# Transfer Learning in Abusive Language Classification

**Ji Ho Park**
jhpark@connect.ust.hk

**Jamin Shin**
jmshinaa@connect.ust.hk

## Abstract

Recent work in abusive language detection has been mainly focusing on dataset collection and improving performance in each datasets. In our work, we explore the applications of transfer learning in abusive language classification. We first take an approach of *domain adaptation* as we hypothesize that each dataset collected from different social media can be viewed as separate domains. However, our analysis and results show that due to sampling noise within the dataset the hypothesis does not hold. Hence, we then focus on measuring and mitigating *unintended bias*, an effect caused by such sampling noises, with transfer learning.

## 1 Introduction

Automatic detection of abusive language is an important task since such language in online space can lead to personal trauma, cyber-bullying, hate crime, and discriminations. As more and more people freely express their opinions in social media, the amount of textual contents produced every day grows almost exponentially, rendering it difficult to effectively moderate user contents. For such reasons, leveraging big data and learning algorithms to develop an automatic abusive language detection system has already established its usefulness.

However, the problem is very complicated due to the inherent subjectivity of abusive language and difficulties of gathering and annotating datasets. Hence, although machine learning methods have shown promising results in classifying abusive language, many works recently have raised concerns of the robustness of those models related to noise in data, more specifically regarding unfair biases toward certain groups of people. We discuss those works in detail in Section 2.

In our work, we explore transfer learning methods to utilize different datasets together. Exploiting knowledge from other datasets is important in this problem since data collection and annotation are hard and expensive. We first use *domain adaptation* by adapting models trained on datasets from Wikipedia discussion pages to Twitter tweets.

We then address the unintended bias specific to gender identity terms existing in downstream abusive language datasets by conducting experiments to measure the biases and propose methods to effectively mitigate them. We evaluate our novel approach, which is to transfer knowledge from a larger and less-biased corpus to a build robust model invariant to bias in other datasets, by comparing with mitigation methods of previous works. We believe that correctly handling such bias is not only directly related to the generalization capability of learning algorithms, but also especially crucial to the performance of the model in production stage, in which such errors could be fatal.

The main contributions of our work first lies in the attempt of defining domains in abusive language data and applying *domain adaptation* methods to address in-domain data scarcity issues by leveraging out-of-domain corpus. We also propose a novel approach to mitigate bias inherent in small datasets by transferring knowledge from less-biased datasets.

## 2 Related Works

### 2.1 Abusive Language

Recently many efforts were put into defining and constructing abusive language datasets, mainly gathered in Twitter, Wikipedia forum discussions, or news comments and labeled through crowdsourcing or user moderation, (Djuric et al., 2015;

Nobata et al., 2016; Djuric et al., 2015; Waseem and Hovy, 2016; Waseem, 2016; Wulczyn et al., 2017; Founta et al., 2018). Driven by the enhancement of public resources, research on automatically detecting abusive language has also quickly gained momentum. Many feature-engineering based classification methods were proposed together with these datasets, Djuric et al. (2015); Nobata et al. (2016); Waseem (2016), mainly focused on word or character n-gram features.

Meanwhile, other approaches including deep learning have also been explored recently. Park and Fung (2017) systematically compared previous feature-engineering based methods such as n-gram based Support Vector Machines and Logistic Regression with both character-based and/or word-based Convolutional Neutral Networks (CNN) (Kim, 2014; Zhang et al., 2015) on the Sexist/Racist Tweets dataset Waseem (2016); Waseem and Hovy (2016), whereas Badjatiya et al. (2017) used not only deep learning but also tree-based gradient boosting methods to solve the same task. Pavlopoulos et al. (2017) applied Recurrent Neural Networks (RNNs) with self attention mechanism to not only improve performance of classification, but also visualizing and highlighting offensive words to design a semi-automatic user content moderation system.

However, several works discovered the limitations of current abusive language detection systems. Hosseini et al. (2017) showed that Google Perspective API, which detects toxic comments, is not robust enough to handle simple spelling errors and grammatic variation and has many false positives. Moreover, Dixon et al. (2017) was the first to address the problem of *unintended bias* inherent in such systems. Along with the potential detrimental effects on performance of deployed models, they introduced several metrics to evaluate and simple methods to mitigate such biases.

