AAA: Fair Evaluation for Abuse Detection Systems Wanted

Agostina Calabrese
a.calabrese@ed.ac.uk
The University of Edinburgh
Edinburgh, United Kingdom

Michele Bevilacqua
bevilacqua@di.uniroma1.it
Sapienza University of Rome
Rome, Italy

Björn Ross
b.ross@ed.ac.uk
The University of Edinburgh
Edinburgh, United Kingdom

Rocco Tripodi
Roberto Navigli
tripodi@di.uniroma1.it
navigli@di.uniroma1.it
Sapienza University of Rome
Rome, Italy

ABSTRACT

User-generated web content is rife with abusive language that can harm others and discourage participation. Thus, a primary research aim is to develop abuse detection systems that can be used to alert and support human moderators of online communities. Such systems are notoriously hard to develop and evaluate. Even when they appear to achieve satisfactory performance on current evaluation metrics, they may fail in practice on new data. This is partly because datasets commonly used in this field suffer from selection bias, and consequently, existing supervised models overrely on cue words such as group identifiers (e.g., *gay* and *black*) which are not inherently abusive. Although there are attempts to mitigate this bias, current evaluation metrics do not adequately quantify their progress. In this work, we introduce Adversarial Attacks against Abuse (AAA), a new evaluation strategy and associated metric that better captures a model's performance on certain classes of hard-to-classify microposts, and for example penalises systems which are biased on low-level lexical features. It does so by adversarially modifying the model developer's training and test data to generate plausible test samples dynamically. We make AAA available as an easy-to-use tool, and show its effectiveness in error analysis by comparing the AAA performance of several state-of-the-art models on multiple datasets. This work will inform the development of detection systems and contribute to the fight against abusive language online.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; • **Social and professional topics** → **Hate speech**; • **General and reference** → **Evaluation**; • **Networks** → *Online social networks*.

KEYWORDS

abuse detection, hate speech, evaluation

ACM Reference Format:
Agostina Calabrese, Michele Bevilacqua, Björn Ross, Rocco Tripodi, and Roberto Navigli. 2021. AAA: Fair Evaluation for Abuse Detection Systems Wanted. In *13th ACM Web Science Conference 2021 (WebSci '21), June 21–25, 2021, Virtual Event, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3447535.3462484

1 INTRODUCTION

The Web is full of abusive language, from hate speech and racist or sexist stereotypes to targeted cyberbullying and profane insults [22]. Since the volume of new posts is unmanageable for human moderators, there is an unquestionable need for reliable automatic techniques that help to keep the online space safe. The ability to detect and quantify abuse is also important in itself for web science, for example, to be able to research questions about what kind of content attracts hateful discourse [26]. This makes abuse detection an important task for Natural Language Processing (NLP).

To address the problem researchers have employed various heuristics to upsample the abusive messages in order to make it possible to train and evaluate supervised systems. Among the problems that make the development of abuse detection systems hard is the unsuitableness of the available automatic evaluation tools. It is standard practice to report F1 scores achieved on held-out data and/or on existing published datasets, but this strategy is likely to give systems unrealistically high scores. It has been shown that datasets that are commonly in use in the field show strong biases towards certain words or topics [25], partly as a result of the sampling strategy used to obtain a sufficient number of abusive posts. As a result of this, classifiers learn to exploit said bias. For example, the well-known dataset of Waseem and Hovy [23] features tweets discussing whether women are suitable as football commentators. This results in words such as *commentator* and *football* being strong hate cues in the dataset, harming the generalisability and increasing the rate of false positives[1]. Wiegand et al. [25] showed that classification scores on popular datasets reported in previous works are much lower under realistic, less biased settings.

WebSci '21, June 21–25, 2021, Virtual Event, United Kingdom

ACM ISBN 978-1-4503-8330-1/21/06…$15.00
https://doi.org/10.1145/3447535.3462484

[1] In the context of binary abuse detection, the abusive class is regarded as the positive class.

Although the evaluation of a system on different datasets might help in assessing its ability to generalise, there is no guarantee that continuing to use *only static datasets* for benchmarking will actually help in detecting data-specific biases or assessing whether the model's capability to detect abuse is robust, leaving researchers with no clues about what the weaknesses of their systems are. Furthermore, it is rarely the case that a suitable resource is available. This is because abuse detection datasets are very context-dependent, due to their focus on different subtypes of harmful content (e.g., sexism/racism, hate speech/offensive language) and platforms (e.g., Twitter, Wikipedia).

Our approach makes progress on these issues, by providing a more comprehensive evaluation of abuse detection systems. Specifically,

(1) we introduce a set of techniques to adversarially modify examples. The generated new examples are harder to classify, especially for systems that rely on dataset bias, but they are still plausible. They retain the lexical material of the original text, and we maintain full control of the output label. Our attacks cover categories that were previously absent in the literature, such as the transformation of abusive texts into non-abusive ones.

(2) We define Adversarial Attacks against Abuse (AAA), an evaluation strategy and associated metric for abuse detection systems. AAA measures the performance of a system on the examples that have been dynamically generated from the model developer's training and test data. It adapts to a wide range of *toxic* microposts (e.g., abusive, racist, hateful) and domains. We make the AAA tool[2] publicly available so that researchers can easily evaluate models on the dataset of interest.

(3) We empirically evaluate state-of-the-art architectures with the AAA score, showing that even systems which are specifically meant to address the bias issue are in fact not very resilient to it.

2 RELATED WORK

We use *abuse* as an umbrella term covering any kind of harmful content on the Web, as this is accepted practice in the field [19, 22]. Abuse detection is plagued by a number of problems, and as a consequence, systems are hard to develop and evaluate. Human annotators are notoriously unreliable at deciding whether or not a given social media post is hateful [14]. There is considerable ambiguity in definitions and a variety of overlapping related concepts [22]. Despite the amount of effort that goes into creating them, existing datasets are often limited to certain subtypes of abusive language (such as racism and sexism), and the sampling strategies used to obtain abusive microposts from social media are far from ideal because they can lead to selection bias [25]. This makes evaluation challenging.

Researchers have started to combine the introduction of new models with a qualitative inspection of the model's behaviour and explicit attempts to mitigate the effects of known biases on it. For example, Mozafari et al. [12] introduced a BERT-based model [5] fine-tuned for the abuse detection task and reported astonishing F1 scores on the Waseem et al. [24] and Davidson et al. [4] datasets, but they at the same time recognised how such performance was mainly due to the system's ability to model data bias. Kennedy et al. [9] and Zhang et al. [27] demonstrated how hate speech classifiers tend to systematically produce false positives when an input sentence contains group identifiers, such as "gay" or "black". This is due to the fact that models are not able to detect hateful statements as humans would (understanding the message as a whole) and focus, instead, on specific shallow features of the training set. As a partial solution, Kennedy et al. [9] proposed the use of a *post-hoc* regularization technique, called Sampling and Occlusion [8, SOC]. This technique allows reducing the importance given by a model to a set of group identifiers, to give more importance to more generalisable features. However, this approach is only able to make models more receptive to general features, but not to understand the context and the use of language. We will demonstrate this in Section 5 with experiments tailored to probe the ability of a model to identify when a message is used in a derogatory way or when it contains just a mention of a hateful message.

