Decoupling Adversarial Training for Fair NLP

Xudong Han **Timothy Baldwin** **Trevor Cohn**
School of Computing and Information Systems
The University of Melbourne
Victoria 3010, Australia
xudongh1@student.unimelb.edu.au {tbaldwin,tcohn}@unimelb.edu.au

Abstract

Adversarial debiasing can help to learn fairer models. Previous work has assumed that both main task labels and protected attributes are available in the dataset. However, protected labels are often unavailable, or only available in limited numbers. In this paper, we propose a training strategy which needs only a small volume of protected labels in adversarial training, incorporating an estimation method to transfer private-labelled instances from one dataset to another. We demonstrate the in- and cross-domain effectiveness of our method through a range of experiments.

1 Introduction

Protected attributes such as user gender can act as confounding variables in models, and spurious correlations with task response variables can lead to unfair predictions, as seen in tasks such as part-of-speech tagging (Hovy and Søgaard, 2015), hate speech detection (Huang et al., 2020), and sentiment analysis (Kiritchenko and Mohammad, 2018).

Adversarial methods are a popular method for mitigating bias associated with protected attributes, wherein the encoder attempts to prevent a discriminator from identifying protected attributes (Zhang et al., 2018; Li et al., 2018; Han et al., 2021). An adversarial network consists of a discriminator A and an encoder E. Each input x_i is required to be annotated with both a main task label y_i and protected attribute label g_i, and the discriminator identifies protected information in the representation generated by the encoder ($\hat{g}_i = A(h_i)$). The objective of the encoder training incorporates two parts: (1) predicting the main task label ($\hat{y}_i = C(E(x_i))$); and (2) preventing protected attributes from being identified by the discriminator.

An important limitation of previous adversarial debiasing work is that training instances must be annotated with both main task and protected labels (Li et al., 2018; Wadsworth et al., 2018; Zhang et al., 2018; Wang et al., 2019; Han et al., 2021). However, sourcing protected labels can be difficult, for reasons ranging from privacy regulations/ethical concerns, to only a small subset of users explicitly publicly disclosing protected attributes.

Our contributions are as follows: (1) we present a novel way of training the main task model and the discriminator separately, removing the restriction that every training instance needs to be annotated with protected labels; (2) we conduct in-domain experiments with diminishingly small amounts of protected-labelled data for sentiment analysis and hate speech detection, and show that our method can be successfully applied with remarkably little protected data; and (3) we present preliminary results for cross-domain transfer of protected attributes for sentiment analysis and POS tagging. The source code and data associated with this paper are available at: https://github.com/HanXudong/Decoupling_Adversarial_Training_for_Fair_NLP.

2 Methodology

Adversarial Separation Training Intuitively, adversarial supervision can be decoupled from the main task training, i.e., the inputs used for training the main model do not have to be annotated with protected labels. In doing so, we attain flexibility in being able to train the discriminator over only those instances where we have access to the protected attribute, as well as being able to transfer private attributes between datasets. Following the setup of Li et al. (2018), the optimisation objective is:

$$\min_{E,C} \max_{A} \sum_{i \in \mathcal{D}_{\text{main}}} \mathcal{X}(y_i, \hat{y}_i(E, C)) - \lambda_{\text{adv}} \sum_{j \in \mathcal{D}_{\text{adv}}} \mathcal{X}(g_j, \hat{g}_j(E, A)),$$

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 471–477

Algorithm 1: Predictability Estimation

Input: Out of domain dataset $D_O = (X_O, G_O)$, pretrained main task model M_I in the target domain, $\mathcal{M}$, number of folds k, number of test folds at each step t

Output: protected label predictability scores, $\mathcal{P}$

```
Calculate hidden representations H_O of X_O from M_I
Partition D_O into k equi-sized folds as {F_1, ..., F_k}
Create (k choose t) test fold combinations as T
for i ← to (k choose t) do
    Use T_i as F_test, remaining folds as F_train
    Train M on F_train
    forall (h_{O,j}, g_{O,j}) ∈ F_test do
        Make prediction ĝ_{O,j} = M(h_{O,j})
        Accumulate correct predictions
          P(j) += δ(ĝ_{O,j}, g_{O,j})
    end
end
return P
```

where $\mathcal{X}$ is cross entropy loss, and λ_{adv} is a tunable hyperparameter. The critical observation of this work is that the two sources of data $\mathcal{D}_{\text{main}}$ and $\mathcal{D}_{\text{adv}}$ need not be the same, but may overlap or be entirely disjoint, as we explore in Section 3.