### 2.2 Transfer Learning

In recent years, transfer learning has gained wide interest from researchers as real data often do not arise in the same feature space and distribution as the training data. The specific case having only labeled data in source domain and adapting to the target domain with same or similar tasks is referred to as *Domain Adaptation* (Pan and Yang, 2010). For sentiment analysis on Amazon Product Reviews, Domain Adversarial Neural Networks (DANN) has been widely researched using adversarial training between different domains (i.e. Baby products, kitchen, electronics) to create more robust models that can create common representations between domains (Ajakan et al., 2014; Ganin et al., 2016).

In our work, we employ DANN to transfer abusive language knowledge from domains with larger datasets such as Wikipedia comments to relatively smaller ones like Sexist/Racist Tweets. Furthermore, we analyze downstream abusive language datasets to measure and mitigate inherent unintended biases.

## 3 Datasets

We use three datasets for training and evaluation: Wikipedia Attacks (`wiki`) dataset (Wulczyn et al., 2017), Sexist/Racist Tweets (`srt`) dataset (Waseem, 2016), and Abusive Tweets (`at`) dataset (Founta et al., 2018). As shown in Table 1, all datasets were constructed with different definitions of abusive language, so they all have different labels for abusive language. As we mainly aim to transfer knowledge from the largest dataset, the Wikipedia Attacks dataset, we also cast other datasets to a binary problem of detecting abusive language. We split all three datasets into train/valid/test of 80/10/10% and denote each as $D^{\{train,valid,test\}}_{\{\texttt{wiki},\texttt{srt},\texttt{at}\}}$ (i.e. $D^{train}_{\texttt{wiki}}$ or $D^{test}_{\texttt{srt}}$).

### 3.1 Wikipedia Attacks

The Wikipedia Attacks dataset contains approximately 115K comments from English Wikipedia discussion pages of articles. These comments were labeled as personal attacks by at least 10 annotators for each comment. For this dataset, we follow the approaches of Wulczyn et al. (2017); Pavlopoulos et al. (2017) using probabilistic labels, in which the percentage of agreement between annotators are used as probabilistic scores of each comment being a 'Abusive' or not. Hence, the annotations, instead of being discrete, form an empirical distribution over the opinions of annotators. Note that the original dataset uses the label 'attack' while we use 'abusive' for positive labels for consistency with other datasets.

### 3.2 Sexist/Racist Tweets

The Sexist/Racist Tweets dataset, as its name suggests, is consisted of tweets with sexist and racist tweets collected from Twitter by searching for

| Name | Size | Labels | Positives (%) | $\mu$ | $\sigma$ | $max$ |
|---|---|---|---|---|---|---|
| Sexist/Racist Tweets | 18K | None vs Sexist/Racist | 33% | 15.6 | 6.8 | 39 |
| Abusive Tweets | 60K | None/Spam vs Abusive/Hateful | 20% | 17.9 | 4.6 | 65 |
| Wikipedia Attacks | 115K | None vs Abusive (Attack) | 11.7% | 70.8 | 128.8 | 2832 |

Table 1: Dataset statistics. $\mu, \sigma, max$ are mean, standard deviation, and maximum of sentence lengths

tweets that contain common terms pertaining to sexism and racism such as 'feminazi' or 'islam terrorism'. The tweets were then annotated by experts based on a criteria founded in critical race theory (Waseem and Hovy, 2016), and Waseem (2016) further extended this dataset with same methodology. The combined dataset provided by Waseem (2016) contains approximately 18K tweets with around 2K racist tweets and 4K sexist tweets, hence, having around 6K tweets as 'Abusive'.

### 3.3 Abusive Tweets

Recently, Founta et al. (2018) has published a large scale crowdsourced abusive tweet dataset with 60K tweets. Their work incrementally and iteratively investigated methods such as boosted sampling and exploratory rounds, to effectively annotate tweets through crowdsourcing. Through such systematic processes, they identify the most relevant label set in identifying abusive behaviors in Twitter as $\{None, Spam, Abusive, Hateful\}$ resulting in 11% as 'Abusive', 7.5% as 'Hateful', 22.5% as 'Spam', and 59% as 'None'. Again, we transform this dataset for a binary classification problem by concatenating 'None' and 'Spam' together as negative, and 'Abusive' and 'Hateful' together as positive, as shown in Table 1. For more detailed information on their methodology, we refer any interested readers to the works of Founta et al. (2018).

