2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA) | 978-1-7281-8470-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICMLA51294.2020.00165

DeL-haTE: A Deep Learning Tunable Ensemble for Hate Speech Detection

Joshua Melton*, Arunkumar Bagavathi†, Siddharth Krishnan*
Department of Computer Science
University of North Carolina at Charlotte, Oklahoma State University†*
jmelto30@uncc.edu, abagava@okstate.edu, skrishnan@uncc.edu

***Abstract*—Online hate speech on social media has become a fast-growing problem in recent times. Nefarious groups have developed large content delivery networks across several mainstream (Twitter and Facebook) and fringe outlets (Gab, 4chan, 8chan, etc.) to deliver cascades of hate messages directed both at individuals and communities. Thus addressing these issues has become a top priority for large-scale social media outlets. Three key challenges in automated detection and classification of hateful content are the lack of clearly labeled data, evolving vocabulary and lexicon - hashtags, emojis, etc - and the lack of baseline models for fringe outlets such as Gab. In this work, we propose a novel framework with three major contributions. (a) We engineer an ensemble of deep learning models that combines the strengths of state-of-the-art approaches, (b) we incorporate a tuning factor into this framework that leverages transfer learning to conduct automated hate speech classification on unlabeled datasets, like Gab, and (c) we develop a weak supervised learning methodology that allows our framework to train on unlabeled data. Our ensemble models achieve an 83% hate recall on the HON dataset, surpassing the performance of the state of the art deep models. We demonstrate that weak supervised training in combination with classifier tuning significantly increases model performance on unlabeled data from Gab, achieving a hate recall of 67%.**

***Index Terms*—ensemble classifier, transfer learning, weak supervision, hate speech detection**

I. Introduction

The proliferation of online hate speech has become prevalent in recent times. Numerous social media outlets and the computational social science community are looking at various automated techniques to detect and classify hate speech. However, most models, nascent in nature, have significant limitations due to the complexity of the problem. Primarily, the lack of a reliable baseline coupled with an evolving vocabulary of hateful content makes this a particularly challenging issue. For instance, many studies have classified this problem as a binary classification task [1], [2], but this fails to address the subtleties of hate speech, such as direct (use of vulgar language against an individual or community) vs. indirect (the content can be lexicographically clean but implies negative content) hate speech. These binary classification models also fail to identify different types of hate speech like racism, sexism, antisemitism, etc. or their varying degrees. Another key obstacle that plagues these binary models is their inability to distinguish between general offensive language and hate speech [3]. A third issue that arises in designing automated approaches is class imbalance—hate speech is usually a small percentage of the overall data—and the need to adequately upsample hate observations without model overfitting.

In our work, inspired by the recent successes in developing multi-class hate speech models that separate hate speech from offensive content [3], [4], we propose *DeL-haTE*, an ensemble of tunable deep learning models that leverages CNN and GRU layers. The CNN layer extracts higher-order features from the word embedding matrix that then inform the GRU layer, which extracts informative features from the sequence of words. These features are utilized for automatic detection of hate speech on social media. Our novelty lies in using a tuning procedure to adapt the model to individual dataset characteristics.

Our major contributions can be summarized by answering the following questions.

1) **How do you leverage existing state of the art classification models for automatic hate speech detection?** We utilize existing deep model topologies to develop an ensemble classifier model for hate speech detection. An ensemble approach effectively tackles issues of class imbalance and model variability that are significant problems for automatic detection of hate speech.
2) **How can you engineer a generalized framework for hate speech detection on unlabeled data?**
    a) **How can you extend trained classification models to evolving, unlabeled data?** We extend pretrained models by applying transfer learning to tune the classifiers to new target datasets. The tunability of our framework allows the model to adapt to new and ever-changing data.
    b) **How can you develop an unsupervised framework for unlabeled data?** We develop a weak supervision methodology that allows our framework to train and tune entirely on unlabeled data, further extending the applicability of our model to new data.

Summary of Results: Our best ensemble on the HON dataset achieves a 65% F1 Macro and an 83% hate recall, surpassing the performance on the HON dataset of current state of the art models by 33%. We show that the ensemble models outperform individual models by an average of 5% hate recall and 8% F1 macro across all datasets. When applied to unlabeled Gab data, tuning improved the pretrained models

978-1-7281-8470-8/20/$31.00 ©2020 IEEE
DOI 10.1109/ICMLA51294.2020.00165

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

by an average of 12%, with the best tuned ensemble models achieving 57% hate recall. Our model trained using weak supervision achieved a 67% hate recall on posts from Gab.