One of the first attempts to study the robustness of abuse detection systems through the use of perturbations or modifications of input text was performed by Gröndahl et al. [7]. In this work the authors showed how the injection of adversarial attacks (i.e., perturbations of input posts) can cause huge performance drops in detection systems. In particular, they experimented with the insertion of typos, changes in word boundaries and with the addition of 10 to 50 non-hateful words to posts labelled as abusive. Words to be appended were selected among common English words and words that were "common" among the non-abusive posts contained in the dataset. Although the attacks are very effective, the modifications often resulted in posts that are too artificial (e.g., when appending 10 to 50 words to short posts like tweets), making their usefulness as an evaluation tool questionable. Moreover, none of the described attacks involve changes in the ground truth label of posts, and they therefore fail to detect important weaknesses in current state-of-the-art models. A very popular approach to the creation of adversarial attacks for text classification was presented by Wallace et al. [20], but it is of limited interest for an evaluation tool as a) its "triggers" are often gibberish for humans b) it is concerned with malicious attacks to a model rather than with the robust evaluation of inputs that could have been plausibly produced by a human c) it assumes white-box access to the model weights.

A further step towards a fairer but still automatic evaluation of models is HateCheck [15], a benchmark of 29 functionality tests for the detection of hateful content. The 29 tests were motivated by findings in previous research and interviews with civil society stakeholders, and were designed starting from a fixed set of group identifiers and slurs. Since HateCheck is a static benchmark, researchers who wish to use it to evaluate their models would have to perform non-trivial changes to adapt the tests to their domain of interest (e.g., by skipping tests related to demographic groups that are out of their scope). More importantly, the static nature of HateCheck prevents it from penalising dataset-specific biases, making very promising tests such as counterspeech detection (i.e., identification of references to harmful messages that seek to act against them) less effective. Instead of evaluating a system in a static environment that might not meet the needs of researchers,

[2] https://github.com/Ago3/Adversifier

or measuring its capability to handle very artificial posts, there is a need for an approach that combines the dynamic nature of adversarial attacks [7] with the more "natural" style of handcrafted examples [15].

3 AAA: A NEW EVALUATION METRIC FOR ABUSE DETECTION SYSTEMS

In this work, we introduce AAA, a new evaluation metric for abuse detection systems meant to complement the information provided by the F1-score on standard datasets, by evaluating systems in hard scenarios, built starting from the training and test sets. This approach has two main advantages. First, AAA is not tied to a fixed definition of what "abusive content" is, and is hence suitable for the evaluation of a wide range of models for the detection of toxic (e.g., abusive, racist, hateful) textual microposts. Second, it tailors the definition of "hard examples" to the dataset in use and adapts itself, for instance, to avoid rewarding a system for modelling selection biases in the datasets, and for exhibiting overamplification phenomena [17, selection bias, overamplification]. We achieve this by defining grey-box adversarial attacks (see Section 3.1), and compute the AAA score as a system's performances in such scenarios (see Section 3.2).

3.1 Attacks

We define an attack as a tuple (f, c, c') where f is a function that takes as input a text t belonging to class c and returns the perturbed text $f(t)$, and c' is the ground truth class of $f(t)$. All our attacks are grey-box, meaning that while we do not assume access to the model's gradient, we still require access to the training set. This enables us to adapt the definition of hard-to-classify posts to the dataset in use. In particular, the attacks are designed to fulfil the following criteria:

C.1 $f(t)$ is harder to classify correctly than the original post, especially for systems that overrely on the biases found within the training set.

C.2 $f(t)$ is still a plausible post (i.e., not too artificial).

The resulting attacks are based on three characteristics that make posts hard to classify: when the post merely mentions an abusive message, for example to express disagreement (as part of counter-speech) (Section 3.1.1), when the abuse is hidden in a post that is known to be non-abusive (Section 3.1.2), and finally, when it contains words that the model knows to be strongly associated with the opposite class (Section 3.1.3).

3.1.1 Flipping the Label: Abusive to Non-Abusive. To turn abusive into non-abusive posts, we embed them, in quotation marks, into a non-abusive template that merely mentions the original post, but does not express agreement with it (the $\text{Quo}_{A\to N}$ attack). For each abusive post, we draw a template uniformly at random. For instance, by embedding `'Men ''have a discussion''. Women ''argue'''`[3] into a template, we produce the non-abusive `'Here is what she said: ''Men 'have a discussion.' Women 'argue.''' Bigotry in action.'`. Our approach has several advantages: (1) it leaves the lexical material in the original post unchanged, exploiting the vulnerability of systems that overrely on shallow features such as cue words; (2) the resulting text is plausible and (3) it adapts to any dataset-specific definition of abusiveness.

We mined 317 templates from real tweets semi-automatically as follows. To identify tweets that mention potentially abusive text without expressing agreement, we retrieved a public archive of tweets[4] and filtered out all the unique English tweets that did not contain a pair of quotation marks, obtaining around 925,000 instances; then, we processed tweets by removing all quoted material and substituting it with a special `<mask>` token. We manually selected 5 viable templates from a sample of 5,000 tweets. The rest of the dump was used to automatically search for candidate templates with the help of BERTweet [13], a powerful NLP model that embeds tweets as high-dimensional latent vectors. We use BERTweet to encode both the seeds and the dump: each micropost is encoded by mean pooling over all the hidden states produced by the fourth, third and second to last layers. Then, for each tweet in the dump we computed a relevance score as the maximum cosine similarity between the vector of the tweet and that of each of the seeds. Three annotators each processed 3,000 of the top 9,000 most relevant tweets. Annotators were asked to decide whether it was clear that the candidate template did not express agreement with the quoted text, for *all possible messages* that could be embedded inside of it – if there was a reasonable counterexample the tweet had to be discarded. Approved tweets were lightly edited to remove sensitive or too specific information, e.g. URLs and user handles, together with vulgar or derogatory terms[5]. In a second stage each of the annotators validated the accepted candidate templates produced by the other two annotators. They were able to either discard or slightly revise them. In the third and final stage a fourth annotator validated the templates, marking each item as either valid or invalid. Templates were found to be of high quality, with 99.4% approved. The resulting templates feature many different varieties of English and degrees of formality. Most express disagreement, while some report on what was said without taking a stance.