## 4 Domain Adversarial Neural Networks

**Domains in Abusive Language** Domains in *Domain Adaptation* are not what we typically use in math; they instead simply refer to data distributions, so if two datasets have different marginal distributions or feature spaces, they are considered different domains (Ben-David et al., 2007; Pan and Yang, 2010; Ajakan et al., 2014). We hypothesize that different abusive datasets collected and annotated differently will have different feature spaces and very different vocabularies.

In general, Twitter tweets and Wikipedia comments have different topics, as tweets are often more focused on current news, and Wikipedia comments are more about contents of Wikipedia articles. The fact that "mkr" or "gamergate" is not considered abusive in Wikipedia comments but are abusive, or sexist, in tweets serves as an example of such hypothesis. Another example would be that tweets with the word 'trump' tend to be abusive, whereas it is not the case for Wikipedia comments. Hence, we define datasets from different social media as different domains.

**Domain Adaptation** Formally, we have two different data distributions, the *source domain* $D_{\text{S}}$ and the *target domain* $D_{\text{T}}$. We define the input $\mathbf{X} \in \mathbb{R}^n$ and a binary output $\mathbf{Y} = \{0, 1\}$ for simplicity. Finally, we define a classifier $f : \mathbf{X} \rightarrow \mathbf{Y}$ as such:

$$Pr_{\mathbf{x}^t, d \sim D_{\text{T}}}(\hat{d} = d | \mathbf{z}^t) = f^t(\mathbf{z}^t)$$

where $\mathbf{z}^t$ is the hidden representation of $\mathbf{x}^t$,

$$\mathbf{z}^t = h(\mathbf{x}^t)$$

and $h$ is any feature extractor such as RNN.

The objective of this learning algorithm is simply to train $f$ only with *labeled i.i.d* samples drawn from $D_{\text{S}}$ and minimize the risk with *unlabeled i.i.d* samples drawn from $D_{\text{T}}$.

### 4.1 Methodology

We apply the model used in sentiment analysis tasks introduced by Ajakan et al. (2014); Ganin et al. (2016), Domain Adversarial Neural Network (DANN). The model architecture of DANN is composed of three components: *feature extractor*, *label predictor*, and *domain classifier*.

The *feature extractor* can be any type of neural network that takes in an input sequence $\mathbf{x}^s \sim D_{\text{S}}$ and $\mathbf{x}^t \sim D_{\text{T}}$. We have chosen to use a Gated Recurrent Unit (GRU) and denote it as $h$, which is a variant of RNN introduced to the literature by Cho et al. (2014). We use the last hidden state of

the GRU as our hidden representations of samples drawn from the *source domain* and *target domain*, namely $\mathbf{z}^s = h(\mathbf{x}^s)$ and $\mathbf{z}^t = h(\mathbf{x}^t)$.

The *label predictor*, denoted as $f^y$ can also be any type of classifier such as logistic regression unit or a multilayer perceptron, which, in our case, is chosen to be simple logistic regression as the feature extractor is a GRU. It takes in $\mathbf{z}^s$ as input and is defined as such:

$$Pr_{\mathbf{x}^s, y \sim D_{\text{S}}}(\hat{y} = y | \mathbf{z}^s) = f^y(\mathbf{z}^s)$$

where $y \in \{0, 1\}$, the class label.

Finally, the *domain classifier*, denoted as $f^d$ can also be any type of classifier such as logistic regression unit or a multilayer perceptron, which we also chose as a logistic regression unit for simplicity. It receives either $\mathbf{z}^s$ or $\mathbf{z}^t$ as input and is defined as such:

$$Pr_{\mathbf{x}, d \sim D_{\{\text{S,T}\}}}(\hat{d} = d | \mathbf{z}) = f^d(\mathbf{z})$$

where $d \in \{\text{S}, \text{T}\}$, the domain label.