II. Related Work

Developing a consistent definition of *hate speech* is difficult due to its controversial and subjective nature [5], [6]. Social media sites define hate speech in legal terms as a "*direct attack*" or "*promoting violence*" against various characteristics of people, including race, ethnicity, nationality, religion, gender, and others. Many previous analyses have approached the study of hate speech analysis through the lenses of these characteristics. Examples include automatic hate speech detection modeled as - a binary classification problem [1], [2], an attention-based multi-task learning model to identify toxic comments [7], a quantification of conflicting opinion among communities [8], and a racism/sexism classifier with embeddings learned from multiple deep learning architectures [9]. A binary classification based approach, although simple, ignores the many subtleties of hate speech, such as indirect vs. direct hate speech and different forms of hate speech, such as racism, sexism, or antisemitism. Furthermore, the high prevalence of offensive language on social media presents an additional challenge for automatic hate speech detection online [3].

Several multi-class classification models [1], [3], [4], [10] have been introduced recently to better distinguish hate speech from offensive content on social media and to improve the automatic identification of various types of hate speech. In a similar vein, hierarchical annotation systems have been proposed that further distinguish the type and target of offensive posts, providing additional granularity for automatic detection models [11]. One of the major issues with automatic hate speech detection research is the limited amount of manually labeled data. Weak supervised training allows for the use of large unannotated datasets by programmatically generating "weak" labels using heuristic approaches [12], [13]. Weak supervised learning has been applied for problems like cyberbullying detection to give better performance than traditional approaches [14].

Despite the strong performance of recent automatic hate speech detection models, with high reported recall and F1 scores (above 90%), efforts to replicate reported findings and to generalize models to other similar datasets have often failed [10], [15]. While some of these failings are due to methodological shortcomings such as overfitting, they are also related to the inherent subjectivity of hate speech, the noisiness of short-text social media posts, and the biases present in datasets [10]. With binary classification, it is also difficult to report the extent to which hate speech detection models are conflating hate speech with general offensive speech online [3]. In addition to these challenges, hate speech constitutes a small portion of the overall content on social media [5], [16], leading to the presence of severe class imbalance in hate speech datasets. Example include the datasets we use in this work as given in Table I. Despite recent efforts to identify and address these challenges [1], [4], [10], there remains room for improvement in developing robust and generalized frameworks for automatic detection of hate speech on social media.

In this work, we develop a number of multi-class classifiers comparing four sets of pretrained word embeddings and three different deep model architectures. We leverage an ensemble approach to address issues of class imbalance and the limited number of per-class observations during model training. In order to assess the generalizability of our model frameworks, we apply transfer learning to tune classifiers trained on existing labeled Twitter datasets and test using a small sample of manually labeled posts from Gab.ai.

III. Datasets

We use two Twitter datasets for our experiments, which are referred to as HON [3] and OLID [11] throughout this paper. We also use unlabeled posts from the social media forum called *gab.com* or *gab.ai*. *gab.com* started gaining traction immediately after the 2016 U.S. Presidential election due to its support for *free speech* in online media. Much previous research has showcased evidence of the spread of antisemitic and racist ideologies in this forum [5], [16]. We utilize this data to examine the generalizability of our developed framework, and the dataset is referred to as Gab throughout this paper. For the HON dataset, Davidson et al. [3] classified tweets into three categories: hate speech, offensive language, and neither. In order to standardize our experimental setup, we apply this data representation to all the above mentioned datasets in our analysis.

The HON dataset[1] consists of approximately 25,000 labeled tweets. These tweets were originally sampled using the hate speech lexicon from *Hatebase.org*[2], which identifies common words and phrases that are marked as hate speech by online users. This data corpus is a random sample the complete sets of tweets from selected users (8.54 million). This sample was then manually labeled using *CrowdFlower* to produce the final labeled dataset [3].