Filtering Cross-domain Data The inputs used for discriminator training do not have to be annotated with the main task label. Inspired by the domain robustness results of Li et al. (2018) with adversarial training, we examine cross-domain adversarial learning where protected labels are unknown for the target task in two settings: (1) sentiment analysis classification, and (2) part of speech (POS) tagging. In both cases we use external race labels from a hate speech dataset, and ignore any protected attributes in the original dataset.

According to our experiments, one problem associated with using cross-domain protected-labelled data is that some protected labels may not be relevant to the target domain. To address this problem, inspired by adversarial filtering (Le Bras et al., 2020), we conduct preliminary exploration on filtering cross-domain data in adversarial separation training. This method finds out-of-domain instances with the most confident predictions of the protected label, and selects these instances to use as a silver standard in training in-domain adversaries.

To estimate the protected label predictability of a cross-domain instance $(x_{O,i}, g_{O,i})$ in the target domain given a trained main task model in the target domain (M_I) and an estimator $\mathcal{M}$ (a logistic regression classifier), the protected label predictability of each instance is estimated as shown in Algorithm 1. Specifically, for folds k and test folds t, the predictability of each instance is estimated n times (i.e., by n different models) over different training sets. n can be derived from k and t as follows: $n = \frac{(k-1)!}{(k-t)!(t-1)!}$. Note that when $t = 1$, our method equates to k-fold cross-validation, and the predictability of each instance is estimated once. We demonstrate how the estimated predictability $\mathcal{P}$ can be used in Section 3.5.

3 Experiments and Analysis

In this section, we report on experiments under two scenarios: (1) in-domain, where protected labelled data and main labelled data are from the same domain; and (2) cross-domain, where protected labelled data are from a different domain to the main task data.

3.1 GAP

A common way of measuring fairness is GAP: the absolute difference for a metric between data subsets selected by different settings of the protected attribute. For instance, in the binary setting, we can compare the true positive rate (TPR) for male- vs. female-authored documents in the test set; this difference is the TPR-GAP, and is zero for a fair model.

3.2 In-domain: Sentiment Analysis

Data We experiment with the dataset of Blodgett et al. (2016), which contains tweets that are either African American English (AAE)-like or Standard American English (SAE)-like (following Han et al. (2021)). Each tweet is annotated with a binary "race" label (based on AAE or SAE), and a binary sentiment score determined by the (redacted) emoji within it.

We use the train/dev/test splits from Han et al. (2021) of 100k/8k/8k instances, respectively. The full dataset is artificially balanced across the four race–sentiment combinations. To (re)introduce bias into the dataset, previous work has skewed the training data to generate race–sentiment combinations (AAE–happy, SAE–happy, AAE–sad, and SAE–sad) of 40%, 10%, 10%, and 40%, respectively, leaving the dev and test data unbiased.

To examine how much private labelled data is needed, we randomly mask the protected attribute from up to 99% of the training data.

Models We use the same model architecture as Han et al. (2021), in the form of the fixed-parameter DeepMoji encoder (Felbo et al., 2017)

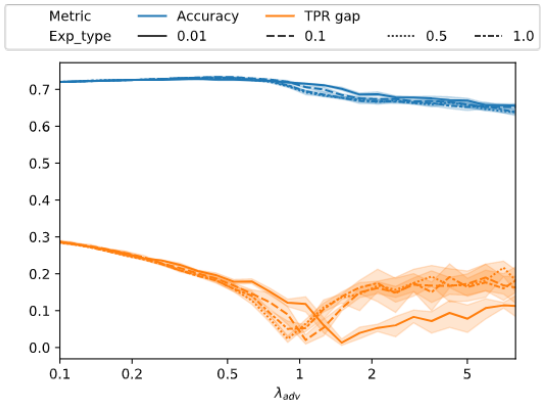


Figure 1: In-domain Sentiment Analysis: main task accuracy and GAP with respect to the trade-off hyperparameter λ_{adv}; shaded areas = 95% CI estimated over 5 runs.