The first two components, the *feature extractor* and *label predictor*, already form a standard neural network architecture for classification, and they are trained similarly. However, the domain classifier, is trained in an adversarial manner, with a simple *gradient reversal* layer between the *feature extractor* and the *domain classifier*. This *gradient reversal* layer reverses the gradient that flows from the *domain classifier* to the *feature extractor*, hence the name 'adversarial'.

In short, the intuition of DANN is to train a strong *label predictor* and a strong *domain classifier* given the feature representations of $\mathbf{x}^s \sim D_{\text{S}}$ and $\mathbf{x}^t \sim D_{\text{T}}$, while the GRU *feature extractor* is optimized to extract features that minimize the loss $\mathcal{L}_y(\mathbf{x}^s)$ from the *label predictor*, and maximizing the losses $\mathcal{L}_d(\mathbf{x}^s)$ and $\mathcal{L}_d(\mathbf{x}^t)$ from the *domain classifier*. More formally, the objective function is:

$$\min_{\theta_h, \theta_y} \Bigg[ \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}_y(\mathbf{x}^s) + \lambda \max_{\theta_d} \Big( - \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}_d(\mathbf{x}^s) - \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_d(\mathbf{x}^t) \Big) \Bigg]$$

where both loss functions are cross entropy.

### 4.2 Experiment Settings

As the main objective is to transfer knowledge from a large dataset, we mainly train the classifier with Wikipedia Attacks dataset and evaluate on the two other smaller ones from Twitter. Based on the theoretical results of Ben-David et al. (2007) on generalization bounds in *domain adaptation*, we define the baselines as lower bound and upper bound for each datasets `srt` and `at`. For the upper bound, we simply train on the *target domain* data and evaluate on *target domain*, while the lower bound is training on *source domain*. For experiments with DANN, we train with $D^{train}_{\texttt{wiki}}$ for label classification and both *source* and *target domain* data for domain classification. The experiment settings are summarized in Table 2.

| Experiment Name | Train | Valid | Test |
|---|---|---|---|
| `srt` UPPER | $D^{train}_{\texttt{srt}}$ | $D^{valid}_{\texttt{srt}}$ | $D^{test}_{\texttt{srt}}$ |
| `srt` DANN | $D^{train}_{\texttt{wiki}}$ | $D^{valid}_{\texttt{srt}}$ | $D^{test}_{\texttt{srt}}$ |
| `srt` LOWER | $D^{train}_{\texttt{wiki}}$ | $D^{valid}_{\texttt{srt}}$ | $D^{test}_{\texttt{srt}}$ |
| `at` UPPER | $D^{train}_{\texttt{at}}$ | $D^{valid}_{\texttt{at}}$ | $D^{test}_{\texttt{at}}$ |
| `at` DANN | $D^{train}_{\texttt{wiki}}$ | $D^{valid}_{\texttt{at}}$ | $D^{test}_{\texttt{at}}$ |
| `at` LOWER | $D^{train}_{\texttt{wiki}}$ | $D^{valid}_{\texttt{at}}$ | $D^{test}_{\texttt{at}}$ |

Table 2: Experiment setup. Validation set is used for adversarial training in DANNs.

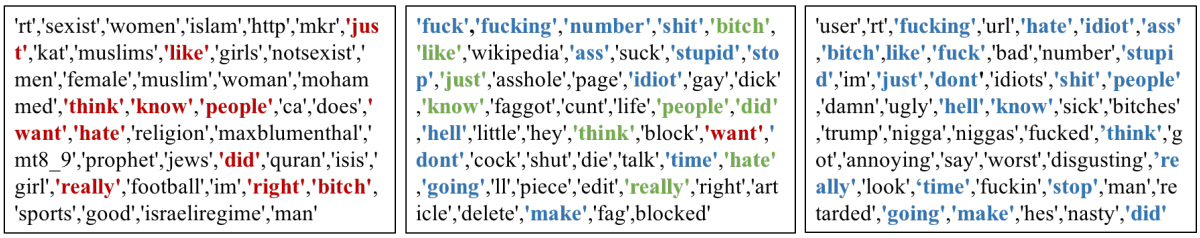
| Experiment | F1 |
|---|---|
| `srt` UPPER | 0.710 |
| `srt` LOWER | 0.294 |
| `at` UPPER | 0.779 |
| `at` LOWER | 0.762 |

Table 3: Experiment results.

