Multi-Task Learning for Hate Speech Detection

APPROVED BY

SUPERVISING COMMITTEE:

Matthew Lease, Supervisor

Gregory Durrett

Multi-Task Learning for Hate Speech Detection

by

Sooyong Lee

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2023

Multi-Task Learning for Hate Speech Detection

Sooyong Lee, M.S.Comp.Sci.
The University of Texas at Austin, 2023

Supervisor: Matthew Lease

Amidst the proliferation of social media and the accompanying explosion of information and content generation, the amount of online hate speech has grown rapidly. In efforts to build and train hate speech detection models to counter this, datasets have been annotated for hate speech. However, there exists incompatibility of categories of hate speech across different datasets, the lack of clear and ubiquitous definitions for hate speech, and generalization issues of models which depend highly on training data. To address this, we propose framing hate speech detection as multi-task learning (MTL) which provides a natural and principled way for a model to specialize on dataset-specific hate speech detection tasks while leveraging shared notions of hate speech across datasets to acquire more general notions of hate speech.

Table of Contents

Chapter 1

Introduction

In order to build natural language processing models equipped for hate speech detection, we need datasets annotated for hate speech. However, hate speech datasets are annotated in slightly different ways, as people have differing notions of what hate speech entails and different understandings of terminology in the hate speech domain [17]. Additionally, hate speech datasets are often generated from different domains and distributions. As a result, it can be difficult to incorporate multiple datasets into model training since different datasets may have different output label spaces and come from different contexts. While prior work [8] has shown that models trained on specific datasets suffer from limited generalizability, we believe that there is still valuable information for a hate speech detection model to garner by intelligently training on a diversity of datasets via an MTL architecture to acquire a more general understanding of hate speech.

We evaluate the MTL hate speech detection model on two widely-used hate speech datasets from Davidson et al. and Qian et al. [6, 23]. While other datasets used in prior work such as HateCheck [24] or CONAN [7] are annotated for a binary classification problem of hate versus non-hate, Davidson

[6] is annotated for a more granular multi-class task, with 24783 Twitter posts annotated for three categories of Hate, Offensive, and Normal. The dataset introduced in Qian et al. [23] is annotated for Hate Speech vs. Non-Hate Speech and is composed of posts collected from Reddit.

Some prior MTL work from Liu et al. [18] and Vasileva et al. [27] operates under the assumption that each training point is labeled for all tasks. However, this is not the case in our problem setting, as we use distinct hate speech datasets as input to the model. We use a Conditional MTL approach to ensure that the data points from each dataset influence only the shared layer and the task-specific layer associated with the dataset, which is a key distinction from the traditional MTL schema, which generally operates using a single dataset with densely labeled data points for the different tasks.

The problem of incorporating multiple hate speech datasets for training a hate speech detection model is unclear when the datasets do not share the same label space (such as a binary hate versus non-hate setting). Thus, we deliberately choose datasets with differing output label distributions in order to demonstrate MTL's ability to perform dataset-specific hate speech detection tasks while leveraging shared notions of hate speech across datasets. Additionally, we choose two datasets from two different domains (Twitter versus Reddit) to see whether a model can leverage commonalities of hate speech present across different social media platforms. In our experiments, we observe that the MTL model results in notable gains when evaluated on one of the datasets, but experiences degradation for the other dataset during evaluation.

Chapter 2

Related Work

Multi-task learning has been widely used in machine learning for both computer vision and natural language processing tasks. MTL architectures are generally categorized as either hard or soft parameter sharing of hidden layers. Hard parameter sharing shares the hidden layers between all tasks while keeping several task-specific output layers. Hard parameter sharing is widely used [15, 20, 12] and has been empirically shown to reduce the risk of overfitting, as the model attempts to learn more tasks simultaneously and a shared representation which captures all of the tasks. On the other hand, soft parameter sharing maintains independent task-specific layers with either some form of regularization loss [28] or with correlation units across tasks [21]. Soft parameter sharing models are able to learn an optimal shared and task-specific representation in [10].

2.1 MTL for Hate Speech Detection

Liu et al. use a fuzzy MTL setup for hate speech detection to identify hate speech from single-labeled data [18]. For hate speech, typically the hate class is the smaller class with fewer examples, so Liu et al. mark all unlabelled

examples in their dataset as the larger class, non-hate. This assumption leads to a risk of label contamination, addressed by the conditional MTL architecture described by Gupta et al. [11]. They apply MTL in a demographic-specific toxicity detection setting, but present results on only one dataset in a binary classification setting.

