Separating Hate Speech and Offensive Language Classes via Adversarial Debiasing

Shuzhou Yuan *
Karlsruhe Institute of Technology
shuzhou.yuan@kit.edu

Antonis Maronikolakis
CIS, LMU Munich
antmarakis@cis.lmu.de

Hinrich Schütze
CIS, LMU Munich

Abstract

Research to tackle hate speech plaguing online media has made strides in providing solutions, analyzing bias and curating data. A challenging problem is ambiguity between hate speech and offensive language, causing low performance both overall and specifically for the hate speech class. It can be argued that misclassifying actual hate speech content as merely offensive can lead to further harm against targeted groups. In our work, we mitigate this potentially harmful phenomenon by proposing an adversarial debiasing method to separate the two classes. We show that our method works for English, Arabic German and Hindi, plus in a multilingual setting, improving performance over baselines.

1 Introduction

Online hate speech has become a pernicious phenomenon of modern society and a lot of effort is being expended in tackling this challenge. While there has been plenty of work to develop automatic methods for hate speech detection (Schmidt and Wiegand, 2017), this has proven to be a difficult challenge to tackle with impractically poor performance.

In the NLP community, a prevailing convention is to frame this problem as a three-way classification: between *hate speech*, *offensive language* and *neither* (Davidson et al., 2017; Mulki et al., 2019; Founta et al., 2018; Mubarak et al., 2017; Mathur et al., 2018). While this convention allows for the application of more traditional NLP pipelines, performance has been low (Mozafari et al., 2019; Davidson et al., 2017) especially when it comes to generalization to unseen data (Swamy et al., 2019), with even humans struggling to distinguish hate speech (Chatzakou et al., 2017; Waseem, 2016).

In our work we also adopt the wide-spread 3-class definition of hate speech, where *hate speech* is defined as language used to express hatred towards a targeted group/individual based on protected attributes such as race or religion, *offensive language* contains offensive terms but is not targeting any group in particular, while *neither* is the case where none of the other two classes are present. As an example, in Table 1 we present the tweets that are annotated as hate speech and offensive language respectively, from Davidson et al. (2017), alongside `DistilBERT` predictions.

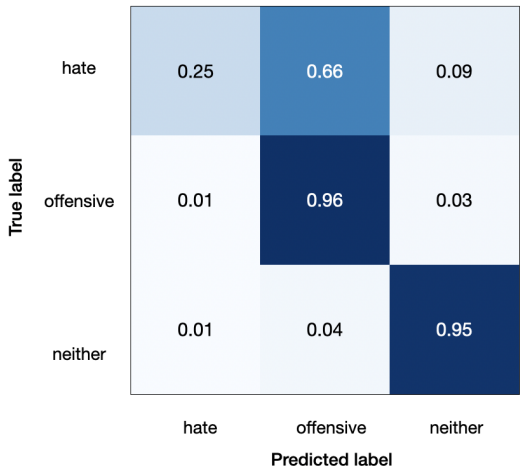


Figure 1: Performance of `BERT` on Davidson et al. (2017). We see the confusion between hate speech and offensive language, with numerous False Negatives. We argue that these are very insidious mistakes that could lead to further harm against target groups. With our adversarial debiasing method, we can separate these two classes further and thus minimize this type of error as well as increase overall performance.

An observation that can be made from the way classifiers operate is that oftentimes hate speech is misclassified as offensive language and vice-versa (Davidson et al., 2017; Mozafari et al., 2019). We showcase this in Figure 1. We argue that for hate speech detection models to be trustworthy, we need to work along two axes: increasing overall efficiency while keeping false negatives (i.e., hate speech marked as offensive language) to a mini-

* The work was done while at LMU Munich.

Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH), pages 1 - 10

tweet	class	prediction
bitch get off my twitter hoe	offensive	offensive
You ain't gunna do shit spear chucker	hateful	offensive
LMFAO I HATE BLACK PEOPLE	hateful	hateful

Table 1: Labeled tweets (from Davidson et al. (2017)) alongside model prediction.

mum, and that we cannot have one if it comes at the expense of the other. Thus, models should be able to tackle both challenges simultaneously. If models have low accuracy, they will not find application in real-world settings, and if there is too much hate speech content slipping through the cracks, targeted communities will experience increased harm online.