### 4.3 Results & Discussion

Interestingly, Table 3 shows that there is a very small margin of difference between the performances in the lower and upper bounds of `at` dataset. Note that, although not reported, our domain adversarial approach fails improve from the lower bound baseline for both datasets. This result indicates that our initial hypothesis mentioned in Section 4 that different datasets from different social media can be considered as different domains in abusive language classification, might not hold.

Taking a deeper look into the marginal distributions of `wiki` and `at` datasets, shown in Figure 1, qualitatively, we can see that there is a lot of overlapping "abusive" words between the two datasets. This implies that while the distribution




(a) Sexist/Racist Tweets (b) Wikipedia Attacks (c) Abusive Tweets

Figure 1: Comparison of words with top TF-IDF scores of abusive samples. Blue words are overlaps of `wiki` and `at`, Red words are overlaps of `wiki` and `srt`. Green words are words that overlap in all three.

over the entire vocabulary may be different for the two datasets, but the marginal distribution over abusive samples show a very similar result, leading to a very similar feature space. Hence, it is safe to say that the two datasets are practically sampled from the same domain, which clearly explains our results.

On the other hand, results on the `srt` dataset show the exact opposite trend, in which the performance difference between the lower and upper bounds is large, while still having unimproved results for DANNs. Running the same analysis on these datasets, Figure 1 shows the opposite trend as before, in which there is very little overlap in "bad" words between the `wiki` and `srt` datasets. Based on the poor results of DANN and the comparison shown in Figure 1, we conjecture that these two datasets show too different marginal distributions and feature spaces, and that they can be even considered as different tasks.

However, although our analysis has effectively nullified our initial hypothesis, referring back to our observations from Section 4, we still believe that abusive language will show different behavior under different social media contexts. The caveats can be mainly attributed to the inherent noise that arise from small dataset sizes as we do not have enough examples that show such different behaviors among the domains. Data collection and annotation methods may have also affected such results because categories like sexism and racism are very subjective, so mostly profane words annotated as abusive.

## 5 Measuring Unintended Biases

**Unintended Bias** While working with these datasets we found out that the classifiers trained from some of these datasets are not easy to use for practical application since they tend to overfit to certain words that are neutral, but occur frequently in the training samples. For example, the sentence "You are a good woman" was given a high probability for being sexist due to the word "woman". This phenomenon, called *false positive bias*, has already been reported from the previous work (Dixon et al., 2017) on abusive language detection. They further define a broader term *unintended bias* as "a model contains unintended bias if it performs better for comments containing some particular identity terms than for comments containing others". In this work, we will follow this definition but focus on gender identity terms.

Although such biases frequently exist in abusive language tasks due to its problem nature and datasets imbalance, how to measure and alleviate them is still an ongoing research. First, we perform a comparative analysis of unintended biases several models trained with different datasets and different pretrained word embeddings. Furthermore, we explore such methods as data augmentation and transfer learning to mitigate them.

### 5.1 Methodology

Unintended bias cannot be measured when the evaluated on the original dataset as the test sets will follow the same biased distribution, so normal evaluation set will not sufficiently measure any unintended bias. Therefore, we generate a separate *unbiased test set* for two gender identity groups, male and female, using the identity term template method proposed in Dixon et al. (2017).

The intuition of this template method is that given a pair of sentences with only the identity terms different (ex. "He is happy" & "She is happy"), the model should be able to generalize well and output same prediction for abusive language. This kind of evaluation has also been per-

formed in *SemEval 2018: Task 1 Affect In Tweets* (Mohammad et al., 2018) to measure the gender and race bias among the competing systems for sentiment/emotion analysis.

Using the released code[1] of Dixon et al. (2017), we generated 1152 samples (576 pairs) by filling the templates with common gender identity pairs (ex. male/female, man/woman, etc.). We created templates that contained both neutral and offensive nouns and adjectives inside the vocabulary to retain balance in neutral and abusive samples.