The OLID dataset[3] consists of about 13,000 labeled tweets. This data corpus was collected from Twitter using a set of keywords and constructions that are often included in offensive tweets, with main emphasis on political keywords that are more likely to result in offensive content. These tweets are then manually labeled using *Figure Eight* as *offensive* and *not offensive*. Each tweet in this dataset is annotated with a three level hierarchical scheme: denoting offensive language, the category of offensive language, and the target of offensive language [11]. For our experiments, we adapt the OLID hierarchical labels into the (H)ate, (O)ffensive, (N)either three-class labels by labeling offensive posts that are targeted at a group as **H**ate, remaining offensive posts as **O**ffensive, and not offensive posts as **N**either.

The Gab dataset consists of approximately 1,500 posts that were randomly sampled from a Gab posts database containing over 35 million posts. These posts were manually annotated by

[1]https://github.com/t-davidson/hate-speech-and-offensive-language
[2]https://hatebase.org
[3]https://scholar.harvard.edu/malmasi/olid

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

TABLE I
DISTRIBUTION OF HATEFUL (H), OFFENSIVE (O), AND NEITHER (N) DATA SAMPLES IN MULTIPLE DATASETS. HATEFUL POSTS ARE ONLY 5%-11% OF THE TOTAL CORPUS FOR MACHINE LEARNING TRAINING

Dataset	Class (in %)			# Samples
	H	O	N	
HON	5.74	77.41	16.85	22,305
OLID	8.24	25.09	66.67	11,916
Combined	6.61	59.19	34.20	34,221
Gab - Test	11.19	22.05	66.76	1,465
Gab - Train	–	–	–	90,899

two researchers utilizing the procedure described by [3]. For weak supervised training, a randomly sampled set of nearly 100,000 unlabeled posts was utilized.

IV. METHODOLOGY

The code for this paper is available on Github[4]. We conduct a comparative analysis of three deep model architecture variants in order to determine the optimal deep model architecture and word embedding representation. We compare the following five word embedding methods: Word2Vec vectors trained on Google News corpus [17], GloVe vectors trained on CommonCrawl (GLoVe-CC) and Twitter (GLoVe-Twitter) corpora [18], and FastText vectors trained on CommonCrawl (FastText-CC) and Wikipedia (FastText-Wiki) corpora [19]. The pre-trained distributional embedding vectors are all implemented using PyTorch's torchtext.Vocab library. To examine the issues of class imbalance, limited labeled training data, and model generalizability, we develop an ensemble model approach and compare transfer learning and weak supervised training using data from Gab. For our experiments, we report Macro F1, which weights each class equally regardless of size, and hate class recall. All reported results are averaged across 5 trials with an ensemble size of 5 models.

A. Text Processing

We follow a standardized procedure to preprocess all posts in all datasets that are used in our experiments. In the preprocessing, we remove all extra text elements like URLs and emojis. User mentions are normalized to "MENTIONHERE", and hashtags are normalized to "HASHTAGHERE <hashtag_text>". We then apply tokenization and stemming to the normalized texts using NLTK's word tokenizer and Porter Stemmer, respectively. We retrieve word embeddings for the stemmed tokens with pre-trained models: *Word2vec*, *GloVe*, and *FastText*. We standardize the size of word embedding with zero left padding to convert the word embeddings matrix dimension to 100×300 for *word2vec*, *GloVe-CC*, and both *FastText* models and 100×200 for the *GloVe-Twitter* model.

B. Deep Model Implementation

Our deep learning framework is motivated from the deep model topologies proposed in [1] and [4]. A schematic of our model is given in Figure 1. A 1-dimensional CNN layer takes the 100×300 word embedding matrix as input. We utilize a CNN layer with 32 filters and a filter width of 17 with padding of 8 to match input and output dimensions. We then apply max pooling with an undersampling rate of 4 to generate a 75×32 feature space. With this vector space, we compare between two sub-models using combinations of convolutional (CNN), recurrent (RNN), and fully connected (FC) layers:

- **CNN-RNN-FC**: For the CNN-RNN-FC model variant, the 75×32 feature space is passed to RNN layer with 100 hidden units. We use an LSTM or GRU as the RNN layer in our experiments. The 100×32 output is flattened to a 3,200-dimension vector, and global max pooling is applied to generate a 100-dimension feature vector. The 100-dimension feature vector is passed to an FC layer that utilizes ReLU activation and outputs 25 hidden units. For the hidden FC layer, we use dropout with a probability of 0.2 during training. A final output FC layer utilizes Softmax activation and outputs class probabilities for **H**ate, **O**ffensive, and **N**either.
- **CNN-FC**: For the CNN-FC model variant, the 75×32 output feature space from the CNN layer is flattened to a 2,400-dimension feature vector and passed to the two FC layers.