3.1.2 Flipping the Label: Non-Abusive to Abusive. We turn non-abusive microposts into abusive ones by combining them with ones that are known to be abusive (the $\text{Flip}_{N\to A}$ attack). What exactly is considered abusive depends on the definition adopted. For this reason, we pick the abusive messages (p_a) from the test set and prepend it to the original post: $f(t)$ = "p_a t". For instance, we turn the non-abusive tweet `'I have a few favourites #mkr'` into abusive by prepending the abusive tweet `'Men ''have a discussion''. Women ''argue'''` to it. Since $f(t)$ is the result of the concatenation of two existing tweets, it is likely to also be a grammatical utterance, and not too artificial (although the topics discussed in the two components might be different). Furthermore, picking the abusive posts from the test set makes the resulting post $f(t)$ more challenging for a biased system, hence fulfilling our criteria.

[3] All the examples are taken from the Waseem et al. [24] dataset. Therefore, assessing the actual abusiveness level of the mentioned tweets is out of the scope of this work.

[4] Downloaded from https://archive.org/details/twitterstream. Period: November 2018 - February 2019.

[5] According to the Google Dictionary taxonomy (https://languages.oup.com/google-dictionary-en/).

To assess the quality of the generated posts, we apply the $\text{Flip}_{N\rightarrow A}$ attack to the Waseem et al. [24] dataset, and ask 3 annotators to review a sample of 100 instances. In particular, annotators are shown the original posts (p_a and t) with their original labels (abusive and non-abusive, respectively) and asked to verify that "p_a t" is abusive (e.g., t does not express disagreement with p_a). As a result, 94% of the generated posts were approved by all the annotators, and 100% were approved by at least 2 annotators.

3.1.3 Keeping the Label Constant. To perturb posts without changing the label, we build on the word appending attack of Gröndahl et al. [7], which involves the concatenation of 10 to 50 non-hateful words to hateful speech, and define two new, less artificial variants. In the first, we aim to make the classification of abusive posts harder, without changing the label of the original post ($\text{Corr}_{A\rightarrow A}$). To this end, we create a dictionary of non-abusive expressions by looking for common words that have a high correlation with the non-abusive class in the training set. Gröndahl et al. [7] used correlation as the only criterion for word selection, but this approach is prone to selecting rare words. To identify words that a naive supervised model is likely to rely on, we train a binary logistic regression model with word-count features on the training set and select the 100 words with the highest regression coefficients. Then, for each test instance, we sample k words from this dictionary (k is drawn uniformly at random from $[1, 5]$), and append them to the original abusive post in the form of hashtags. Hence, a sexist tweet like `'Men ''have a discussion''. Women ''argue'''` is changed to the still sexist `'Men ''have a discussion''. Women ''argue'' #calls'`.

The second variant uses a similar technique to modify non-abusive posts without changing their label ($\text{Corr}_{N\rightarrow N}$). In particular, we exploit the binary logistic regression model as described above to create a dictionary of words that are predictive of the abusive class. However, not every word that has a high correlation with the abusive class can be appended to a non-abusive message without flipping its label. Therefore, we filter out from the set of such words any token that has an entry (after lemmatisation) in HurtLex, a multilingual lexicon for hate speech [1], and append k of the remaining words to the original post in the form of hashtags. Since current technologies are not able to perfectly tokenise hashtag content, making it harder to detect the presence of hateful words within hashtags, we choose to discard hashtags while building the dictionary. As an example, consider the tweet `'I have a few favourites #mkr #black #transgender'` obtained starting from the non-abusive post `'I have a few favourites #mkr'`.

We assess the quality of the generated posts by applying the $\text{Corr}_{A\rightarrow A}$ and $\text{Corr}_{N\rightarrow N}$ attacks to the Waseem et al. [24] dataset, and asking 3 annotators to review a sample of 100 instances for each attack. In particular, annotators are shown the original posts with their original labels, and asked to verify that the appended hashtags do not affect the label. As a result, 100% of the posts generated with the $\text{Corr}_{A\rightarrow A}$ attack were approved by all the annotators, while 87% of the instances generated with the $\text{Corr}_{N\rightarrow N}$ attack were approved (with 92% approved by at least 2 annotators). Importantly, the higher error rate observed for the $\text{Corr}_{N\rightarrow N}$ attack is due to the absence of the word "feminazi" in HurtLex, which in turn did not prevent the tool from selecting it as a non-abusive hashtag.

What If Hashtags are Ignored? Appending words in the form of hashtags allows us to freely concatenate words to the original post t, without making the outcome $f(t)$ nonsensical. However, we are aware that some models ignore hashtags, and for example replace each occurrence with a generic token. This pre-processing step would neutralise the effects of the $\text{Corr}_{A\rightarrow A}$ and $\text{Corr}_{N\rightarrow N}$ attacks, hence resulting in falsely high scores. To avoid rewarding a system for the (questionable) choice of discarding hashtags, we design a test meant solely to determine whether hashtags are being ignored or not. More specifically, we create a copy of the test set where each word in each post is a hashtag, by simply prepending the symbol "#" to any non-hashtag word. For instance, the tweet `'I have a few favourites #mkr'` becomes `'#I #have #a #few #favourites #mkr'`. We then compare the True Positive and Negative rates obtained by the target system on both the original test set and the new version. If a significant (χ^2 test, $p = 0.05$) drop in performances is found, then we assume that hashtags are being wrongly pre-processed, and assign a score of 0 to both the $\text{Corr}_{A\rightarrow A}$ and $\text{Corr}_{N\rightarrow N}$ attacks. Guidelines about pre-processing choices are given in Section 6.1.

3.2 Formula

Once a system has been evaluated on the dataset of interest perturbed by all our adversarial attacks, one at a time, we compute the AAA score as the geometric mean of the Correct Prediction Rates (CPRs) reported in each setting. We choose to use CPR as the measure of performance since, in each setting, all the instances share the same ground truth label. More specifically, the AAA score is obtained as:

$$\text{AAA} = \mu(\text{Quo}_{A\rightarrow N}, \text{Corr}_{N\rightarrow N}, \text{Flip}_{N\rightarrow A}, \text{Corr}_{A\rightarrow A}) \qquad (1)$$

where μ is the geometric mean function, and any other term a refers to the CPR obtained when attack a is applied.

Since each of the attacks exposes a different weakness in the model under evaluation, AAA can penalise systems overrelying on *some* data bias. But does AAA capture all the known biases in the context of abuse detection? We used two approaches to address this question, a literature review and a computational one.