With that goal in mind, we propose a novel method to separate the two classes (*hate speech* and *offensive language*) via adapting adversarial debiasing methods to the problem at hand. This model stabilizes and improves classifier behavior and performance, increasing metrics across all classes, while at the same time keeping performance for hate speech content stable (or improving upon it). We thus strike a balance between performance overall and specifically for hate speech.

We experiment with different architectures for the classifier (`BERT` (Devlin et al., 2019), `DistilBERT` (Sanh et al., 2020)) and the adversary (`BERT`, `DistilBERT` and LSTMs (Hochreiter and Schmidhuber, 1997)). We perform hyperparameter tuning on English data before applying our findings on several languages (German, Arabic and Hindi) diverse in script, typography and grammar, as well as on a multilingual task setting using `mBERT`. To more objectively frame the benefit of our method, we compare against a battery of baselines, while we also perform error analysis to identify patterns where our method helps.

In summary, our contributions[1] are: **i)** Employing adversarial debiasing to separate hate speech and offensive language **ii)** Showing that our method works in keeping false negatives to a minimum and increasing F1-scores on multiple English datasets **iii)** Generalizing our findings to other languages, including a multilingual setting.

[1]Code available at `https://github.com/ShuzhouYuan/hate_speech_adversarial_debiasing`

2 Related Work

For hate speech detection, supervised learning approaches are often used. Schmidt and Wiegand (2017) provide a comprehensive survey on the earlier research of hate speech detection. In more recent work, focus has been placed on various classification methods and curation of datasets (Davidson et al., 2017; Wulczyn et al., 2017; Zhang et al., 2018; Mozafari et al., 2019; Qian et al., 2021).

In Davidson et al. (2017), the prevailing definition of the task as a three-way classification was formulated concretely. In their work, despite the high overall accuracy, over 30% of hate speech was misclassified as offensive language, which saliently sheds light on this pervasive challenge in hate speech detection. This finding was corroborated more recently in Mozafari et al. (2019), where the state-of-the-art `BERT` model (Devlin et al., 2019) was applied on a hate speech detection task, with over 60% of hate speech misclassified as offensive language. In the other direction, efforts have also been made to tackle false positives (Markov and Daelemans, 2021).

Further, recent efforts in hate speech detection have increased language coverage from English to multiple languages around the globe, including Hindi (Mathur et al., 2018), Arabic (Mubarak et al., 2017), Levantine (Mulki et al., 2019), Indonesian (Ibrohim and Budi, 2019), Danish (Sigurbergsson and Derczynski, 2020) as well as more general multilingual data (Ousidhoum et al., 2019; Ranasinghe and Zampieri, 2020; Basile et al., 2019) and code-mixing (Bohra et al., 2018).

A similar methodology to adversarial debiasing was applied to recidivism prediction (Wadsworth et al., 2018). There, racial biases existing in criminal history datasets were mitigated through adversarial training. This method was also applied in hate speech research to minimize bias against AAE text (Xia et al., 2020). In this case, adversarial debiasing was employed to counteract the disproportionate labeling of AAE text as offensive or hate speech. These works have shown the potential of adversarial debiasing methods in training fairer models.

3 Data

Since we wanted to evaluate the 3-class setting (*hate speech*, *offensive language* and *neither*), we either used datasets that already utilized these classes or equivalent ones (for example, in Founta

et al. (2018) *offensive language* is called *abusive language*). Overall, we made use of seven datasets. A summary of each dataset is presented in Table 2.

3.1 English

Davidson17. Davidson et al. (2017) is a well-studied English hate speech dataset collected from Twitter. It contains 25K tweets that are annotated as hate speech, offensive (but not hate) speech, or neither hate speech nor offensive language. The definition of hate speech and offensive language is the same as in §1. We utilize this dataset's development set for the early phase of experimentation to make design decisions, e.g. selecting model architectures, hyperparameters, baselines, etc.

Founta18. Founta et al. (2018) contains 100K English samples collected from Twitter. The definition of hate speech is the usual definition (as described in §1), while the *abusive language* class is defined as any impolite content using profanity, which is equivalent to the definition of offensive language. Thus, we regard it as offensive language for our experiments.

HasocEn19. Mandl et al. (2019) is an English hate speech dataset of 6K samples from Twitter and Facebook. The samples were labeled into four categories: *hate speech*, *offensive language*, *profanity*, and *normal*. *Offensive language* is defined as unacceptable language in the absence of insults and abuse. The *profanity* class expands on this definition to include swear words. We merged the two classes, because both classes meet our definition.