followed by a trainable 3-layer MLP. DeeMoji contains 22.4 million parameters and is pretrained over 1246 million tweets to predict one of 64 common emojis. The discriminator for adversarial training (for all experiments in this paper) is trained to predict the protected attribute from the hidden states of the last hidden layer of the MLP classifier. Full training details are provided in the Appendix.

Results We explore 4 dataset settings where 100%, 50%, 10%, and 1% of the training data is labelled with its private attribute. We tune λ_{adv} log-uniformly under each data setting, using the same case-control training strategies for all experiments in this paper.

As shown in Figure 1, tuning λ_{adv} results in a series of candidate models, and there is a clear inflection point for the TPR GAP under each data setting, at different values of λ_{adv}.

To compare the adversarial training performance across different numbers of private labels, we show the trade-off plot in Figure 2a. Each point reflects the Accuracy and TPR GAP of a candidate model with a given λ_{adv}. The points for the three data settings of 100%, 50%, and 10% are hard to separate, indicating that adversarial training with only 10% of protected labels can achieve similar results to using protected labels for 100% of the data. Even with 1% of private labels, debiasing is evident, but this comes at a lower accuracy for a given TPR GAP level.

3.3 In-domain: Hate Speech Detection

Data Our second in-domain dataset is the English Twitter hate speech detection dataset of Huang et al. (2020), where each tweet is labelled with a binary hate speech label and also contains (binary) demographic indicators for the tweet author: binary gender (female or male), location (U.S. or other), age (older or younger than the median), and race (white or other). We focus on *age*, which has been shown to result in the greatest model unfairness (Huang et al., 2020), and use the train/dev/test splits of Huang et al. (2020). Since age information is not available for all authors, we downsample to get a subset of tweets which are annotated with age, with approximately 31k/6.7k/6.7k in training/dev/test.

Model Huang et al. (2020) compare 4 different model architectures for the hate speech detection task — TF-IDF-weighted feature-based logistic regression, convolutional neural network, an RNN (in the form of a biGRU), and BERT (Devlin et al., 2019) — and found the RNN model to consistently perform best. Based on this, we use the same RNN model, and perform debiasing on top of it.

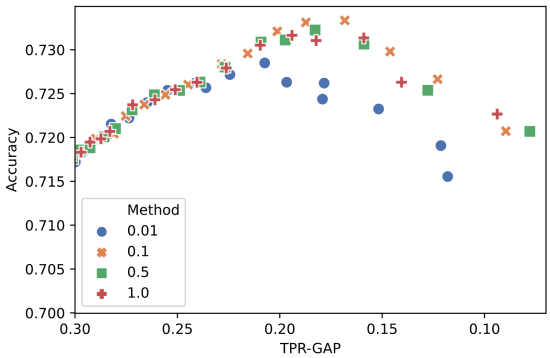
Results Figure 2b shows the trade-off plot with respect to hate speech detection models under similar data conditions as our first experiment (100%, 50%, 10%, and 0.1%). Consistent with our previous observations, there is little distinguishing 100%, 50%, and 10%. In fact, when we further decrease the proportion of protected labels, we observe that even with 0.1% of protected attributes, the trade-off is close to the 100% model.

3.4 Cross-domain: Sentiment Analysis

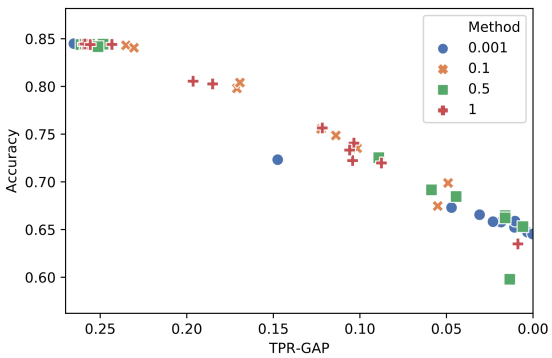
Next, we turn to the cross-domain setting, in taking protected labelled data from one domain and using it to adversarially debias a sentiment analyser over a different but similar domain.