Neural abuse detection systems are known to be over-sensitive to group identifiers like "woman" or "black". Dixon et al. [6] identified the source of this bias in the disproportionate representation of identity terms in datasets, where terms like "gay" happen to occur more frequently in abusive posts than non-abusive ones. Although none of our attacks directly addresses this issue, models that overrely on group identifiers would be penalised by the $\text{Corr}_{N\rightarrow N}$ attack. In fact, dynamically building a dictionary of common words that have a high correlation with the abusive class allows us to capture these group identifiers (in datasets where these actually represent an issue). When applied to the Waseem et al. [24] dataset, hashtags like `#girls`, `#blonde`, `#female` and `#muslim` are appended to non-abusive tweets, testing a model's capability to correctly identify the resulting sentence as non-abusive despite the presence of the group identifier.

Abuse detection systems are also known to regard posts containing features associated with some dialects (e.g., African American English) as more likely to be abusive than posts that do not contain these features [2, 3, 16]. We acknowledge that the current version of the AAA metric is not fully able to penalise models presenting such bias.

In addition, we exploited a recently introduced computational technique for dealing with unknown biases in NLP models [18] to check for further easy-to-spot biases that were not yet captured by AAA, and potentially design new attacks to fill the gap. In particular, Utama et al. [18] proposes to learn a biased version of a model M by training M on a sample of the training set containing N instances for e epochs. The parameters N and e are tuned so that M achieves an accuracy score of 60-70% on the unseen training examples, and more than 90% of M's predictions are in the high confidence bin. We follow this procedure using the BERT-based system introduced by Mozafari et al. [12] (namely, BERT_{MOZ}), since the authors themselves suggested using their model as a tool to spot the presence of biases in a dataset. As for the data, we choose the widely-used Waseem et al. [24] dataset and split it into training, validation and test sets according to the 80%/10%/10% rule, while keeping the distribution of the labels equal in each split. We find the two conditions above to be satisfied when BERT_{MOZ} is trained for 5 epochs on 1,500 samples. Then we manually checked the predictions of the model on the unseen training examples, but we were unable to find evidence of biases besides the ones already discussed in this section.

4 EVALUATION SETUP

4.1 Datasets

We experiment with the widely-used Waseem and Hovy [23] (*Waseem*) and Davidson et al. [4] (*Davidson*) datasets. For *Waseem*, we follow Waseem et al. [24] and merge the examples from [23] and [21], obtaining 19,639 unique tweets annotated with the classes "sexism" (4,115), "racism" (2,061), "neither" (13,417) and "both" (46). When evaluating with AAA, we map the "sexism", "racism" and "both" classes to the abusive one, and regard "neither" as non-abusive. The *Davidson* dataset focuses on a different concept of abuse, and contains 24,783 tweets labelled as "hate speech" (1,430), "offensive" (19,190) and "neither" (4,163). At testing time, we map all instances labelled as "hate speech" or "offensive" to the abusive class and consider the remaining instances as non-abusive. For both *Waseem* and *Davidson*, we use stratified sampling to split 0.8, 0.1, and 0.1 portions of tweets from each class into training, validation and test sets. For the sake of replicability, we release the ids of the examples in each split for both datasets.

4.2 Models

The first baseline that we use is the BERT-based model presented in [9] (BERT_{KEN}). This architecture is composed of the fine-tuned $\text{BERT}_{\text{BASE}}$ followed by a fully connected neural network without a hidden layer, and is trained using a regularization technique meant to reduce the bias on group identifiers. Since an analogous architecture had already been proposed in [12], but with no attempts to mitigate modelling the bias, we decided to add a re-implementation of this model as a comparison system (BERT_{MOZ}). Finally, we include the results for an SVM model with bag-of-words features, since such simple architecture has been shown to obtain promising results.

We did not re-implement the BERT_{KEN} model, but used the code (and parameters) made available by the authors[6].

We train our implementation of BERT_{MOZ} using the same parameter values as in [12]. More specifically, we train the classifier with a dropout probability of 0.1 and use the Adam optimiser [10] with a learning rate of 2e-5. We increase the number of training epochs to 20, and register the best performance on the validation set (w.r.t the F1 score) after 9 epochs for *Waseem* (batch size 32), and 18 for *Davidson* (batch size 16).

As for the SVM model, we exploit word count features, use a linear kernel and set the regularization parameter to 1.

We adopt the same pre-processing procedure for both the *Waseem* and *Davidson* datasets, and replace usernames, URLs and numbers with special tokens. We mark hashtags using the special tokens ⟨hashtag⟩ and ⟨/hashtag⟩, and tokenise their content using the `TweetTokenizer` provided within the NLTK library. All punctuation marks are replaced with spaces, and posts are converted to lower case.

5 EVALUATION RESULTS

Our experiments highlight critical weaknesses in current state-of-the-art classifiers (Section 5.1) and demonstrate AAA's capability to penalise models that are biased on low-level lexical features (Section 5.2). We show that existing models exhibit alarmingly low performance in recognising speech that only quotes abusive language, and highlight the importance of testing models dynamically (Section 5.3).

5.1 How Do State-of-the-Art Models Perform on AAA?

The results achieved by the SVM, BERT_{MOZ} and BERT_{KEN} models on the *Waseem* and *Davidson* datasets are shown in Table 1. The first column group (delimited by ||) contains the micro-F1 scores achieved by the models on the **original dataset**. Although this is a binary classification setting, we show micro-averaged scores because this reflects standard practice in the field, where, as a consequence of averaging over multiple classes, performance on the non-abusive class is regarded as equally important as performance on any other class. On *Waseem* all the models under study achieve high (> 82%) F1-scores. For the SVM and BERT_{MOZ} models such scores are mainly the result of the systems' ability to detect non-abusive posts, with True Negative Rates (TNRs) above 91% and True Positive Rates (TPRs) below 70%. BERT_{KEN} exhibits a different behaviour, achieving comparable performances on the two classes (TPR ≈ TNR ≈ 82). Since BERT_{KEN} is optimised to reduce the number of false positives by reducing the importance of some group identifiers such as hate cues, one might expect it to achieve a higher TNR than the non-debiased equivalent (BERT_{MOZ}). This is not what we observe in our experiments: on the contrary, BERT_{KEN} has a ∼ 9 percentage points lower TNR, and a ∼ 15% points higher TPR than BERT_{MOZ}. This suggests that the optimisation technique employed to reduce bias towards group identifiers might have the collateral effect of increasing the importance of other low-level

[6] https://github.com/BrendanKennedy/contextualizing-hate-speech-models-with-explanations