3.2 German

GermEval18. Wiegand et al. (2018) is a Twitter dataset containing 5K German tweets annotated as abuse, insult, profanity, and other/normal. The authors define the class *abusive* as behaviour that promotes dehumanization towards a target societal group or individual. Since it is as same as the aforementioned definition of hate speech, we rename it as hate speech in our research. *Profanity* is defined as text containing profane words and the class *insult* expresses a clear intention to insult or offend somebody. The two categories are merged into one class, *offensive language*.

HasocDe19. Mandl et al. (2019) is a 4K German dataset collected from Twitter and Facebook. The classes of *HasocDe19* are the same as *HasocEn19*: hate speech, offensive language, profanity, and normal. Similarly, the class *profanity* and *offensive language* are merged in our work.

3.3 Arabic

L-HSAB19. Mulki et al. (2019) contains 5K Arabic tweets. They were annotated as hate tweets, abusive tweets, and normal tweets. The definition of hate tweets is the same as our definition of hate speech in §1. The abusive tweets are defined as including offensive, aggressive or insulting language, which is equivalent to our definition of offensive language. We rename the class *abusive* as *offensive language* in our work.

3.4 Hindi

HasocHin19. Mandl et al. (2019) is a dataset of 5k samples written in Hindi. This dataset also comes from the *Hasoc* family of data, and therefore has the same classes: hate speech, offensive language, profanity and normal. As with the other two *Hasoc* datasets, the classes *offensive language* and *profanity* are merged into *offensive language*.

4 Adversarial Debiasing

In this section, we detail our adversarial debiasing scheme. In this setup two models are trained in conjunction: the classifier (predictor) and the adversary. The classifier is predicting the actual class of an example, while the adversary learns to predict a protected variable.

For the classifier, we compare the performance of three different models. And for the adversary, we investigate three different architectures, loss functions and protected variables[2]. In a first step, the models are trained and evaluated with *Davidson17*, *Founta18* and *HasocEn19* (the English datasets in our experiments).

4.1 Classifier

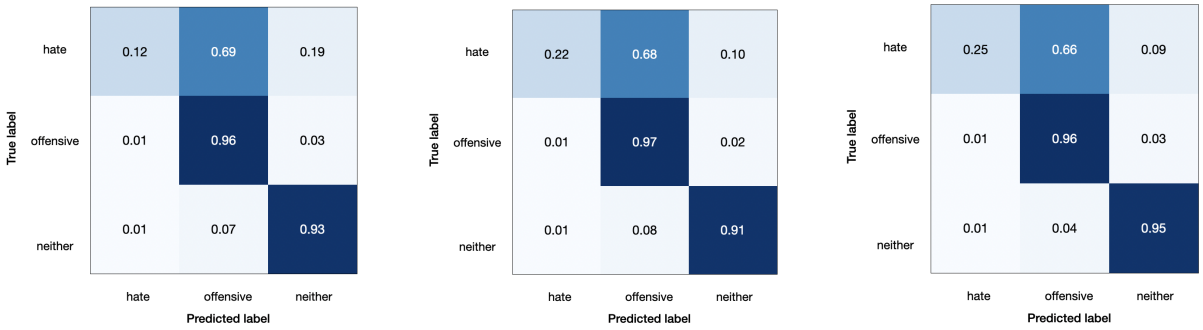
In the adversarial debiasing setting, the classifier is the component making predictions for the given task. The goal is to use the adversarial component to "debias" the classifier in order to achieve a desired result. In our case, our goal is to separate the *hate speech* from the *offensive language* class. We hypothesize this is going to improve performance. Here we explored `BERT`, `DistilBERT` and LSTM models for the classifier.

In our preliminary experiments (without adversarial debiasing), we found that LSTMs performed poorly in classifying hate speech. `BERT` and `DistilBERT` fared much better. All models, though, made a lot of false positive predic-

[2]In some papers it is called "protected attribute/label".