C. Ensemble Training

We follow ensemble training with five independent versions of our developed CNN-RNN-FC model variants discussed in the previous section. We use a standard data split of 80-10-10 train-valid-test for the ensemble training, tuning, and testing. For each ensemble training epoch, we pass equal class distributions of hate, offensive, and neither. Due to high frequency of *Offensive* and *Neither* classes, we randomly sample *Offensive* and *Neither* class data with replacement. We run 20 epochs of training, utilizing early stopping to save the model weights and biases at the epoch with the minimum validation loss. The prediction from the ensemble classifier is the majority decision from five independent classifiers.

D. Transfer Learning

To test the generalizability of our framework's hate speech prediction, we experiment with transfer learning by training the classifiers on Twitter-based HON, OLID, and Combined datasets and evaluating on Gab posts. As reported in Section V-C, we find that the models tested on completely unseen data samples give low performance. Thus we tune classifiers trained on our labeled experiment datasets with a small, manually labeled set of posts from Gab. A set of 150 posts (50 posts for each class Hate, Offensive, and Neither) is selected as a balanced training set for transfer learning. For the tuning, we freeze the weight and bias parameters of feature extraction CNN-RNN layers, while the classifier component with the two FC layers, is allowed to train on the Gab data. The exceptionally small size of the training data makes the models highly sensitive to overfitting on the Gab observations. For

[4]https://github.com/NASCL/DeL-haTE

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

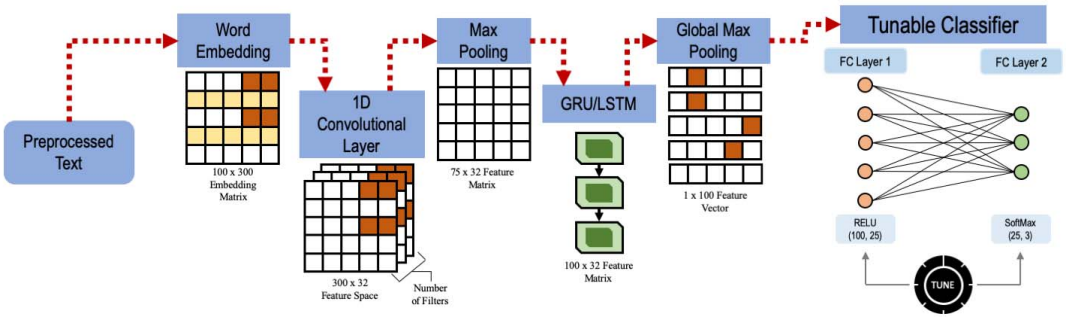

Fig. 1. Overview of the entire pipeline process for a single post. The model topology depicted illustrates the CNN/GRU feature component and the tunable classifier. These models are combined into an ensemble model for class prediction.

tuning the models' classifier components, the training process runs for 10 epochs at a reduced learning rate of $5e{-}4$.

E. Weak Supervised Training

Utilizing weak supervision during model training is a possible alternative to reliance on manually annotated hate speech datasets. The major benefit of weak supervision is the greatly expanded pool of training observations by leveraging abundant unlabeled social media data. For our classifier models, the framework utilizes a weakly supervised form of cross-entropy as the loss function f. This loss function is the multi-class extension of the weak supervised loss function utilized by [14] for binary classification of cyberbullying. Our weak supervised loss relies on lexicons of words indicative and counter-indicative of hate speech and offensive language. For a post containing n unique words, let n_h, n_o, and n_p denote the number of hate words, offensive words, and positive words, respectively. For each post, we calculate the bounds of each class using the algorithm described in Figure 2.

If this bound is violated, the loss function penalizes the model using the weak supervised loss function:

$$f(y_m) = \sum_{c \in C} (-log(min\{1, 1 + y_{m,c} - lb_{m,c}\}) \\ -log\{min(1, 1 + ub_{m,c} - y_{m,c}\})) * w_c$$

With the lack of *a priori* class labels, this form of weak supervised training is susceptible to the class imbalance problem due to the small proportion of hate speech in the overall content on social media. To mitigate this issue, a per-class weight w_c is applied to the weak supervised loss contribution for each class. One benefit of our weak supervised heuristic is the tunable per-class weight applied during the loss calculation which can be adjusted for varying class balances in datasets.