Dataset	Model	F1	Non-Abusive (TNR)			Abusive (TPR)			AAA
		Original	Original	Quo$_{A\rightarrow N}$	Corr$_{N\rightarrow N}$	Original	Flip$_{N\rightarrow A}$	Corr$_{A\rightarrow A}$	
Waseem	SVM	82.18	95.01	51.93	59.46	54.50	45.53	33.28	46.51
	BERT$_{\text{MOZ}}$	84.42	91.58	35.21	60.28	68.97	61.85	51.29	50.94
	BERT$_{\text{KEN}}$	82.18	82.19	19.61	40.01	82.15	74.14	73.63	45.50
Davidson	SVM	91.81	88.73	9.65	51.80	92.43	79.14	54.03	38.24
	BERT$_{\text{MOZ}}$	95.76	89.45	4.12	41.49	97.04	96.88	95.59	35.47
	BERT$_{\text{KEN}}$	94.39	73.86	1.21	26.86	98.55	99.28	97.87	23.72

Table 1: Results achieved on the AAA attacks by the SVM, the finetuned BERT$_{\text{BASE}}$ model (BERT$_{\text{MOZ}}$), and the finetuned BERT$_{\text{BASE}}$ model optimised to reduce bias towards group identifiers (BERT$_{\text{KEN}}$) on the respective dataset. The four column groups (delimited by ||) contain, from left to right, the micro F1-scores on the original dataset, the True Negative Rate (TNR) on the original dataset and on the non-abusive examples generated by our attacks, the True Positive Rate (TPR) on the original dataset and on the abusive examples generated by our attacks, and the AAA scores. All scores are reported as percentages (higher is better).

	Original		Hashtag		
Model	**TNR**	**TPR**	**TNR**	**TPR**	**AAA**
SVM	95.01	54.50	99.25	12.86	46.44
BERT$_{\text{MOZ}}$	91.58	68.97	93.14	45.50	51.00
BERT$_{\text{KEN}}$	82.19	82.15	71.01	78.46	45.50
BERT$_{\text{MOZ-NH}}$	87.18	74.92	100.00	00.00†	0.00

Table 2: The True Negative and True Positive rates for the hashtag check, and the AAA scores, achieved by the SVM, the finetuned BERT$_{\text{BASE}}$ model (BERT$_{\text{MOZ}}$), the finetuned BERT$_{\text{BASE}}$ model optimised to reduce bias towards group identifiers (BERT$_{\text{KEN}}$), and the variant of BERT$_{\text{MOZ}}$ that fully discards hashtags (BERT$_{\text{MOZ-NH}}$). Results are shown for the Waseem dataset (*Original*), and a variant of the dataset where each word is turned into an hashtag (*Hashtag*). All scores are reported as percentages. †: statistically different from the corresponding score on the *Original* setting (χ^2 test, $p = 0.05$).

lexical features that characterise the abusive class within the dataset, but less precisely (i.e., causing more false positives).

The second column group in Table 1 shows the models' TNRs on the hard non-abusive examples generated by the **Quo**$_{A\rightarrow N}$ and **Corr**$_{N\rightarrow N}$ attacks. Discrimination of harmful content from counterspeech is shown to be remarkably hard for state-of-the-art models, with the best model (SVM) performing just above random. Importantly, the SVM achieves a TPR of 54.50% on the original dataset, showing that the "high" performance on the Quo$_{A\rightarrow N}$ setting is mainly due to its inability to recognise abusive posts in the first place. Overall, our experiments demonstrate that state-of-the-art models are not able to judge the abusiveness of a post within the context it appears in. Even a simple perturbation such as the addition of a few hashtags to the original post is very challenging for the studied models, with SVM and BERT$_{\text{MOZ}}$ achieving a TNR of $\sim$ 60%. BERT$_{\text{KEN}}$ is systematically at least $\sim$ 15 points below the other models in both the scenarios, supporting our conjecture around the collateral effects of the employed optimisation technique.

The third column group in Table 1 reports the TPRs achieved on the hard abusive examples generated by the **Flip**$_{N\rightarrow A}$ and **Corr**$_{A\rightarrow A}$ attacks. Our experiments show the SVM model to be the most sensitive to these attacks, which is consistent with its higher TNR on the original dataset and its tendency to assigning higher importance to features predictive of the non-abusive class (i.e., the features exploited by the Flip$_{N\rightarrow A}$ and Corr$_{A\rightarrow A}$ attacks). BERT$_{\text{MOZ}}$ achieves a TPR of $\sim$ 51% on the Corr$_{A\rightarrow A}$ setting, while it is more robust to the Flip$_{N\rightarrow A}$ attack, on which it registers a decrease of only $\sim$ 7 points compared with its TPR on the original *Waseem* dataset. Contrary to what was observed for the non-abusive class, BERT$_{\text{KEN}}$ scores at least 15 points above the other models on both scenarios. However, the decrease observed on the Flip$_{N\rightarrow A}$ setting (compared with the original dataset) is comparable to the one registered by the SVM and BERT$_{\text{MOZ}}$ models. The robustness of BERT$_{\text{KEN}}$ to the Corr$_{A\rightarrow A}$ attack is less surprising when considering that it achieves the lowest TNR on the original dataset (see also the results of the SVM model).

Overall, **AAA** scores (shown in the fourth column group in Table 1) are very low, with BERT$_{\text{MOZ}}$ being the top-ranked model with a score of 51.00% (a random classifier would achieve a AAA score of 50.00%). The SVM and BERT$_{\text{KEN}}$ models achieve similar scores, obtaining AAA scores of 46.44% and 45.50%, respectively.

The results discussed thus far equally apply to our experiments on the *Davidson* dataset (see second row group in Table 1). Scores in the Quo$_{A\rightarrow N}$ scenario are strikingly low, with TNRs ranging from $\sim$ 9% for the SVM model to $\sim$ 1% for BERT$_{\text{KEN}}$. This is not surprising if considering that TPRs on the original *Davidson* dataset are extremely high (> 92%) (and remarkably higher than the corresponding scores on *Waseem*). In fact, these systems are so reliant on low-level lexical features in the abusive posts in the dataset that they fail to recognise when such posts are inserted in a non-abusive context. In contrast, adding non-abusive cues to abusive posts has limited effects on the decisions made by the models, thus resulting in high TPRs in the $\text{Flip}_{N\rightarrow A}$ and $\text{Corr}_{A\rightarrow A}$ scenarios.

Finally, we test whether the hashtag check (Section 3.1.3) works as intended and penalises models that ignore hashtags. All the experiments discussed above were run using a pre-processing procedure that does not discard hashtags: this is correctly recognised by the hashtag check (see Table 2), which therefore does not apply a penalty. When testing a system that fully discards hashtags ($\text{BERT}_{\text{MOZ-NH}}$), the drop in performance becomes significant, and the model is assigned a AAA score of 0%.