Data We use the same model architecture and sentiment analysis dataset as in Section 3.2, but source private attributes (in the form of race) from the hate speech dataset from Section 3.3 to train the discriminator. Note that the race labels for the hate speech dataset (white or other) diverge slightly from those in the target domain (SAE or AAE).

Results Figure 3 shows the trade-off plot for the model. To compare the cross- and in-domain settings, we include in-domain adversarial debiasing results (1% data setting). Compared with in-domain results, the trade-off for cross-domain debiasing is worse than the 1% in-domain setting, but substantially better than random.



(a) Sentiment Analysis



(b) Hate Speech Detection

Figure 2: In-domain evaluation showing the accuracy–TPR-GAP trade-off for different fractions of protected-labels in the training dataset. For each data setting, we evaluate predictions for λ_{adv} settings near to that where the model achieves its smallest development TPR GAP. Since there is strong correlation between TPR GAP and TNR GAP, we only include TPR GAP.

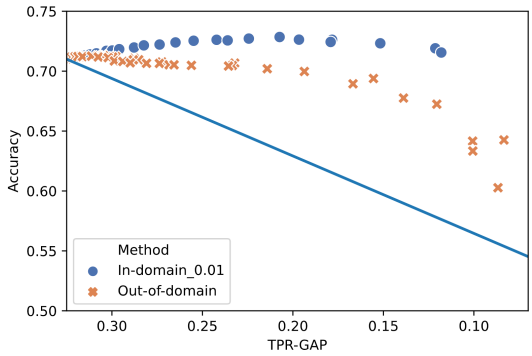


Figure 3: Cross-domain Sentiment Analysis: trade-off plot. The blue solid line denotes baseline debiasing results, based on randomly replacing main task predictions with a Bernoulli r.v. sampled from $p = 0.5$.

In terms of the drop in model accuracy during debiasing, each point in Figure 3 corresponds to a candidate model, and in this cross-domain setting, some models (e.g., for those models with TPR-GAP around 0.2) are able to reduce the bias by about 50% while maintaining performance that is close to the vanilla model. Managing the trade-off relates to model selection and the requirements of a given application scenario, for example, choosing a model that is able to achieve at least a certain fairness level. Overall, at a given bias level, our method doesn't make use of any in-domain protected labels, and consistently outperforms the random baseline.

3.5 Cross-domain: POS tagging

As second cross-domain task, we follow Li et al. (2018) in performing POS tagging.

Data We use three datasets for different purposes: main task training, adversarial training, and out-of-domain evaluation.

Following Li et al. (2018), we train a biLSTM POS tagging model on the English Web Treebank (Bies et al., 2012), comprising 13.5k POS-tagged sentences without protected labels. To evaluate model performance and fairness, we use the TrustPilot English POS-tagged dataset (Hovy and Søgaard, 2015), consisting of 600 sentences with both POS labels and binary author-age labels (over-45-year-old and under-35-year-old).

To train the discriminator, we use unlabelled TrustPilot data (Hovy, 2015), which consists of 156.5k English reviews with author-age labels. Based on protected-label predictability estimation (Algorithm 1), we examine 4 subsampling strategies: (1) "random", based on random-sampling; (2) "largest leakage", select instances with the highest predictability (intuitively the most biased instances); (3) "smallest leakage", select instances where the predictability is below a majority-class baseline; and (4) "absolute leakage", a combination of largest and smallest leakage where equal number of instances from the largest leakage sampling and the smallest leakage sampling are concatenated together.

Results We follow Li et al. (2018), in evaluating fairness via the difference in tagging accuracy between age groups.