Language	Dataset	Domain	Classes: size	Source
English	Davidson17	Twitter	hate speech: 1431 offensive language: 19190 neither: 4163	Davidson et al. 2017
	Founta18	Twitter	hate speech: 4065 abusive (OFF): 17150 normal (NEI): 53851	Founta et al. 2018
	HasocEn19	Twitter, Facebook	hate speech: 1143 offensive ∪ profanity (OFF): 1118 none (NEI): 3591	Mandl et al. 2019
German	GermEval18	Twitter	abuse (HAT): 1022 insult ∪ profanity (OFF): 19190 Other(NEI): 3321	Wiegand et al. 2018
	HasocDe19	Twitter, Facebook	hate speech: 111 offensive ∪ profanity (OFF): 296 none(NEI): 3412	Mandl et al. 2019
Arabic	L-HSAB19	Twitter	hate speech: 417 abusive (OFF): 1559 normal (NEI): 3285	Mulki et al. 2019
Hindi	HasocHin19	Twitter, Facebook	hate speech: 556 offensive ∪ profanity (OFF): 1913 none (NEI): 2197	Mandl et al. 2019

Table 2: Summary of the datasets used in our research. HAT: hate speech, OFF: offensive language, NEI: neither



(a) LSTM (b) DistilBERT (c) BERT

Figure 2: Confusion matrices for different classifier models

tions, classifying hate speech as offensive language. Since `BERT` and `DistilBERT` had the highest true positive rate and both had similar performance, we chose to continue experimentation with `DistilBERT` to save on computational resources without a large performance drop. Confusion matrices for all three models are shown in Figure 2.

4.2 Adversary

The adversary in the setup is used to debias the classifier, learning to predict a particular attribute given the representations learned by the classifier. Then, via joint updating of weights, the classifier learns to generate representations that are not useful to the adversary, i.e., the goal is for the adversary to be unable to complete its task. We experiment with various protected variables, loss functions and architectures.

4.2.1 Adversary Architecture

The classifier we used in our final experiments was `DistilBERT`. Given some textual input, `DistilBERT` computes its internal representation which is then given as input to the adversary to predict the corresponding target. We experimented with two architectures for the adversary: Feed Forward Neural Networks (FFNs) and LSTMs. Accuracy for both adversaries was on average similar, plateauing around 75%. Since there is little difference in accuracy, we chose the FFN as our adversary since it requires fewer computational resources.

4.2.2 Protected variable and loss function

While in other research with adversarial debiasing (Xia et al., 2020; Sap et al., 2019; Han et al., 2021) has focused on debiasing for a protected variable (for example African American English), we instead propose a novel objective. In our exper-

iments the adversary learns to predict the offensiveness of a sample, either by separating it from hate speech or merging it (and thus separating *hate speech* and *offensive language* in the classifier).

Adversary predicts *hate speech* ∪ *offensive language* jointly (`Adversary`$_{joint}$).[3] Here the adversary is trying to jointly predict the *hate speech* and *offensive language* classes. Thus, we merge the two classes for the adversary's task, by labeling both classes as *offensive*. *Neither* is relabeled as *not-offensive*. In this case, the adversary learns to predict all hate speech and offensive language examples as one class from the representation of the classifier. Thus, since the goal is for the adversary to be unable to do so, the classifier learns how to *separate* these two classes. The loss function is defined as

$$loss_{total} = loss_{classifier} - \alpha * loss_{adversary}.$$

The loss function is the same as in Wadsworth et al. (2018); Xia et al. (2020), with $loss_{adversary}$ being the loss of the adversary for its task, $loss_{classifier}$ the loss of the classifier for the original task (hate speech vs. offensive language vs. neither) and α being a parameter to regulate the effect of $loss_{adversary}$. Xia et al. (2020) found that the value should be neither too large nor too small. Empirically, they set α=0.05. After some hyperparameter tuning, we found that in this setup an α value of 0.05 was the best-performing. Under $loss_{total}$, the classifier minimizes its original loss while maximizing the adversary's loss. As a result, the classifier is encouraged to actively develop diverging representations for the two classes.

Adversary discriminates between hate speech and offensive language (`Adversary`$_{sep}$). We also experiment with another adversarial setup: the adversary acts like "support", actively aiding the classifier in separating hate speech from offensive language. This is accomplished by employing an adversary that learns to model the "offensiveness" property, by discriminating between the *hate speech/neither* classes and *offensive language*. Since this method is aimed at directly helping the classifier, instead of subtracting this adversary loss, we *add* it instead:

$$loss_{total} = loss_{classifier} + \alpha * loss_{adversary}.$$

For this setup, we set the α hyperparameter to 2. The value of α was tuned on the development set of *Davidson17*, achieving the highest true positive rate for hate speech.