5.2 Does AAA Penalise Biased Models?

AAA is designed to penalise models that overrely on group identifiers when making predictions, within the $\text{Corr}_{N\rightarrow N}$ setting. At a first glance, this does not seem to be confirmed by our experiments: BERT_{KEN}, a model explicitly optimised not to overrely on some group identifiers, is the most sensitive to the attack. Assuming a pass/fail threshold of 60%, BERT_{KEN} does not pass the test, while the non-debiased variant of the same model (BERT_{MOZ}) does. However, we recall that the $\text{Corr}_{N\rightarrow N}$ attack exploits non-rare words that are relevant for the abusive class, hence including, but not limited to, group identifiers. The low TNR achieved by BERT_{KEN} in this setting suggests that the technique used to reduce bias towards group identifiers might have the collateral effect of increasing the importance of other low-level lexical features that characterise the abusive class (see also Section 5.1).

To test this, we evaluate the three models with a variant of $\text{Corr}_{N\rightarrow N}$ that only exploits the 25 group identifiers used for the debiasing of BERT_{KEN} ($\text{Ident}_{N\rightarrow N}$). In such a setting, the improvement of BERT_{KEN} over BERT_{MOZ} becomes evident, confirming that the low TNR in the $\text{Corr}_{N\rightarrow N}$ scenario must be due to other unknown biases tackled by this attack. Interestingly, neither BERT_{KEN} nor BERT_{MOZ} passes the test on *Waseem*. When evaluated on the HateCheck functional tests for group identifiers [15, F18,F19], two static sets of examples containing neutral (*Ident-Neutral*) and positive (*Ident-Neutral*) statements about some demographics, BERT_{KEN} is the top-ranked model, but BERT_{MOZ}, whose bias on group identifiers is known, does not fail the test either. This highlights the importance of dynamic evaluation tools such as AAA.

Among the models, the overall AAA score for BERT_{KEN} is the lowest on both the *Waseem* and *Davidson* datasets. This is because, by testing on hard examples for both classes, and by dynamically tailoring the definition of *hard* to the dataset in use, AAA is actually able to measure the *overall* improvement of a model and BERT_{KEN} is not awarded a better score for moving the bias problem from some group identifiers to other low-level features.

5.3 Detection of Counterspeech

Counterspeech is speech that directly counters hate, for example by presenting facts, pointing out hypocrisy or reacting with humour or, sometimes, open hostility [11]. Despite its educational importance, counterspeech is often misclassified by existing abuse detection models, in particular when it contains quotes from the addressed harmful message. Misclassification of counterspeech was the most mentioned issue in the interviews with civil society stakeholders conducted in Röttger et al. [15], and the resulting increase in false positive rates represents a cause of concern for the workload of human content moderators at social media platforms. Since most of the posts in the $\text{Quo}_{A\rightarrow N}$ attack express their disagreement with the original post, the attack effectively generates examples of counterspeech from the input data and thereby tests a model's capability to discriminate this type of counterspeech from harmful messages. Assuming a pass/fail threshold of 60%, none of the models passes this test on either of the datasets. Table 3b summarises the results for the models trained on the *Waseem* dataset. In contrast, all the models perform well on the existing tests for counterspeech detection accuracy in the HateCheck test suite [15, F20,F21] (*Counter-Quote*) and (*Counter-Ref*). This indicates that the score on the $\text{Quo}_{A\rightarrow N}$ component can serve as a useful measure to evaluate a model's ability to distinguish counterspeech from abusive language.

6 DISCUSSION

The results show that existing state-of-the-art models still perform poorly at distinguishing abusive from non-abusive language in various challenging scenarios. While we aimed for the generated posts to be hard to classify, they are not artificial examples that would only appear if an attacker were to try intentionally to fool the model; instead, the microposts were carefully designed to be plausible.

The strategies that are traditionally used to evaluate the models do not show this. This is particularly noticeable in the $\text{Quo}_{A\rightarrow N}$ attack. Despite the excitement around the ability of BERT and related attention-based models that dominate the field, such state-of-the-art models still fail when the speaker repeats a hateful message merely to state their disagreement with it. The models that apparently perform better here are those that were more likely to misclassify the original sentence in the first place, which suggests that their supposedly correct classification of the hard-to-classify post is really the combination of two mistakes. Clearly, there is much work left to do.

6.1 Guidelines for Future Research

In AAA, we introduce an easy-to-use tool that allows the developers of systems for the detection of abusive microposts to evaluate their systems against certain classes of hard-to-classify microposts. Users are simply required to merge all the abusive (non-abusive) classes into a single positive (negative) class. The tool will then build hard examples starting from the training and test sets, and query the model under evaluation in a black-box manner (see Figure 1).

AAA is designed as a diagnostic tool that complements the information provided by the F1-score on standard datasets. We intend for it to be used in error analysis. A good practice for future research in this area is to report both the F1 and AAA scores. Since each AAA component tackles a different weakness in a model's behaviour, reporting all 4 subscores would give a much clearer picture of a classifier's flaws. When summarising a model's performance in a single score is necessary, for example in competitions with leaderboards, we suggest to report the geometric mean of the F1 and the 4 AAA

(a) Group Identifiers

Model	$\text{Corr}_{N\to N}$		$\text{Ident}_{N\to N}$		Ident-Neutral		Ident-Pos	
	TNR	O	TNR	O	TNR	O	TNR	O
SVM	59.46	✗	79.73	✓	85.71	✓	86.24	✓
BERT_{MOZ}	60.28	✓	50.00	✗	96.83	✓	89.95	✓
BERT_{KEN}	40.01	✗	56.86	✗	100.00	✓	99.47	✓

(b) Counterspeech

Model	$\text{Quo}_{A\to N}$		Counter-Quote		Counter-Ref	
	TNR	O	TNR	O	TNR	O
SVM	51.61	✗	91.91	✓	87.23	✓
BERT_{MOZ}	35.37	✗	92.49	✓	95.74	✓
BERT_{KEN}	19.61	✗	90.75	✓	85.82	✓

Table 3: The True Negative Rate achieved by the SVM, the finetuned $\text{BERT}_{\text{BASE}}$ model (BERT_{MOZ}), and the finetuned $\text{BERT}_{\text{BASE}}$ model optimised to reduce bias towards group identifiers (BERT_{KEN}). Results are reported for: (a) our $\text{Corr}_{N\to N}$ attack applied to the Waseem dataset, a variant of this attack restricted to group identifiers ($\text{Ident}_{N\to N}$), and the corresponding functional tests in HateCheck [15, F18,F19] (*Ident-Neutral* and *Ident-Pos*); (b) our $\text{Quo}_{A\to N}$ attack applied to the Waseem dataset, and the corresponding functional tests in HateCheck [15, F20,F21] (*Counter-Quote* and *Counter-Ref*). All scores are reported as percentages, but only outcomes (O columns) are directly comparable among the scenarios.