V. Results

In this section, we present the findings from our experiments on ensemble training, transfer learning, and weak supervision for automatic hate speech detection. With our experiments we show that (i) our ensemble approach outperforms the state of the art deep models on the HON dataset, (ii) our transfer learning with classifier tuning improves performance of pre-trained models on a novel dataset, and (iii) our weak supervision methodology can train our framework on entirely unlabeled data.

Algorithm – 1: Bounds
Require: Post "p"
num_unique ← number of unique words in *p*
num_hate ← number of hate words in *p*
num_offensive ← number of offensive words in *p*
num_positive ← number of positive words in *p*

if (hate word in p):
  hate_LB ← (num_hate + num_offensive) / num_unique
  hate_UB ← 1
  offensive_LB ← 0.5 * num_offensive / num_unique
  offensive_UB ← 1 - (num_hate + num_positive) / num_unique
  neither_LB ← 0
else if (offensive word in p):
  hate_LB ← 0
  hate_UB ← 1 - (num_offensive + num_positive) / num_unique
  offensive_LB ← (num_offensive + 0.5 * num_positive) / num_unique
  offensive_UB ← 1 - 0.5 * num_positive / num_unique
  neither_LB ← 0.5 * num_positive / num_unique
else:
  neither_LB ← num_positive / num_unique
neither_UB ← 1 - (num_hate + num_offensive) / num_unique

return ((hate_LB,hate_UB), (offensive_LB,offensive_UB), (neither_LB,neither_UB))

Fig. 2. Weak supervised bounds algorithm that calculates a lower and upper bound for each class.

A. Surpassing Performance on the HON Dataset

We first focus on extending and improving the CNN-RNN-FC model architecture and surpassing the performance of the current best models on the HON dataset (Table II). The CNN-RNN-FC toplogy with a single 1-dimensional convolutional layer and an LSTM/GRU layer has been shown to be an effective model for hate speech classification [1], [4]. We adapt this model to develop an ensemble classifier utilizing both the HON and OLID datasets. Table III summarizes our findings on the Twitter datasets. The model variant trained on HON consisting of word embeddings from a pre-trained FastText model trained on a Wikipedia corpus in combination with a CNN+GRU architecture achieves F1 Macro of 65% and a hate recall of 83%, outperforming the state of the art deep models on the HON dataset by 33% on hate recall.

Due to the smaller size of OLID dataset—with only half the size of the HON dataset—and with imperfect mapping of the original hierarchical labels to the hate, offensive, and neither

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

TABLE II
BASELINE CLASSIFICATION MODELS APPLIED TO THE HON DATASET. REPORTED RESULTS ARE FROM THE ORIGINAL CORRESPONDING STUDIES.

HON - Baseline

Topology	**F1-Micro**	**Hate Recall**
LogRegBase [3]	0.91	0.61
DeepBase [1]	0.94*	n/a*
	F1-Macro	**Hate Recall**
GloVe-CC+CNN+LSTM+BestAug [4]	0.741	0.496

* Binary classification task - hate recall was not reported.

labels, Table III illustrates that performance on the OLID dataset is lower than the HON dataset. Our primary motivation in utilizing the OLID dataset is to augment the available training data and to improve the model's generalizability to new datasets.

Integrating the HON and OLID datasets into a single training set increases both the size (a nearly 50% increase from the HON dataset alone) and the diversity of training examples for the model to learn from during training (Table I). The increased training pool also improves the downsampling procedure utilized to mitigate the class imbalance of hate speech datasets. The size of the train set passed to the model at each epoch is limited by the size of the minority class. The Combined dataset contains an additional 1,000 observations in the hate class. Table III presents our findings for the ensemble models trained on the Combined HON+OLID dataset. We find that the F1-Macro of these models is able to equal the performance of the models trained on HON alone.

B. Benefits of the Ensemble Approach

A major factor contributing to the variability in hate speech detection model performance is the severe class imbalance present in hate speech datasets [10], [15]. Similar to [4], we likewise found that using downsampling during training is critical in preventing the model from almost exclusively predicting the majority class. One drawback of downsampling during training is overfitting on the smaller distribution of hate class examples during training. In addition, random sampling of observations in the remaining classes can lead to variability in the resultant models. We therefore employ a simple ensemble approach [20] to counteract the variability in individual classifier models for hate speech detection.