This "supportive" setup (discriminating between *hate speech* ∪ *neither* and *offensive*) was the best performing, so for the majority of our experiments we are using `Adversary`$_{sep}$.

Adversary predicts whether text contains swear words (`Adversary`$_{swear}$).[4] We also evaluated an adversary that predicts whether swear words are present in the text or not. We measured the proportion of *hate speech* and *offensive language* examples that contain a word from a dictionary of swear words[5] and found that in both classes more than 90% of examples contain at least one swear word. A lot of hate speech is labeled by annotators as such because of the presence of swear words in the text even when that should not be an indicator of hatefulness (Sap et al., 2019).

So, in this instance we train the adversary to predict whether swear words are present in text and then subtract this loss from the classifier's loss function. This forces the classifier to base its decisions on features other than the presence of swear words. The loss function is then

$$loss_{total} = loss_{classifier} - \alpha * loss_{adversary}.$$

4.3 Class Rebalancing

One thing to note is that data is heavily imbalanced against hate speech across all datasets (Table 2). For example, the number of offensive samples in *Davidson17* is 15 times higher than the number of hate speech samples. Before our adversarial debiasing experiments, we perform a study on the effect of imbalance on the training set of *Davidson17*. To compare against the original training set (denoted with *original dataset*), we sampled equally-sized sets from each class. Henceforth, we call this new, balanced dataset *uniform dataset*. Note that the development and testing sets remained unchanged for fair comparison: only the training sets were rebalanced. In Table 3 we see that the improvement of the true positive rate of hate speech is significant, from 22.0% to 81.8%. Although the overall accuracy drops by 16%, we believe this model would be more applicable in a real world scenario. If we build hate speech de-

[3] Even though this method did not work consistently, we mention it as a good starting point of discussion.

[4] This can only be applied in settings where swearword dictionaries are available, in our case we only applied it on the English datasets.

[5] https://www.cs.cmu.edu/~biglou/resources/bad-words.txt

tection models, we should be aiming for acceptable accuracy for the problematic class. Since we see that the uniform training set helps the model achieve acceptable performance for hate speech, we continue further experimentation using the uniform dataset.

Data	Best TPH	Accuracy
Original dataset	22.0%	91.2%
Uniform dataset	81.8%	75.2%

Table 3: Comparison of original and uniform dataset with *Davidson17*, evaluated on the same test set

5 Experimental Setup

For each dataset, we either use the provided training, development and testing set splits, or we sample them at 80:10:10 rates randomly. Then, we further downsample (to the number of examples in the smallest class according to each dataset) the training set classes to generate a uniform training set.

For *HasocEn19*, *HasocDe19*, *HasocHin19* and *GermEval18*, the dataset was already split in training and test sets. The original rates are presented in Table 4. In these cases, we keep the test set invariant and take 10% samples from the training set to form our development set.

Dataset	training:test
HasocEn19	84:16
HasocDe19	82:18
HasocHin19	78:22
GermEval18	59:41

Table 4: Original split distributions

For each experiment, we train for five epochs and keep the best-performing model across the epochs as evaluated on the development set. Then, we compute this model's performance on the held-out test set. We repeat this process three times and average the results.

Adversarial debiasing setup. The main setup we examine is $\texttt{Adversary}_{sep}$ where the adversary actively supports the classifier in separating the *hate speech* from the *offensive language* class, as defined in Section 4.2.2.

Multilingual dataset. To obtain a multilingual hate speech dataset, we combine all the datasets in Table 2 together. This new multilingual hate speech dataset contains 110k samples with the distribution of the three classes presented in Table 5.

Baselines. To evaluate the benefits of our method, we compare against vanilla finetuning with `DistilBERT` on each hate speech dataset as well as a simple class weighting baseline. We experimented with different weights and found that the best performing one (on the *Davidson17* dataset which served as an overall development set for design decisions) was $[1, 0.5, 1]$[6]. That is, we halve the weight of the *offensive* class.

6 Results

Results are summarized in Table 6. All the experiments are conducted on the uniform dataset, since the goal is to achieve an acceptable true positive rate (**TPH**: **T**rue **P**ositive rate for **H**ate speech) and this is a more solid starting point than the original distributions. We provide both macro and weighted F1 to show a more complete picture. Since our dataset is imbalanced, we focus on macro F1 for a more representative picture.