Kovatchev et al. [16] use the dataset from Davidson et al. [6] but frame the problem as a binary classification task by combining hate speech and offensive language into one category, following prior work by Park et al. [22] which similarly binarizes the dataset from Founta et al. [9] by collapsing the 'Abusive'/'Hateful' and 'None'/'Spam' categories together. However, Davidson et al. [6] note that prior work still tend to conflate hate speech and offensive language, and as a result, they label tweets into the three categories of hate speech, offensive language, and neither. Drawing from these claims, we elect to maintain the multi-class labels relevant to a more nuanced hate speech detection task when using the datasets introduced by Davidson et al. [6].

Chapter 3

Problem Statement and Experimental Setup

We are given two datasets $\mathcal{D}_1, \mathcal{D}_2$ with N_1, N_2 samples (posts) and F features. These F dimensional features can be extracted using any off-the-shelf NLP model. We observe multi-class, disparate label sets for the datasets $\mathcal{D}_1$ and $\mathcal{D}_2$.

Our objective is minimizing Ordinal Categorical Cross Entropy (OCE) for $\mathcal{D}_1$ and Binary Cross Entropy (BCE) for $\mathcal{D}_2$ which we do over each dataset in their respective task-specific branches. For each post in $\mathcal{D}_1$, the corresponding label can be either Hate, Offensive, or Normal, and for each post in $\mathcal{D}_2$ the corresponding label can be either Hate Speech or Non-Hate Speech. We use DistilBERT for extracting the F dimensional features from the posts in $\mathcal{D}_1, \mathcal{D}_2$ [25].

3.1 Datasets

Davidson We use Davidson et al. [6]'s dataset which consists of 24783 Twitter posts labeled as hate, offensive, and normal. There are 1430 hate tweets, 19190 offensive tweets, and 4163 normal tweets.

Qian We use the Reddit portion of Qian et al. [23]'s dataset. This

dataset consists of 5K conversations retrieved from Reddit from ten toxic subreddits. The conversations extracted from the subreddits are broken down into 22324 comments, each which are labeled as Hate Speech or Non-Hate Speech. 5257 of the comments are labeled as Hate Speech and 17067 comments are labeled as Non-Hate Speech.

3.2 Architecture

For our baseline models, we use a DistilBERT [25] representation layer to extract the F features from posts. This is followed by layers of dense neuron connections with relu activation and added biases, ending in a classification node with softmax activation. For our experiments, we freeze the weights of the DistilBERT representation layer. The only trainable parameters in the models are the dense neuron units that follow the DistilBERT layer until the output branch.

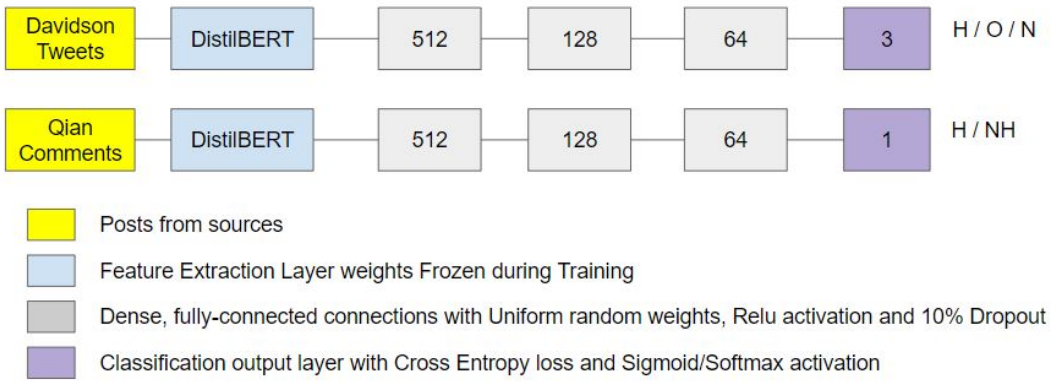

Figure 3.1: Decoupled single task classifiers (STL) for Davidson and Qian

Fig. 3.2 shows the architecture of the MTL model. The MTL model contains a shared layer which consists of 512 dense neurons across both tasks, while the individual task-specific layers for Davidson and Founta have dense connections of 128, 64, and 3 each, following the architecture of the baseline models. The shared layer learns a shared and general representation of hate speech while the task specific layers enable the MTL model to learn representations specific to each dataset. For the Davidson task branch, the task specific layer learns to differentiate the nuances of hate speech with respect to its dataset-specific labels of Normal, Offensive, and Hate, whereas the Qian branch is responsible for the binary task of Hate versus Non-Hate.