Figure 4 shows the trade-off plot with respect to the 4 filtering strategies. Note that the test set only includes 600 instances, so we explore a wider range of the λ_{adv}, and train 5 random initialized

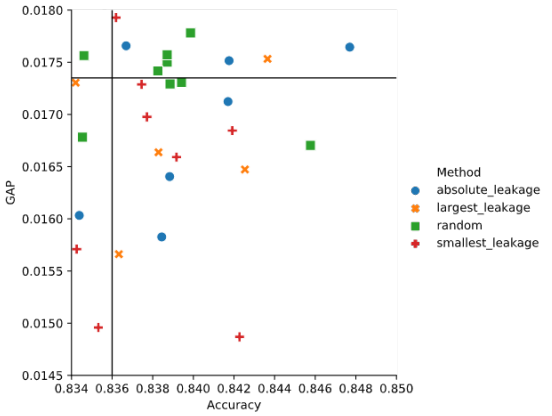


Figure 4: Cross-domain POS Tagging. The vertical line denotes the biased model accuracy, and the horizontal line denotes the biased model Accuracy GAP. Points in the upper-right quadrant are preferable.

	Top P		Top F	
	Acc↑	GAP↓	ACC↑	GAP↓
Biased	83.60	1.74	83.60	1.74
Random	83.94	1.73	83.89	1.70
Largest	83.86	1.66	83.63	1.63
Smallest	83.92	1.68	83.53	1.57
Absolute	84.18	1.75	83.84	1.64

Table 1: Evaluation results on the test set, median value over 5 best models. Biased stands for the non-debiasing model. **Top P** = 5 models with best performance, and **Top F** = 5 models with best fairness. "↑" and "↓" indicate that higher and lower performance, resp., is better for the given metric.

models for each λ_{adv} and take the average. Points in the lower-right quadrant are preferable, in that they decrease bias while increasing accuracy.

We report models of each predictability estimation based sampling method, and all models from random sampling (with respect to different λ_{adv}). Compared to the biased model performance and fairness (vertical and horizontal lines, respectively), the random sampling method does not lead to clear improvements, while our proposed methods lead to consistent gains.

We further compare these methods numerically in Table 1 by selecting top 5 best models from what has been shown in Figure 4. Specifically, we select models with top 5 performance (largest accuracy score) or fairness (smallest GAP) separately, and report the median values of accuracy and GAP over the selected models.

Largest and smallest leakage show close results and are safer choices that consistently outperform random and non-debiasing methods. Intuitively, in a binary classification problem, instances within the the smallest leakage group could also be informative as they could be transformed to largest leakage groups by reverting the predictions, i.e., $\hat{g}_{O,j} = 1 - \mathcal{M}(h_{O,j})$, thus using largest leakage sampling is similar to using smallest leakage sampling. Combining largest and smallest leakage instances together, the absolute sampling method achieves slightly better accuracy performance than other sampling strategies and consist better performance-fairness trade-off than the biased model.

4 Conclusion

We propose a novel training strategy for adversarial debiasing which decouples the training of the main task model and discriminator, including the possibility of training on different data. Based on in-domain evaluation over sentiment analysis and hate speech detection, our method performs as well as the standard adversarial method using only 10% of protected labels. Furthermore, experiments in a cross-domain setting demonstrate the potential feasibility of the method in settings where protected labels are not available.

Acknowledgments

We thank Lea Frermann, Shivashankar Subramanian, and the anonymous reviewers for their helpful feedback and suggestions.

Ethical Considerations

This work aims to use less protected-labelled data in adversarial debiasing. Adversarial training in NLP can help to learn fairer models across demographic attributes used. However, due to the lack of protected labels, previous work has had to use identified demographic attributes for adversarial debasing. By reducing the number of protected labels that are needed in adversarial training, using only self-identified characteristics becomes an effective choice.

References

Ann Bies, Justin Mott, Colin Warner, and Seth Kulick. 2012. English Web Treebank LDC2012T13. Linguistic Data Consortium.

Su Lin Blodgett, Lisa Green, and Brendan O'Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1119–1130, Austin, Texas. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

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. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

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

Dirk Hovy. 2015. Demographic factors improve classification performance. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 752–762, Beijing, China. Association for Computational Linguistics.