For the evaluation metric, we use both the AUC scores on original test set (Orig. AUC), unbiased generated test set (Gen. AUC), and the error rate equality difference proposed in Dixon et al. (2017) which aggregates the difference between the overall false positive/negative rate and gender-specific false positive/negative rate. False Positive Equality Difference (FPED) and False Negative Equality Difference (FNED) are defined as below, where $T$ includes *male* and *female*. Since the classifiers output probabilities, equal error rate thresholds are used for inference.

$$FPED = \sum_{t \in T} |FPR - FPR_t|$$

$$FNED = \sum_{t \in T} |FNR - FNR_t|$$

Whereas the two AUC scores show the performances of the classification models in terms of accuracy, the two equality difference scores show them in terms of fairness, which we believe is another dimension for evaluating the model's generalization capability.

### 5.2 Experiment Setting

We first measure unintended biases in two `srt` and `at` datasets. For the former, we only keep the "sexist" samples to focus on biases related to gender. We then explore three neural models commonly used in abusive language classification: Convolutional Neural Network (CNN) (Park and Fung, 2017), Recurrent Neural Network (RNN), and Bidirectional RNN with self-attention ($\alpha$-RNN). For both types of RNNs, we use Gated Recurrent Units (GRU) (Cho et al., 2014) same as Pavlopoulos et al. (2017) did. However, their multilayer attention did not work well so we opted to a simpler self-attention mechanism used in Felbo et al. (2017). Moreover, we compare different pretrained embeddings, `word2vec` (Mikolov et al., 2013) trained from Google News corpus and `fasttext` (Bojanowski et al., 2016)) trained on Wikipedia corpus, to analyze their effects on unintended bias. Randomly initialized word embeddings (*random*) are also evaluated to serve as a baseline. The texts are all preprocessed [2]. Finally, all experiments were run 10 times and averaged.

| Model | Embed. | Orig. ROC | Gen. ROC | FNED | FPED |
|---|---|---|---|---|---|
| CNN | random | .881 | .572 | .261 | .249 |
| | fasttext | **.906** | .620 | .323 | .327 |
| | word2vec | **.906** | .635 | .305 | .263 |
| RNN | random | .854 | .536 | **.132** | **.136** |
| | fasttext | .887 | .633 | .301 | .254 |
| | word2vec | .887 | .633 | .301 | .254 |
| $\alpha$-RNN | random | .868 | .586 | .236 | .219 |
| | fasttext | .891 | **.639** | .324 | .365 |
| | word2vec | .890 | .631 | .315 | .306 |

Table 4: Results on Sexist Tweets dataset. Note that false negative/positive equality difference tends to be larger when pretrained embedding is used and CNN or $\alpha$-RNN models is trained

| Model | Embed. | Orig. ROC | Gen. ROC | FNED | FPED |
|---|---|---|---|---|---|
| CNN | random | .926 | .893 | .013 | .045 |
| | fasttext | .955 | .995 | .004 | **.001** |
| | word2vec | **.956** | **.999** | **.002** | .021 |
| RNN | random | .919 | .850 | .036 | .010 |
| | fasttext | .951 | .997 | .014 | .018 |
| | word2vec | .952 | .997 | .017 | .037 |
| $\alpha$-RNN | random | .927 | .914 | .008 | .039 |
| | fasttext | **.956** | .998 | .014 | .005 |
| | word2vec | .955 | **.999** | .012 | .026 |

Table 5: Results on Abusive Tweets dataset. The false negative/positive equality difference is significantly smaller than the Sexist Tweets dataset

### 5.3 Results & Discussions

Tables 4 and 5 show the bias measurement experiment results for `srt` and `at`, respectively. As expected, pretrained embeddings improved task performance. The score on the unbiased generated test set (Gen. ROC) also improved since word embeddings can provide prior structural knowledge of words.