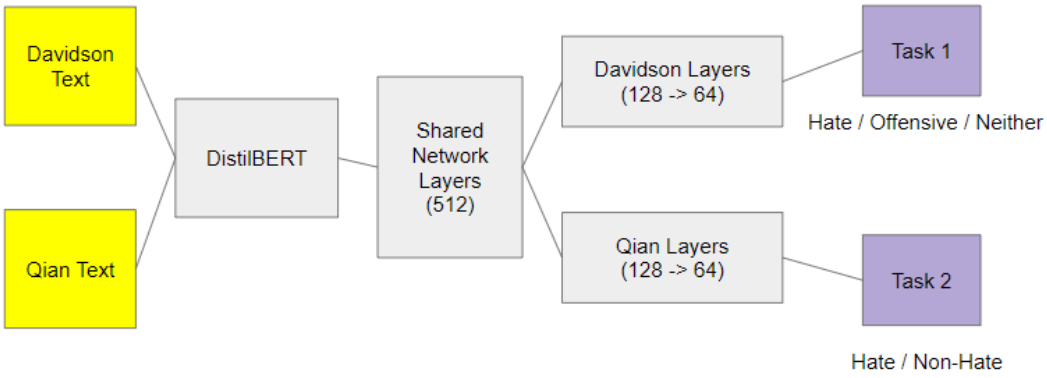

Figure 3.2: MTL architecture

Chapter 4

Results

Experiments use a Nvidia GeForce GTX 1660 SUPER, Intel Core i5-10400F 2.9GHz 6-core CPU and 16GB DDR3 memory. We use the Keras [4] library on a Tensorflow 2.8 backend with Python 3.7 to train the networks in this paper. We present the performance of the MTL architecture when compared with the single task baseline. For optimization, we use AdaMax [13] with (lr=5e-4). We examine post hoc classification performance and performance disparities and compare confusion matrices for the models. We also present our analysis on the results of the experiments.

4.1 Performance Measures

We show the performance comparison of the MTL and STL models on the two datasets, Davidson and Qian after training for 20 epochs. Drawing from the same reasoning from Gupta et al. [11], we consider Recall, F1, and Precision to more holistically evaluate model performance versus accuracy, and we focus our analysis on these metrics. Because the Davidson and Qian datasets are imbalanced datasets, a model which trivially predicts all test posts as either hate (for Davidson) or non-hate (for Qian) would achieve $\sim$ 77%

accuracy. The STL and MTL models achieve ∼80% accuracy on the Qian classification task and ∼86% accuracy for Davidson.

		Davidson			Qian	
		H	O	N	H	NH
Recall	STL	10.5	96.6	64.4	41.7	93.1
	MTL	15.3	95.4	77.8	41.8	94.9
F1	STL	17.4	91.8	71.4	52.8	89.3
	MTL	24.3	92.8	77.8	50.8	88.3
Precision	STL	50.8	87.4	80.2	71.6	84.3
	MTL	59.2	90.3	77.7	65.0	84.0

Table 4.1: Statistical Comparison between different methods based on internal stats: Recall, F1 and Precision. Davidson is skewed towards Offensive and Normal language (1430 Hate, 19190 Offensive, 4163 Normal) while Qian is skewed towards Non-Hate (5257 Hate, 17067 Non-Hate) The MTL model achieves better recall and F1 values for the hate category of Davidson and these results are bolded.

We observe in table 4.1 that the MTL model generally performs better across the board for the Davidson classification task. For the Davidson dataset, the data is primarily skewed towards Offensive language (19190 offensive tweets), with minimal instances of Hate (1430) and Normal language (4163) tweets. In spite of this, it is important that a general hate speech detection model be able to distinguish and identify instances of Hate and Offensive tweets. Thus, the MTL model's ability to better capture both Hate and Offensive tweets is ideal, which is illustrated by the better Recall and F1 scores for the Davidson dataset. When evaluating Davidson, we consider a prediction to be correct only if it exactly matches the true label. Although predicting a tweet to be offensive if the true label is hate is potentially not as bad as if the model had predicted it to be normal, we evaluate such that predicting one of the other two classes is treated as an incorrect prediction.