For *Davidson17* and *Founta18*, our method does not provide positive findings. In *Davidson17* both TPH and overall performance are lower, while in *Founta18*, adversarial debiasing does provide a performance boost, but it comes at the expense of TPH. For these two datasets, $\texttt{Adversary}_{swear}$ was applied as well, improving the TPH for *Davidson17* but not for *Founta18*.

Results are better for the final English dataset, *HasocEn19*. There, even though TPH drops substantially (10%), overall performance increases by more than 0.2 F1 points. Without an adversary, accuracy and F1-scores suffer, making for sub-par classifiers biased heavily towards hate speech. Instead, with our method, more separation is achieved and the model manages to separate the two contentious classes (*hate speech* and *offensive language*) with greater efficiency. Whereas before the classifier would not be practical due to low accuracy, with our method F1-scores and accuracy increase to acceptable levels.

HasocDe19 follows the same pattern, with the vanilla model being unable to provide strong overall results, instead becoming biased towards hate speech and dropping the rest of the classes. With our method, better balance is struck and we see an improvement of 0.15 F1-score over the vanilla model.

In *GermEval18*, we see stronger performance gains both for the TPH (+5.7% over the non-adversary model) and overall (+0.02 in F1-score). Even though the class-weighting baseline does

[6] [hate speech, offensive language, neither]

	Hate		Offensive		Neither	
	total	%	total	%	total	%
training set	8,042	7.3%	42,037	38.0%	60,604	54.8%
dev set	946	7.1%	5,204	38.3%	7,446	54.8%
test set	1,835	9.9%	5,925	32%	10,754	58.1%
uniform training set	8,042	33.3%	8,042	33.3%	8,042	33.3%

Table 5: Class distribution in the training, development and testing sets of the multilingual dataset

score higher in TPH, we note a drop in F1-scores and accuracy.

For *HasocHin19*, we observe the same pattern as the other *Hasoc* family of datasets, although to a lesser extent. In *L-HSAB19*, we show that the simple class-weighting baselines is better than both the vanilla and adversary models.

Finally, in the multilingual setting, we get mixed results. Compared to the vanilla model, adversarial debiasing offers a better TPH score with minimal drop in performance (while macro F1 increases by 0.02 too). Against the baseline model, while the baseline has a higher TPH, in all the other metrics performance is worse.

In synopsis, apart from *L-HSAB19*, performance is better when using the adversarial debiasing method, either for the TPH or the overall F1-score metrics. For the multilingual dataset, performance of our method is mixed, improving upon the vanilla model on the TPH and upon the baseline model on the other metrics.

All in all, our method manages to strike a better balance between TPH and overall performance and we thus believe these models are more applicable to a real-world scenario where both axes need to be taken into consideration.

7 Error Analysis

We observe that a few samples of hate speech are misclassified as offensive language by the vanilla model without the adversary, but correctly predicted by the adversarial model. In Table 7, we show six examples which indicate the significant improvement of adversarial models.

In English, we see that hateful speech was marked as merely offensive by the vanilla model, potentially because no slur was used, but only some offensive language ('c*nt', 'ass' and 'bad ass'). The model failed to take into account the context in which these words were used, or failed to pick up innocuous words used here as slurs (for example, 'orangutan'). The adversarial model was able to make correct predictions, potentially because it is not putting as much weight on individual words, but the combinations between them.

In German, the vanilla model's shortcomings are again centered around a lack of slurs. In both examples, there are no direct slurs so the model interprets it as offensive because of the overall negative sentiment (created through phrases such as 'böse Männer', meaning 'evil men' and 'sexuelle Gewalt', meaning 'sexual violence'). In one of the examples, the model misses that 'Froschfresser' (meaning 'frog eaters') is used as a slur. The adversarial model again shows an ability to expand from keyword-based predictions to a better understanding of context.

In Hindi, while the example is merely offensive, the vanilla model has marked it as hateful, potentially because of the politically heavy 'terrorist' term. The adversarial model has not put as much weight on the word and thus made a correct prediction. In Arabic, the vanilla model again misses that innocuous words are used as slurs (eg., 'dogs'), marking the text as offensive instead of hateful.

8 Conclusion

In hate speech detection efforts, it can be observed that a lot of classifiers struggle with the *hate speech* and *offensive language* classes. A lot of models trained on current datasets misclassify hate speech as offensive language. We argue that this type of error is particularly insidious, since it can lead to targeted groups getting exposed to harmful content more often. Further, a lot of hate speech classifiers are impractical, either having a low true positive rate for hate speech or low performance overall.

