatively large spans of values. By comparing the standard deviations on variants of the Civil-Comments dataset in Figure 4b, we found that larger data sizes or smaller imbalance ratios both lead to smaller standard deviations. However, smaller datasets still tend to have higher standard deviations even with a smaller imbalance.

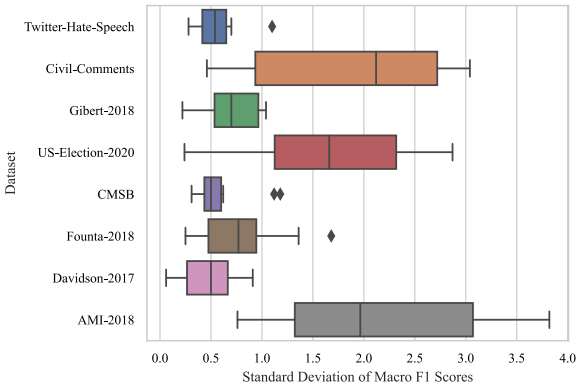
γ	Civil-Comments			Gibert-2018			Founta-2018				
	Macro F1	Non-Toxic	Toxic	Macro F1	Non-Hate	Hate	Macro F1	Normal	Spam	Abusive	Hateful
0.1	78.74	96.77	60.70	79.31	**95.60**	63.02	62.30	86.55	52.49	76.89	**33.27**
0.2	78.80	96.69	60.91	**79.70**	95.39	**64.02**	**62.46**	86.22	54.31	77.37	31.95
0.5	**79.42**	96.90	**61.94**	79.53	95.55	63.51	62.45	**86.98**	53.22	**77.90**	31.70
1.0	77.43	96.71	58.15	78.71	95.47	61.96	61.99	86.65	54.80	76.77	29.75
2.0	78.46	97.03	59.90	79.29	95.18	63.40	62.28	86.54	53.85	76.73	32.00
5.0	78.93	**97.18**	60.68	79.32	95.01	63.62	61.79	84.84	**54.97**	75.51	31.84

(a) Varying γ with moderately divergent model performance.

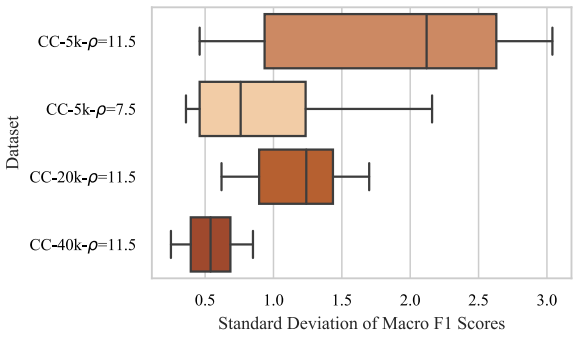
γ	US-Election-2020			AMI-2018					
	Macro F1	Non-HoF	HoF	Macro F1	Discredit	Stereotype	Dominance	Harassment	Derailing
0.1	**81.92**	95.56	**68.28**	52.93	75.64	45.58	35.45	54.40	53.60
0.2	81.60	**96.08**	67.11	**54.00**	76.50	47.97	33.89	52.37	**59.28**
0.5	80.13	95.77	64.49	52.46	76.90	**48.44**	**35.76**	51.19	50.02
1.0	80.78	95.08	66.48	51.67	77.12	47.95	35.19	50.88	47.21
2.0	77.01	95.57	58.46	49.45	76.47	39.76	29.53	**55.63**	45.86
5.0	78.14	95.01	61.27	50.11	**77.47**	39.21	26.72	53.39	53.54

(b) Varying γ with extremely divergent model performance.

Table 8: Macro F1 scores and label-wise F1 scores on the validation set when applying varying γ for Focal Loss. For each column, the highest scores are in bold, while lowest ones are in gray.



(a) Standard Deviations of our main experimental results in Table 2.



(b) Standard deviations of experiments on variants of the Civil-Comments (CC) dataset in Table 5.

Figure 4: Distribution of Standard Deviations.