For Davidson, the recall values for the hate tweets are very low: $\sim$ 11% and $\sim$ 15% for STL and MTL respectively. Similarly, we observe low recall values for Qian: $\sim$ 42% for both STL and MTL. Examining related work in toxicity detection, they observe equally low recall values ($\sim 20 - 30\%$ on the *CivilComments* section of the *Wilds* dataset) [2, 14, 11]. Other prior work also observe low precision and recall values when evaluated on the Founta dataset ($\sim$ 40% recall and $\sim$ 30% precision) for hate speech [9, 5]. While we can achieve better recall values in either the MTL or STL model by tweaking the prediction threshold at the expense of precision, low recall values for hate speech appears to be a common issue for hate speech detection

models. We could also boost recall values by altering the training process and giving significantly more weight to hate training examples, as this would instruct the model to give more emphasis to such training examples; however, this may again be a tradeoff, as it could lead to degradation in performance for the other classes (offensive and normal.)

4.2 Confusion Matrices

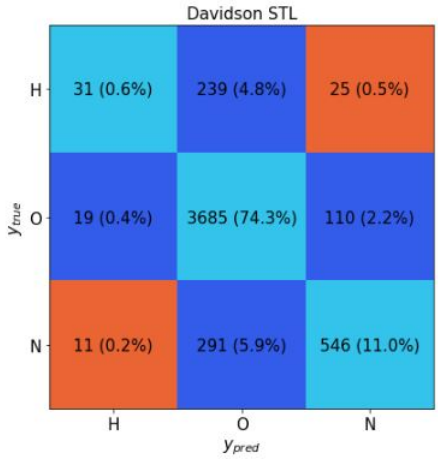

Figure 4.1: Confusion Matrix for STL model on Davidson

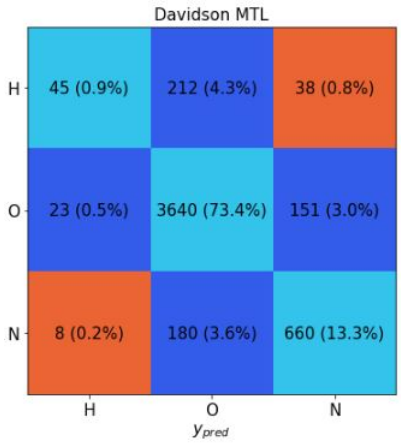

Figure 4.2: Confusion Matrix for MTL model on Davidson

From the confusion matrices in figures 4.1 and 4.2, we can see that generally, the MTL model has learned to better identify the Hate tweets: it captures a greater number of the hate tweets, and makes fewer predictions on hate for tweets that are normal. The MTL model does predict more offensive posts to be hate, but this makes sense given that the model was encouraged to learn this through optimizing for OCE. Additionally, the MTL model is more precise when predicting offensive tweets, as it makes fewer mispredictions, but appears to be less precise when predicting for normal tweets. While it captures a greater number of the normal tweets, it falsely predicts some offensive and hateful tweets as normal. Nevertheless, it does correctly predict for both hateful and offensive content more often than the STL baseline model.