We propose a method to both increase the true positive rate for hate speech and to stabilize the classifiers in general. We base our method on the adversarial debiasing setup, where in our instance we are trying to support the classifier in separating the *hate speech* and *offensive language* classes.

We evaluate on seven hate speech datasets spanning four languages, plus a multilingual set we create by combining all data. Our method is at best performing just as well for all datasets ex-

Dataset	Experiment	Best TPH %	Overall Accuracy%	Macro F1	Weighted F1
Davidson17	Without adversary	77.96	**77.81**	**0.67**	**0.83**
	Adversary$_{sep}$	76.88	76.88	0.66	0.82
	Adversary$_{swear}$	**80.38**	75.1	0.66	0.81
	Baseline	78.77	72.64	0.64	0.79
Founta18	Without adversary	74.37	78.22	0.67	0.83
	Adversary$_{sep}$	68.74	**80.79**	**0.69**	**0.84**
	Adversary$_{swear}$	73.24	77.82	0.67	0.82
	Baseline	**77.57**	78.58	0.68	0.83
HasocEn19	Without adversary	82.53	39.99	0.37	0.32
	Adversary$_{sep}$	72.58	**47.53**	**0.47**	**0.54**
	Baseline	**86.83**	38.51	0.42	0.43
GermEval18	Without adversary	50.41	63.66	0.53	0.65
	Adversary$_{sep}$	56.11	**65.71**	**0.56**	**0.67**
	Baseline	**63.86**	64.45	0.53	0.65
HasocDe19	Without adversary	75.00	31.11	0.25	0.39
	Adversary$_{sep}$	65.04	**41.02**	**0.47**	**0.54**
	Baseline	**92.68**	38.67	0.42	0.43
HasocHin19	Without adversary	**79.90**	58.67	0.56	0.63
	Adversary$_{sep}$	68.95	61.84	**0.59**	**0.66**
	Baseline	73.08	**62.92**	0.58	0.66
L-HSAB19	Without adversary	79.08	53.84	0.48	0.58
	Adversary$_{sep}$	77.78	54.93	0.49	0.59
	Adversary$_{joint}$	81.04	54.64	0.50	0.58
	Baseline	**81.71**	**59.03**	**0.52**	**0.62**
Multilingual	Without adversary	77.04	**72.37**	0.63	**0.76**
	Adversary$_{sep}$	79.45	70.03	**0.65**	0.74
	Baseline	**81.85**	68.41	0.63	0.73

Table 6: Summary of the results

Language	Text	True Label	Vanilla	Adversary
en	I can't stress how much I hate these liberal Muslims that bend over backwards for these cunts and these Uncle Tom ass middle eastern / south Asian /Asian / african peoples who sell out like this	hate	offensive	hate
en	RT @user: This is one bad ass orangutan @emoji; @url	hate	offensive	hate
de	@user @user Glaubte Du echt, eine Frau mit befriedigendem Sexleben rennt durch die Welt und sieht überall böse Männer und sexuelle Gewalt? *(@user @user Did you really believe that a woman with a satisfying sex life runs through the world and sees evil men and sexual violence everywhere?)*	hate	offensive	hate
de	@user Genau. Die Froschfresser haben nichts gelernt! Ihr Untergang ist selbstverschuldet. *(@user Exactly. The frog eaters haven't learned anything! Your downfall is self-inflicted.)*	hate	offensive	hate
hindi	झोंपड़ी के, कुछ दिन पहले तक तो तू उस आदतन 'बाइक चोर' तबरेज के लिये छाती पीट रहा था। हर चैनल पर रंडी रोना मचा रखा था। अब सेक्युलरिज्म का 'अखरोट' अपने पिछवाड़े से तोड़ने की कोशिश कर रहा है। आतंकी साला *(Of the hut, till a few days ago, you were habitually beating your chest for that 'bike thief' Tabrez. Randi was crying on every channel. Now the 'nut' of secularism is trying to break from its backyard. terrorist brother)*	offensive	hate	offensive
arabic	ma fy ay shk bs aldrwz klab w khwnh[7] *(True, there is no doubt, but the Druze are dogs and traitors)*	hate	offensive	hate

Table 7: Error analysis on `DistilBERT` predictions versus actual labels for the examined languages.

cept *L-HSAB19*, while also outperforming baseline models on multiple occasions. Error analysis reveals that the debiased model moves past keyword-based predictions, taking into account the context as well. Both the true positive rate for hate speech and overall performance are improved, showcasing the stabilizing capabilities of our novel methodology on hate speech detection.