Our experiments demonstrate that utilizing an ensemble approach for hate speech classification significantly improves model performance when compared against individual classifier models. In Table IV, we summarize our findings regarding the benefits of an ensemble approach over individual models. The individual results are averaged over the 25 component models, and the ensemble results are averaged over five trials with an ensemble size of five. On average, the ensemble models outperform individual models by 5% in hate recall and 8% in F1 Macro.

In our experiments, we found that the GRU variant of the CNN-RNN topology generally outperformed the other model variants. For the HON dataset, GRU models are on average 17% better on hate recall and are equal in average F1 Macro when compared with the other variants across all embedding choices. A similar trend is seen for the OLID and Combined datasets, where the GRU variants also outperformed the other variants on F1 Macro. Our experiments indicate that the GRU model variant may be better suited to extracting informative patterns from sequential features for hate speech classification tasks. Because of the improved performance, training times, and computational resource limitations, we elected to conduct all experiments with weak supervision using the CNN-GRU-FC model variant.

C. Extension to Gab Data: Transfer Learning

The use of transfer learning for NLP tasks, and particularly for automatic hate speech detection, remains an open research question [21], [22]. Our experiments on the HON and OLID datasets demonstrate the success of our ensemble CNN-GRU-FC model architecture on curated, labeled datasets. In order to examine how well our model can be extended to novel, unlabeled data from a non-Twitter social media source, we conduct a series of experiments applying transfer learning tuning using the Gab dataset. We utilize the labeled HON and OLID data as source datasets to train classifier ensembles and then tune these pretrained models using data from Gab as the target dataset.

Table V summarizes our findings when applying models trained on the HON and OLID datasets to Gab data. The overall reduction in performance is in line with similar decreases in performance found by [10] when applying hate speech classification models to novel datasets. The pre-trained models are evaluated on the manually labeled Gab test set prior to and after tuning, demonstrating an average 12% improvement in hate recall across all models on the Gab test set after tuning. The best performing model is trained on the Combined HON+OLID dataset and after tuning, achieved a hate recall of 57%. These results indicate that tuning of classifier models can lead to significantly improved model performance when applying automatic hate speech detection models to novel datasets.

As we hypothesized, the models trained on the Combined dataset generalize better to the novel Gab dataset. After tuning, ensembles trained on the Combined dataset outperform models trained on the HON dataset alone by an average of 10% on hate recall and 5% on F1 Macro. On average, the ensembles trained on the Combined dataset and then tuned to Gab data achieved a hate recall equivalent to that of the state of the art models on the HON dataset (Table II). Our experiments demonstrate the generalizability of our ensemble framework and indicate that tuning classifier models trained on well-annotated datasets improves the performance of models when extended to novel data.

D. Extension to Gab Data: Weak Supervision

Manually labeled datasets, such as HON and OLID, are a valuable asset for research on hate speech on social media,

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

TABLE III
COMPARISON OF THE NETWORK TOPOLOGIES ON THE *HON*, *OLID*, AND *Combined* DATASETS. EACH BLOCK CORRESPONDS TO A DIFFERENT WORD EMBEDDING AND REPORTS THE RESULTS FOR THE THREE DEEP MODEL VARIANTS.

	HON Ensemble		**OLID Ensemble**		**Combined Ensemble**	
Topology	**F1-Macro**	**Hate Recall**	**F1-Macro**	**Hate Recall**	**F1-Macro**	**Hate Recall**
Word2Vec+CNN+FC	0.63	0.47	0.48	0.16	0.63	0.33
Word2Vec+CNN+GRU	0.57	0.59	0.43	0.14	0.61	0.45
Word2Vec+CNN+LSTM	0.60	0.50	0.44	0.16	0.61	0.44
GloVe-Twitter+CNN+FC	0.66	0.53	**0.35**	**0.44**	0.64	0.46
GloVe-Twitter+CNN+GRU	0.65	0.78	0.47	0.33	0.66	0.65
GloVe-Twitter+CNN+LSTM	0.64	0.53	0.42	0.19	0.63	0.46
GloVe-CC+CNN+FC	0.68	0.44	0.48	0.38	0.65	0.55
GloVe-CC+CNN+GRU	0.65	0.48	0.54	0.41	**0.65**	**0.66**
GloVe-CC+CNN+LSTM	0.67	0.48	0.48	0.23	0.66	0.46
FastText-CC+CNN+FC	0.70	0.53	0.47	0.31	0.67	0.50
FastText-CC+CNN+GRU	0.68	0.74	0.51	0.36	0.70	0.62
FastText-CC+CNN+LSTM	0.68	0.55	0.42	0.20	0.65	0.51
FastText-Wiki+CNN+FC	0.68	0.54	0.47	0.36	0.64	0.54
***FastText-Wiki*+CNN+GRU**	**0.65**	**0.83**	0.56	0.36	0.67	0.65
FastText-Wiki+CNN+LSTM	0.65	0.52	0.43	0.24	0.65	0.46