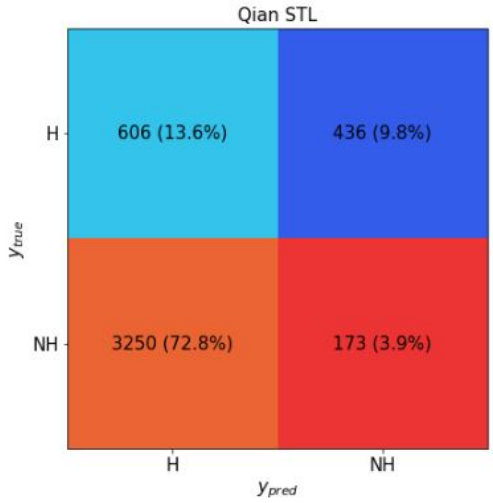

Figure 4.3: Confusion Matrix for STL model on Qian

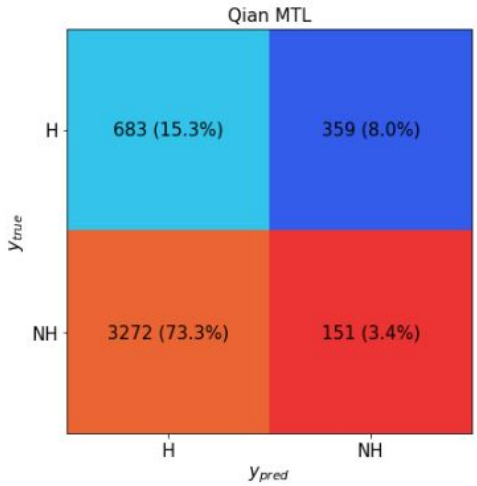

Figure 4.4: Confusion Matrix for MTL model on Qian

For Qian, while the MTL model also captures a greater number of hate comments, this is attributed to the model's propensity for making hate predictions. In the context of hate speech detection, we highly value recall, as we would like to ensure that toxic posts are not mislabeled as non-toxic as such errors could facilitate the spread of hate speech [11]. We suspect that through the shared layer, the MTL model has picked up on patterns of linguistic hate speech from the Davidson dataset during training which has led the Qian branch to make more hate predictions. The MTL model correctly identifies a greater number of hateful tweets (683 vs. 606), while only classifying a few more non-hate tweets as hate (3272 vs. 3250). Additionally, the MTL model makes fewer non-hate predictions for hate tweets (359 for MTL vs. 436 for STL) which is preferable, given that we would like the MTL model to make fewer of these errors.

Chapter 5

Discussion and Future Work

Multi-task learning has been used in many applications of machine learning from natural language processing to computer vision [11, 21]. Prior work has suggested that learning from several related tasks and sharing representations between related tasks can enable the machine learning model to generalize better on each individual task [3]. While this intuitively makes sense, to our knowledge there has not been work in formalizing the benefits of the shared representation layer contingent on the relationship between the tasks. For instance, while Caruna et al. [3] note that using a hard-parameter sharing MTL model yields improved results on their classification tasks, they note that they have yet to determine what mechanism accounts for the observed performance increase. Similarly, our work shows that incorporating data points from another dataset (Qian) leads to increased model performance for the Davidson classification task, but we are unable to determine the specific cause for the observed performance increase.

Drawing from Argyriou et al. [1], future work might develop an interpretable representation of the shared layer of the MTL network. For instance, making assumptions about the similarities across hate-speech detection tasks

(since while different hate-speech detection tasks may vary slightly in their domain or labeling schema, they are still similar tasks) subsequently leads to the conclusion that they should also share a small set of features. Thus, imposing restrictions on the shared layer and efforts towards making the shared layer more interpretable to a practitioner are crucial for future model deployment.

While we apply MTL to the domain of hate speech detection using two datasets, this MTL architecture is easily extensible and can be used to incorporate different datasets and/or more datasets simultaneously. While different datasets typically pose the issue of varying label dimensions, which makes them difficult to incorporate into model training, the MTL architecture enables joint learning across similar tasks. We suspect that adding more datasets to the MTL model should improve its generalizability through the shared layer, but measuring this generalization capability is difficult in the current disparate label setting, because we would lack gold labels when making out of distribution (OOD) predictions on input data through our branches if the gold labels don't match the branch output predictions. Where we consider OOD datasets which contain the same label distribution (such as a binary hate / non-hate dataset), we can evaluate a model by inputting OOD data points and evaluating it on OOD test data and comparing with baseline models.

MTL may not benefit all applications; for instance, applying MTL with two completely unrelated tasks may degrade performance. In the case of this work, we observed that MTL when applied with two similar tasks resulted in performance improvement for one task, but slight degradation for the other.

We also observed overall low recall values for hate speech in both datasets. We emphasize the original motivation of exploring MTL's ability to better capture general notions of hate speech from jointly learning from multiple datasets when compared with decoupled models. Future work could examine when MTL may be beneficial and analyze the effects of joint task learning on the shared layer [19]. Finally, we hypothesize there are still practical applications for MTL approaches in the landscape of large language models. MTL could efficiently train on specialized, confidential data which can be useful when building smaller, distilled models which are more cost-effective when compared with expensive, general-purpose models.

When deploying MTL in practice, a practitioner may use two datasets which are sampled from the same source (e.g. Twitter). However, these datasets may be sampled in different ways which can cause an out-of-distribution issue when applying a branch trained on one dataset to make predictions on test data from another dataset. Furthermore, if the label output for the two datasets are incompatible (for example, one dataset is binary while the other is ternary, or if the datasets have differing output labels), then it can be difficult to evaluate one task-specific branch's prediction for the other dataset due to the absence of a gold label. We resolve this issue in this work by choosing datasets which come from different domains (Twitter and Reddit) so that we can determine given input data which task of the MTL architecture to pass the data through for prediction and evaluation; however, if we trained an MTL architecture on datasets which come from the same domain (e.g. Twitter), then

we would have required additional metadata to determine which task branch would have been valid for the input data to be passed through.