[7]Example transliterated.

9 Ethical Considerations

In our work we deal with hate speech, which could potentially cause harm (directly or indirectly) to vulnerable social groups. We do not support the views expressed in these hateful posts, we merely venture to analyze and provide solutions to mitigate this online phenomenon.

Further, we could only examine a specific problem (neutral vs. offensive vs. hateful language) in specific languages. This is a non-exhaustive list and there is a lot we did not cover. Care should be taken to use these methods only in the examined languages since generalization may not be feasible (in fact, we show there are issues with our method in Arabic).

10 Acknowledgments

This work has been funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The second author was partly supported by the European Research Council (#740516). The authors of this work take full responsibility for its content.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. A dataset of Hindi-English code-mixed social media text for hate speech detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 36–41, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Athena Vakali. 2017. Mean birds: Detecting aggression and bullying on twitter. In *Proceedings of the 2017 ACM on web science conference*, pages 13–22.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Antigoni-Maria 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. In *11th International Conference on Web and Social Media, ICWSM 2018*. AAAI Press.

Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021. Diverse adversaries for mitigating bias in training. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2760–2765, Online. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Muhammad Okky Ibrohim and Indra Budi. 2019. Multi-label hate speech and abusive language detection in Indonesian Twitter. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 46–57, Florence, Italy. Association for Computational Linguistics.

Thomas Mandl, Sandip Modha, Prasenjit Majumder, Daksh Patel, Mohana Dave, Chintak Mandlia, and Aditya Patel. 2019. Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th forum for information retrieval evaluation*, pages 14–17.

Ilia Markov and Walter Daelemans. 2021. Improving cross-domain hate speech detection by reducing the false positive rate. In *Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 17–22, Online. Association for Computational Linguistics.

Puneet Mathur, Ramit Sawhney, Meghna Ayyar, and Rajiv Shah. 2018. Did you offend me? classification of offensive tweets in Hinglish language. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 138–148, Brussels, Belgium. Association for Computational Linguistics.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. 2019. A bert-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer.

Hamdy Mubarak, Kareem Darwish, and Walid Magdy. 2017. Abusive language detection on Arabic social media. In *Proceedings of the First Workshop on*

Abusive Language Online, pages 52–56, Vancouver, BC, Canada. Association for Computational Linguistics.

Hala Mulki, Hatem Haddad, Chedi Bechikh Ali, and Halima Alshabani. 2019. L-HSAB: A Levantine Twitter dataset for hate speech and abusive language. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 111–118, Florence, Italy. Association for Computational Linguistics.

Nedjma Ousidhoum, Zizheng Lin, Hongming Zhang, Yangqiu Song, and Dit-Yan Yeung. 2019. Multilingual and multi-aspect hate speech analysis. 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 4675–4684, Hong Kong, China. Association for Computational Linguistics.

Jing Qian, Hong Wang, Mai ElSherief, and Xifeng Yan. 2021. Lifelong learning of hate speech classification on social media. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2304–2314, Online. Association for Computational Linguistics.

Tharindu Ranasinghe and Marcos Zampieri. 2020. Multilingual offensive language identification with cross-lingual embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5838–5844, Online. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10.

Gudbjartur Ingi Sigurbergsson and Leon Derczynski. 2020. Offensive language and hate speech detection for Danish. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3498–3508, Marseille, France. European Language Resources Association.

Steve Durairaj Swamy, Anupam Jamatia, and Björn Gambäck. 2019. Studying generalisability across abusive language detection datasets. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 940–950, Hong Kong, China. Association for Computational Linguistics.

Christina Wadsworth, Francesca Vera, and Chris Piech. 2018. Achieving fairness through adversarial learning: an application to recidivism prediction. *arXiv preprint arXiv:1807.00199*.

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

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th international conference on world wide web*, pages 1391–1399.

Mengzhou Xia, Anjalie Field, and Yulia Tsvetkov. 2020. Demoting racial bias in hate speech detection. In *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*, pages 7–14, Online. Association for Computational Linguistics.

Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. Detecting hate speech on twitter using a convolution-gru based deep neural network. In *European semantic web conference*, pages 745–760. Springer.