However, the equality difference scores tended

[1] https://github.com/conversationai/unintended-ml-bias-analysis

[2] https://github.com/jihopark/hltc_preprocess

to be larger when pretrained embeddings were used, especially in the `srt` dataset. The direction of the gender bias was towards female identity words. We can infer that this is due to the more frequent appearances of female identity words in "sexist" tweets and lack of negative samples in a small dataset, similar to the reports of Dixon et al. (2017). This is problematic since not many NLP datasets are large enough to reflect the true data distribution (sampling noise), more prominent in tasks like abusive language where data collection and annotation are difficult.

On the other hand, `at` dataset showed significantly better results on the two equality difference scores, of at most 0.04. This means that false negative/positive rate for a certain gender is less than 2%. Performance in the generated test set can achieve an almost perfect score because the models can successfully classify abusive samples regardless of the gender identity terms used. Hence, we can safely assume that `at` dataset is less biased towards certain gender than the `srt` dataset, probably due to its larger size, balance in classes, and systematic collection method.

Another interesting outcome was that the architecture of the models also influenced the unintended biases. Models that "attend" to certain words, such as CNN's max-pooling or $\alpha$-RNN's self-attention, tended to result in higher false positive equality difference scores in `srt` dataset. These models show effectiveness in catching not only the intended biases useful for the task, but also the unintended biases of inherent in the data.

## 6 Mitigating Biases

### 6.1 Methodology

So far we were able to identify the existence of an issue. Naturally, we explore three different ways to reduce unintended bias.

**Debiased Word Embeddings** Pretrained word embeddings (Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2016) are widely used for many downstream NLP tasks. However, previous work (Bolukbasi et al., 2016) shows the existence of gender biases in these embeddings. Our analysis discussed in Section 5.3 also confirms that pretrained word embedding may push the unintended biases further. They propose an algorithm to correct those embeddings by removing gender stereotypical information.

**Gender swapping data augmentation** We augment the training data by identifying male entities and swapping them with equivalent female entities and vice-versa. This simple method removes correlation between gender and classification decision, and has proven to be effective for correcting gender biases in co-reference resolution task (Zhao et al., 2018).

**Bias fine-tuning** We propose a method to use transfer learning from a less biased corpus to reduce unintended bias. The model is initially trained with a larger source corpus with the same or similar task and with less unintended bias, and fine-tuned with a target corpus with a larger bias. This method is inspired by the fact that unintended bias can come from the imbalance of labels and the limited size of data samples. Training the model with a larger and less biased dataset can effectively regularize and prevent the model from overfitting to the small, biased dataset.

### 6.2 Experiment Setting

`debiased word2vec` released by Bolukbasi et al. (2016) is compared with original `word2vec` for evaluation. For gender swapping data augmentation, we use gender term pairs identified through crowd-sourced annotations by Zhao et al. (2018).

After identifying the degree of unintended bias of each dataset, we select a source dataset with less bias and a target dataset with more bias. Vocabulary is extracted from training split of both sets. The model is first trained by the source training dataset until convergence. We the remove final softmax layer attach a new one initialized for training the target task. The target dataset is trained with a slower learning rate until it converges. Early stopping is decided by the validation set of the respective dataset.

Based on the above criterion and bias measurement results from Section 5.3, we choose the `at` dataset as source and `srt` dataset as the target for bias fine-tuning experiments.

### 6.3 Results & Discussion

Table 6 shows the results of the experiment using three methods proposed above. Using debiased word embedding alone deos not correct the bias of the whole system very well. Gender swapping data augmentation significantly reduced equality difference scores as shown in previous work for co-reference resolution task (Zhao et al., 2018). Bias fine-tuning with source dataset helped to improve ROC scores from generated unbiased test

| Model | Debiased Embed. | Gender Swap | Finetune | Orig. ROC | Gen. ROC | FNED | FPED |
|---|---|---|---|---|---|---|---|
| CNN | . | . | . | **.906** | .635 | .305 | .263 |
| | O | . | . | .902 | .627 | .333 | .337 |
| | . | O | . | .898 | .676 | .164 | .104 |
| | . | . | O | .896 | .650 | .302 | .240 |
| | . | O | O | .889 | .671 | .163 | .122 |
| | O | O | O | .884 | .703 | .135 | .095 |
| RNN | . | . | . | .887 | .633 | .301 | .254 |
| | O | . | . | .882 | .658 | .274 | .270 |
| | . | O | . | .887 | .645 | .287 | .258 |
| | . | . | O | .874 | .761 | .241 | .181 |
| | . | O | O | .862 | .768 | .141 | .095 |
| | O | O | O | .854 | .854 | .081 | .059 |
| $\alpha$-RNN | . | . | . | .890 | .631 | .315 | .306 |
| | O | . | . | .885 | .656 | .291 | .330 |
| | . | O | . | .879 | .667 | .114 | .098 |
| | . | . | O | .874 | .756 | .310 | .212 |
| | . | O | O | .866 | .814 | .185 | .065 |
| | O | O | O | .855 | **.912** | **.055** | **.030** |