Chapter 6

Conclusion

In this work, we frame hate speech detection as multi-task learning, which enables a model to learn shared notions of hate speech across different hate-speech datasets in order to acquire a better understanding of hate-speech. We note that while hate-speech datasets used in prior literature often differ slightly which makes incorporating them difficult without making strong pre-processing assumptions, multi-task learning offers an intuitive way for jointly learning the slightly varying hate-speech detection tasks while maintaining independent task identities. We observe that framing hate-speech detection as an MTL problem yields mixed results in our work, showing improved performance for one dataset and slight degradation in the other. Future work could explore the potential limitations and applications of MTL in this space.

Bibliography

[1] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. *Advances in neural information processing systems*, 19, 2006.

[2] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion proceedings of the 2019 world wide web conference*, pages 491–500, 2019.

[3] R Caruana. Multitask learning: A knowledge-based source of inductive bias1. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 41–48. Citeseer, 1993.

[4] François Chollet. keras. `https://github.com/fchollet/keras`, 2015.

[5] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. Racial bias in hate speech and abusive language detection datasets. *arXiv preprint arXiv:1905.12516*, 2019.

[6] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515, 2017.

[7] Margherita Fanton, Helena Bonaldi, Serra Sinem Tekiroglu, and Marco Guerini. Human-in-the-loop for data collection: a multi-target counter narrative dataset to fight online hate speech. *arXiv preprint arXiv:2107.08720*, 2021.

[8] Paula Fortuna, Juan Soler, and Leo Wanner. Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In *Proceedings of the 12th language resources and evaluation conference*, pages 6786–6794, 2020.

[9] Antigoni Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Proceedings of the international AAAI conference on web and social media*, volume 12, 2018.

[10] Yuan Gao, Jiayi Ma, Mingbo Zhao, Wei Liu, and Alan L Yuille. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3205–3214, 2019.

[11] Soumyajit Gupta, Sooyong Lee, Maria De-Arteaga, and Matthew Lease. *Same Same, But Different*: Conditional Multi-Task Learning for Demographic-Specific Toxicity Detection. In *Proceedings of the Web Conference*, 2023.

[12] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using

uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021.

[15] Iasonas Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6129–6138, 2017.

[16] Venelin Kovatchev, Soumyajit Gupta, and Matthew Lease. Fairly accurate: Learning optimal accuracy vs. fairness tradeoffs for hate speech detection. *arXiv preprint arXiv:2204.07661*, 2022.

[17] Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. Benchmarking aggression identification in social media. In *Proceedings of the first workshop on trolling, aggression and cyberbullying (TRAC-2018)*, pages 1–11, 2018.

[18] Han Liu, Pete Burnap, Wafa Alorainy, and Matthew L Williams. Fuzzy multi-task learning for hate speech type identification. In *The world wide web conference*, pages 3006–3012, 2019.

[19] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[20] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Philip S Yu. Learning multiple tasks with multilinear relationship networks. *Advances in neural information processing systems*, 30, 2017.

[21] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3994–4003, 2016.

[22] Ji Ho Park, Jamin Shin, and Pascale Fung. Reducing gender bias in abusive language detection. *arXiv preprint arXiv:1808.07231*, 2018.

[23] Jing Qian, Anna Bethke, Yinyin Liu, Elizabeth Belding, and William Yang Wang. A benchmark dataset for learning to intervene in online hate speech. *arXiv preprint arXiv:1909.04251*, 2019.

[24] Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, Janet Pierrehumbert, et al. Hatecheck: Functional tests for hate

speech detection models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 41–58. Association for Computational Linguistics, 2021.

[25] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[26] Steve Durairaj Swamy, Anupam Jamatia, and Björn Gambäck. Studying generalisability across abusive language detection datasets. In *Proceedings of the 23rd conference on computational natural language learning (CoNLL)*, pages 940–950, 2019.

[27] Slavena Vasileva, Pepa Atanasova, Lluís Màrquez, Alberto Barrón-Cedeño, and Preslav Nakov. It takes nine to smell a rat: Neural multi-task learning for check-worthiness prediction. *arXiv preprint arXiv:1908.07912*, 2019.

[28] Yongxin Yang and Timothy Hospedales. Deep multi-task representation learning: A tensor factorisation approach. In *5th International Conference on Learning Representations*, 2017.