TABLE IV
COMPARISON OF THE BEST-PERFORMING ENSEMBLE MODELS VERSUS THEIR INDIVIDUAL COMPONENT MODELS. ENSEMBLE MODELS OUTPERFORM THEIR COMPONENT MODELS BY AN AVERAGE OF 8% IN F1 MACRO AND 5% IN HATE RECALL.

Individual vs. Ensemble Models					
		Individual		**Ensemble**	
Dataset	**Topology**	**F1-Macro**	**Hate Recall**	**F1-Macro**	**Hate Recall**
HON	*GloVe-Twitter*+CNN+GRU	0.55	0.69	0.65	0.78
	GloVe-CC+CNN+LSTM	0.62	0.53	0.67	0.48
	FastText-CC+CNN+GRU	0.54	0.61	0.68	0.74
	***FastText-Wiki*+CNN+GRU**	**0.52**	**0.72**	**0.65**	**0.83**
OLID	***GloVe-Twitter*+CNN+FC**	0.31	0.42	0.35	0.44
	GloVe-CC+CNN+GRU	0.44	0.41	0.54	0.41
	FastText-CC+CNN+GRU	0.42	0.34	0.51	0.36
	FastText-Wiki+CNN+GRU	0.46	0.35	0.56	0.36
Combined	*GloVe-Twitter*+CNN+GRU	0.60	0.59	0.66	0.65
	***GloVe-CC*+CNN+GRU**	0.58	0.63	0.65	0.66
	FastText-CC+CNN+GRU	0.62	0.58	0.70	0.62
	FastText-Wiki+CNN+GRU	0.60	0.59	0.67	0.65

but these datasets constitute only a minute portion of the total content on social media. There remains a need for robust metholodogies that can successfully utilize large amounts of unannotated social media data. Besides transfer learning, weak supervision methods that create a set of "weak" labels using heuristics allow for the use of unannotated data for training machine learning models [12], [13]. In our experiments, we develop a heuristic method for the Hate, Offensive, and Neither scheme that generates a per-class bound for each class. We test our procedure on the HON and OLID datasets, as well as on a large unannotated corpus of posts from Gab.

Table VI summarizes our finding regarding weak supervised training using the HON and OLID datasets. The weak supervised models do not equal the performance of the models trained using the dataset labels. But, when tuned and tested on the novel Gab data, the weak supervised ensembles are able to equal or surpass the performance of the standard ensembles. Our experiments demonstrate a marked improvement in hate recall, 18% on average, and in F1 Macro, 3% on average. Table VII summarizes our findings for the weak supervised ensembles when evaluated on the Gab dataset. Our experiments demonstrate that while weakly supervised models fail to equal the performance of standardly trained models on labeled datasets, these ensemble models can equal and exceed their performance on novel data, such as the data from Gab.

We also experiment with training a set of ensembles using our weak supervision heuristic on posts from Gab itself. For social media where there are no manually labeled datasets available, such as Gab, our experiments demonstrate that weak supervision provides a viable alternative for model

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

TABLE V
COMPARISON OF PRE-TRAINED ENSEMBLES EVALUATED ON THE *Gab* DATASET. RESULTS ARE REPORTED BEFORE AND AFTER TUNING IS APPLIED TO THE MODELS. TUNED ENSEMBLES SHOWED AN AVERAGE IMPROVEMENT OF 12% ON HATE RECALL.