Table 6: Results of bias mitigation methods on `srt` dataset. Combining all methods, $FPED$ and $FNED$ decreases 50-90%, while losing only 2.3-3.9% of original test set performance

set and decreased the equality difference scores to some extent, but it had the largest decrease in original test set performance. This could be attributed to the difference in the source and target tasks ("abusive" vs. "sexist"). However, performance decrease was marginal (at most 1-2%), while the drop in bias is quite significant.

All of these methods can easily be applied together since they try to tackle the problem in different ways. When all three methods are applied, false negative/positive equality decreases 50-90%, while losing only 2.3-3.9% of the original test set performance (different model architecture had a different degree of mitigation). Note that the bias mitigation methods all involved some performance loss when unintended biases were reduced. We assume this is because discriminative classification models like CNNs or RNNs cannot distinguish between intended bias (necessary for classification) and unintended bias, and the mitigation methods may confuse the models. Such performance loss may be prevented when mitigation or correction can be performed during training time, but we leave this for future work.

## 7 Conclusion

In this work, we explored the issues inherent in abusive language detection by reviewing available datasets and literature. We attempted to apply transfer learning methods to solve them. First, we experimented using Domain Adversarial Neural Networks to transfer knowledge among abusive language datasets gathered from different social media sources. However, we discovered that abusive language in `wiki` and `at` datasets share a too similar marginal distribution and feature space, whereas `at` and `srt` had not enough common features, mostly caused by sampling noise.

Moreover, we examined *unintended bias* in abusive language datasets in gender identity terms, another effect of sampling noise. We first created a template-based unbiased test set to measure biases, and reviewed metrics like $FPED$ and $FNED$ doneto quantify the biases in the dataset. Different neural models and pretrained embeddings were compared in `srt` and `at` datasets.

We revealed that pretrained embeddings can push the bias more, and certain model architectures can better capture unintended bias. Finally, we studied methods to mitigate those unintended biases and showed that they can significantly reduce the bias together, while only losing an insignificant amount of original task performance.

**Further Development** For future works, bias mitigation without any classification performance loss may be achieved by correcting the model during training time. Also, it would be desirable if more extensive measurements of unintended bias is performed for other subjective tasks such as sentiment/emotion analysis, so that we can be better evaluate the generalization capabilities of the models.

## References


Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. 2014. Domain-adversarial neural networks. *arXiv preprint arXiv:1412.4446*.

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760. International World Wide Web Conferences Steering Committee.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. 2007. Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *EMNLP2014*.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2017. Measuring and mitigating unintended bias in text classification.

Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate speech detection with comment embeddings. In *Proceedings of the 24th international conference on world wide web*, pages 29–30. ACM.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *EMNLP2017*.

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. *arXiv preprint arXiv:1802.00393*.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Hossein Hosseini, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. 2017. Deceiving google's perspective api built for detecting toxic comments. *arXiv preprint arXiv:1702.08138*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

Saif M Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018), New Orleans, LA, USA*.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web*, pages 145–153. International World Wide Web Conferences Steering Committee.

Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Ji Ho Park and Pascale Fung. 2017. One-step and two-step classification for abusive language detection on twitter. *ALW1: 1st Workshop on Abusive Language Online to be held at the annual meeting of the Association of Computational Linguistics (ACL) 2017*.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017. Deeper attention to abusive user content moderation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1125–1135.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP2014*, pages 1532–1543.

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.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

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. International World Wide Web Conferences Steering Committee.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. *NAACL 2018*.