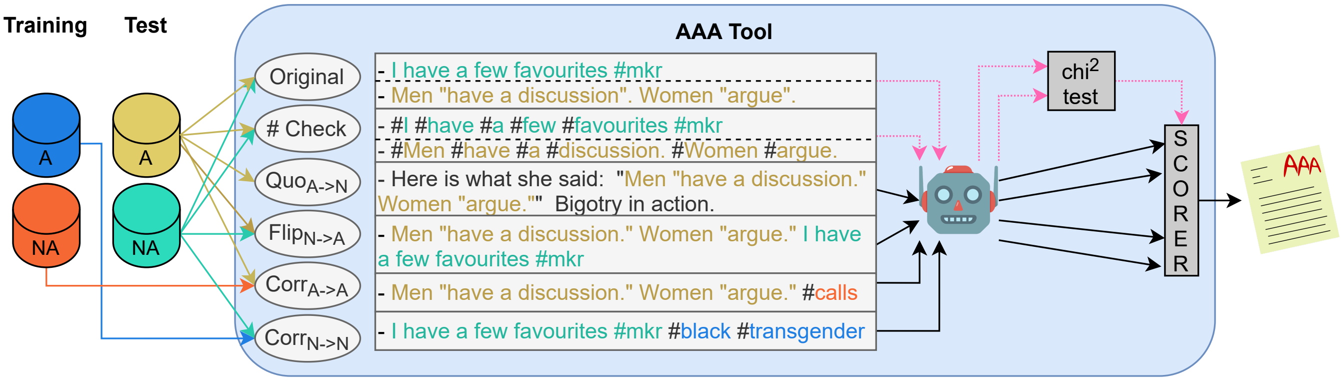
Training
Test
A
NA
A
NA
AAA Tool
Original
# Check
Quo_{A->N}
Flip_{N->A}
Corr_{A->A}
Corr_{N->N}
- I have a few favourites #mkr
- Men "have a discussion". Women "argue".
- #I #have #a #few #favourites #mkr
- #Men #have #a #discussion. #Women #argue.
- Here is what she said: "Men "have a discussion." Women "argue."" Bigotry in action.
- Men "have a discussion." Women "argue." I have a few favourites #mkr
- Men "have a discussion." Women "argue." #calls
- I have a few favourites #mkr #black #transgender
chi² test
SCORER
AAA

Figure 1: The AAA tool builds hard examples starting from the input training and test sets, and after checking whether the model under evaluation discards hashtag content, aggregates the scores on the $\text{Quo}_{A\to N}$, $\text{Flip}_{N\to A}$, $\text{Corr}_{A\to A}$ and $\text{Corr}_{N\to N}$ scenarios.

sub-scores. When instead this necessity arises from a researcher's need to decide for a parameter setting over another, or for a design choice over another, we would advise choosing between the configurations on the Pareto frontier over the five scores.

Although AAA does not come with a pre-defined pre-processing procedure, leaving researchers with full freedom to design their own, researchers should be aware that some choices might be less suitable than others. In particular, we invite researchers to not fully discard the content of hashtags, since this would affect the efficacy of the $\text{Corr}_{A\to A}$ and $\text{Corr}_{N\to N}$ attacks. Models that discard hashtag content are identified by our framework and assigned a score of 0 for the settings corresponding to the mentioned attacks (instead of falsely high scores) (see Section 3.1.3).

Given the importance of distinguishing hate speech from counterspeech [15], which is the focus of our $\text{Quo}_{A\to N}$ attack, researchers may wish to not remove quotation marks. If punctuation is removed, we suggest replacing it with spaces instead of removing it entirely. The main reason for this is that often social media posts are not well-typed, and tweets like `'THIS is why I find the ''antifeminist women just want male attention'' argument moronic...as if there aren't better ways to get that.'` would otherwise give rise to words like *moronicas*. Although we expect such artefacts to have low frequency within the dataset, it might be the case that some of these are selected for the $\text{Corr}_{N\to N}$ attack, and not properly filtered due to the lack of a corresponding entry in HurtLex.

6.2 Limitations

AAA is designed to work for many related NLP tasks such as the detection of abusive language, hate speech, sexism, and racism, but there are a few exceptions in which the $\text{Quo}_{A\to N}$ will not achieve its intended effect of flipping the label. First, since the original quoted material may contain offensive language, AAA is unsuitable for use in situations where offensive words should never be used, not even when reproaching someone else for using them. Second, many of the templates express disagreement with the quoted material, sometimes in the form of a personal attack directed at the abuser (such as calling them ignorant, bigoted, or stupid). Therefore, AAA may not be suitable for use in situations when any kind of insult is considered abusive, although care was taken that the templates never use vulgar or obscene language.

AAA tests models for many of the biases that abusive language detection models are known for. One exception is that it does not fully test a model's ability to predict the abusiveness of a post regardless of the dialect it is written in (Section 3.2). This is because, due to the level of reliability of current technologies for style transfer, it would be impossible to automatically change the style of a post and still be able to construct clear gold labels.

The AAA metric has been designed as an evaluation metric for English datasets. The Flip$_{N\to A}$, Corr$_{A\to A}$ and Corr$_{N\to N}$ attacks can be used on datasets written in any language, although a suitable lexicon for Corr$_{N\to N}$ is needed. To fully enable the use of AAA on non-English datasets, researchers would need to create new templates for the Quo$_{A\to N}$ attack in the target language, following the methodology described in Section 3.1.1.

The AAA metric could be less reliable on datasets where user-related information is provided. For instance, a post containing reclaimed slurs might see its label change depending on the demographic information concerning the author and the target of the post. The Corr$_{A\to A}$ and Corr$_{N\to N}$ attacks could fail at selecting, respectively, non-abusive and abusive words from the training set. To the best of our knowledge, this issue does not concern existing datasets, but it might become relevant in the future.

Finally, datasets containing non-textual information (e.g., images, videos, audio recordings), or long textual posts, are out of the scope of this work.

6.3 Ethical Statement

There are ethical implications to consider in connection to this research. The Quo$_{A\to N}$ attack relies on a collection of templates based on public tweets (Section 3.1.1) that we release together with the AAA tool. In order to comply with Twitter's policies, we discard any information about the authors, anonymise any user mention with `@user`, replace the quoted text, (usually most of the tweet content) with `tweet`, and remove any URL, full name, topic-specific term or strong offence from the post. It might be the case that some templates can still be mapped to the original tweets (and hence users) on the Twitter platform, but the only information deducible from the templates is that the corresponding authors have once quoted and potentially expressed their disagreement about *something*.