Dirk Hovy and Anders Søgaard. 2015. Tagging performance correlates with author age. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 483–488, Beijing, China. Association for Computational Linguistics.

Xiaolei Huang, Linzi Xing, Franck Dernoncourt, and Michael Paul. 2020. Multilingual twitter corpus and baselines for evaluating demographic bias in hate speech recognition. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1440–1448.

Svetlana Kiritchenko and Saif Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 43–53, New Orleans, Louisiana. Association for Computational Linguistics.

Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew Peters, Ashish Sabharwal, and Yejin Choi. 2020. Adversarial filters of dataset biases. In *International Conference on Machine Learning*, pages 1078–1088. PMLR.

Yitong Li, Timothy Baldwin, and Trevor Cohn. 2018. Towards robust and privacy-preserving text representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 25–30, Melbourne, Australia. Association for Computational Linguistics.

Christina Wadsworth, Francesca Vera, and Chris Piech. 2018. Achieving fairness through adversarial learning: an application to recidivism prediction. *FAT/ML Workshop*.

Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. 2019. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5310–5319.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340.

A In-domain Sentiment Analysis Full Plot

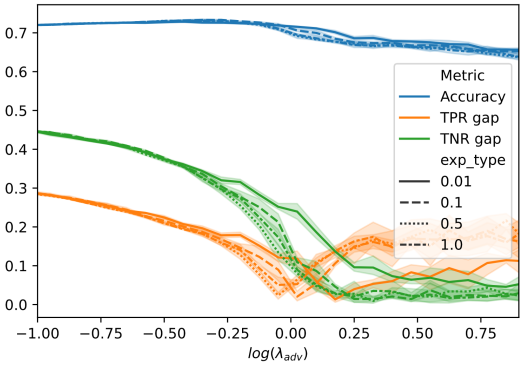


Figure 5: In-domain Sentiment Analysis: main task accuracy and GAP with respect to the trade-off hyperparameter λ_{adv}; shaded areas = 95% CI estimated over 5 runs.

B Computing Infrastructure Used

- CPU: Intel(R) Core(TM) i9-9900K CPU
- GPU: NVIDIA GeForce RTX 2080 Ti
- RAM: 32 GB

C Sentiment Analysis

Models All models are trained and evaluated on the same training/test split. The Adam optimizer is used with learning rates of 3×10^{-5} for the main model and 3×10^{-6} for the sub-discriminators. The minibatch size is set to 1024. Sentence representations (2304d) are extracted from the DeepMoji encoder. The hidden size of each dense layer is 300 in the main model, and 256 in the sub-discriminators. We train M for 60 epochs and each A for 100 epochs, keeping the checkpoint model that performs best on the dev set. Running time: 35 s/epoch

Hyperparameter Range

- λ from 10^{-2} to 10^{+2}. log uniform sampling 30 trails

D Hate Speech

Model Architecture

- hidden size, type=int, 300
- embedding size, type=int, 400
- number of classes, type=int, 2
- adversarial level, type=int, -1 (last hidden layer)
- learning rate, type=float, 0.00003
- number of discriminator, type=int, 1
- adv units, type=int, 256
- batch size, type=int, 512
- epoch, type=int, 100
- dropout, type=float, 0.5
- Running time: 52 s/epoch

Hyperparameter Range

- λ from 10^{0} to 10^{3} based on log uniform sampling over 15 trials

E POS Tagging

Model Architecture

- BATCH SIZE = 64
- LEARNING RATE = 1e-3
- EMBEDDING DIM = 50
- HIDDEN DIM = 100
- N LAYERS = 2
- BIDIRECTIONAL = True
- DROPOUT = 0.25
- EPOCHS = 50
- SEED = 960925
- MIN FREQ = 2
- SAMPLING INDEX = 10
- LAMBDA = 1e-3
- dropout = 0.5
- Running time: 12 s/epoch

Hyperparameter Range

- λ from 10^{-10} to 10^{-8} based on log uniform sampling over 20 trials