7 CONCLUSIONS

In this paper, we have shown various strategies to adversarially modify existing abuse detection evaluation examples, in order to challenge existing automatic classification approaches. We have applied our method to popular abuse detection datasets, and demonstrated that none of the tested classifiers manages to achieve good performances on it. Thus, we introduced AAA, an evaluation metric that complements the information contained within the F1-score obtained on held-out datasets by aggregating performance on the adversarial scenarios. We encourage the community to evaluate future abuse detection approaches with our challenging metric, which requires on the part of the automatic system a more thorough understanding of the text than what is currently the case. It remains to be seen if optimising for AAA (i.e., applying the AAA tool on the validation set at the model selection and hyperparameters tuning phase) might have unintended consequences on the effectiveness of AAA at testing time. We leave this open question for future work.

ACKNOWLEDGMENTS

This work was supported in part by the UKRI Centre for Doctoral Training in Natural Language Processing, funded by the UKRI (grant EP/S022481/1) and the University of Edinburgh, School of Informatics; and the ERC Consolidator Grant MOUSSE No. 726487 under the European Union's Horizon 2020 research and innovation programme.

THE UNIVERSITY of EDINBURGH
UKRI Centre for Doctoral Training
in Natural Language Processing

UK Research
and Innovation

THE UNIVERSITY of EDINBURGH
informatics

erc

REFERENCES

[1] Elisa Bassignana, Valerio Basile, and Viviana Patti. 2018. Hurtlex: A Multilingual Lexicon of Words to Hurt. In *Proc. of CLiC-it*.
[2] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna M. Wallach. 2020. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. In *Proc. of ACL*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (Eds.). 5454–5476.
[3] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial Bias in Hate Speech and Abusive Language Detection Datasets. In *Proc. of ALW3*. 25–35.
[4] Thomas Davidson, Dana Warmsley, Michael W. Macy, and Ingmar Weber. 2017. Automated Hate Speech Detection and the Problem of Offensive Language. In *Proc. of ICWSM*. 512–515.
[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of NAACL-HLT*. Minneapolis, Minnesota, 4171–4186.
[6] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and Mitigating Unintended Bias in Text Classification. In *Proc. of AIES*, Jason Furman, Gary E. Marchant, Huw Price, and Francesca Rossi (Eds.). 67–73.
[7] Tommi Gröndahl, Luca Pajola, Mika Juuti, Mauro Conti, and N. Asokan. 2018. All You Need is "Love": Evading Hate Speech Detection. In *Proc. of AISec* (Toronto, Canada). 2–12.
[8] Xisen Jin, Zhongyu Wei, Junyi Du, Xiangyang Xue, and Xiang Ren. 2020. Towards Hierarchical Importance Attribution: Explaining Compositional Semantics for Neural Sequence Models. In *Proc. of ICLR*.
[9] Brendan Kennedy, Xisen Jin, Aida Mostafazadeh Davani, Morteza Dehghani, and Xiang Ren. 2020. Contextualizing Hate Speech Classifiers with Post-hoc Explanation. In *Proc. of ACL*. 5435–5442.
[10] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*, Yoshua Bengio and Yann LeCun (Eds.).
[11] Binny Mathew, Punyajoy Saha, Hardik Tharad, Subham Rajgaria, Prajwal Singhania, Suman Kalyan Maity, Pawan Goyal, and Animesh Mukherjee. 2019. Thou shalt not hate: Countering online hate speech. In *Proc. of ICWSM*, Vol. 13. 369–380.
[12] Marzieh Mozafari, Reza Farahbakhsh, and Noël Crespi. 2019. A BERT-Based Transfer Learning Approach for Hate Speech Detection in Online Social Media. In *Proc. of COMPLEX NETWORKS*. 928–940.
[13] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English Tweets. In *Proc. of EMNLP*. Online, 9–14.
[14] Björn Ross, Michael Rist, Guillermo Carbonell, Benjamin Cabrera, Nils Kurowsky, and Michael Wojatzki. 2016. Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis. In *Proc. of NLP4CMC (Bochumer Linguistische Arbeitsberichte, Vol. 17)*, Michael Beißwenger, Michael Wojatzki, and Torsten Zesch (Eds.). Bochum, 6–9.
[15] Paul Röttger, Bertram Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, and Janet Pierrehumbert. 2020. HateCheck: Functional Tests for Hate Speech Detection Models. *arXiv preprint arXiv:2012.15606* (2020).
[16] Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The Risk of Racial Bias in Hate Speech Detection. In *Proc. of ACL*, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). 1668–1678.

[17] Deven Shah, H. Andrew Schwartz, and Dirk Hovy. 2020. Predictive Biases in Natural Language Processing Models: A Conceptual Framework and Overview. In *Proc. of ACL*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (Eds.). 5248–5264.

[18] Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Towards Debiasing NLU Models from Unknown Biases. In *Proc. of EMNLP*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). 7597–7610.

[19] Bertie Vidgen, Alex Harris, Dong Nguyen, Rebekah Tromble, Scott Hale, and Helen Margetts. 2019. Challenges and frontiers in abusive content detection. Association for Computational Linguistics.

[20] Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal Adversarial Triggers for Attacking and Analyzing NLP. In *Proc. of EMNLP*. 2153–2162.

[21] Zeerak Waseem. 2016. Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter. In *Proc. of NLP+CSS@EMNLP*, David Bamman, A. Seza Dogruöz, Jacob Eisenstein, Dirk Hovy, David Jurgens, Brendan O'Connor, Alice Oh, Oren Tsur, and Svitlana Volkova (Eds.). 138–142.

[22] Zeerak Waseem, Thomas Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding Abuse: A Typology of Abusive Language Detection Subtasks. In *Proc. of ALW1*. 78–84.

[23] Zeerak Waseem and Dirk Hovy. 2016. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In *Proc. of SRW@HLT-NAACL*. 88–93.

[24] Zeerak Waseem, James Thorne, and Joachim Bingel. 2018. Bridging the gaps: Multi task learning for domain transfer of hate speech detection. In *Online Harassment*. 29–55.

[25] Michael Wiegand, Josef Ruppenhofer, and Thomas Kleinbauer. 2019. Detection of Abusive Language: the Problem of Biased Datasets. In *Proc. of NAACL-HLT*. 602–608.

[26] Savvas Zannettou, Mai Elsherief, Elizabeth Belding, Shirin Nilizadeh, and Gianluca Stringhini. 2020. Measuring and Characterizing Hate Speech on News Websites. In *12th ACM Conference on Web Science* (Southampton, United Kingdom) *(WebSci '20)*. Association for Computing Machinery, New York, NY, USA, 125–134. https://doi.org/10.1145/3394231.3397902

[27] Guanhua Zhang, Bing Bai, Junqi Zhang, Kun Bai, Conghui Zhu, and Tiejun Zhao. 2020. Demographics Should Not Be the Reason of Toxicity: Mitigating Discrimination in Text Classifications with Instance Weighting. In *Proc. of ACL*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (Eds.). 4134–4145.