Pretrained Ensembles - Gab

		Pre-tuning		Post-tuning	
Train Data	**Topology**	**F1-Macro**	**Hate Recall**	**F1-Macro**	**Hate Recall**
HON	***GloVe-Twitter*+CNN+GRU**	0.38	0.24	0.33	0.50
	GloVe-CC+CNN+LSTM	0.42	0.17	0.42	0.34
	FastText-CC+CNN+GRU	0.42	0.25	0.41	0.39
	FastText-Wiki+CNN+GRU	0.40	0.35	0.31	0.44
OLID	***GloVe-Twitter*+CNN+FC**	0.39	0.43	0.31	0.43
	GloVe-CC+CNN+GRU	0.46	0.19	0.36	0.23
	FastText-CC+CNN+GRU	0.39	0.08	0.33	0.22
	FastText-Wiki+CNN+GRU	0.44	0.11	0.39	0.20
Combined	*GloVe-Twitter*+CNN+GRU	0.43	0.41	0.41	0.50
	GloVe-CC+CNN+GRU	0.45	0.40	0.41	0.50
	FastText-CC+CNN+GRU	0.45	0.27	0.46	0.44
	***FastText-Wiki*+CNN+GRU**	**0.46**	**0.44**	**0.40**	**0.57**

TABLE VI
ENSEMBLE CLASSIFIER MODELS TRAINED ON THE HON AND OLID DATASETS USING WEAK SUPERVISION INSTEAD OF THE DATASET LABELS. RESULTS ARE REPORTED FOR EACH MODEL EVALUATED ON ITS RESPECTIVE DATASET.

Weak Supervised Ensembles

Train Data	**Topology**	**F1-Macro**	**Hate Recall**
OLID	*GloVe-Twitter*+CNN+GRU	0.29	0.24
	***GloVe-CC*+CNN+GRU**	0.31	0.27
	FastText-CC+CNN+GRU	0.29	0.15
	FastText-Wiki+CNN+GRU	0.30	0.19
Combined	*GloVe-Twitter*+CNN+GRU	0.40	0.49
	GloVe-CC+CNN+GRU	0.41	0.43
	FastText-CC+CNN+GRU	0.42	0.50
	***FastText-Wiki*+CNN+GRU**	0.42	0.52

training, especially when used in concert with classifier tuning. Table VII illustrates the performance of the Gab classifier ensembles. The best weak supervised ensemble trained on unannotated Gab posts achieved a 62% hate recall, showing an 8% improvement after tuning. In all, our weak supervised models are able to achieve consistent performance on hate recall when evaluated on novel data from Gab, surpassing the hate recall of state of the art deep models on the HON dataset. These weak ensembles, as well as the standard pre-trained ensembles trained on the labeled HON and OLID datasets, demonstrate significant improvement in performance from the application of transfer learning by tuning models to an unseen target dataset, such as Gab.

VI. CONCLUSIONS & FUTURE WORK

In conclusion, in this paper we set out to address the persistent issues of class imbalance and model variability that pose serious challenges for automatic hate speech detection models. Our experiments focus on two major questions regarding how to improve current classification models for automatic hate speech detection and how to develop a generalized framework for hate speech detection that can be extended to unlabeled data. We advance current state of the art models by developing an improved model framework that leverages existing deep topologies to create an ensemble model for automatic hate speech detection Our experiments demonstrate that an ensemble approach outperforms individual models by an average of 5% hate recall and 8% F1 macro and is able to improve upon state of the art performance on the HON dataset, achieving 83% recall on the hate class—a 33% increase over comparable deep models.

We also show that through the application of transfer learning techniques, we are able to tune our classifier ensembles using a small, curated sample of labeled posts from a particular target dataset. In our experiments, we tune our models trained on the labeled HON and OLID datasets to data from Gab. Tuning improved performance on the Gab dataset by an average of 12% hate recall, and the best tuned ensemble models achieved 57% hate recall. We also develop a weak supervision methodology to train and tune our framework entirely on unlabeled data. Our best weak supervised model achieved a 67% hate recall on Gab.

The vocabulary of hate speech online continues to evolve rapidly along with the nature of social media generally; therefore, adaptable and generalizable automatic techniques for hate speech detection are crucial in order to keep up with the pace of change online. Our simple ensemble and tuning approaches show promise and provide avenues for future work on improving co-training routines and decision making processes for the ensemble, with additional further experimentation to optimize the parameters for both the weak supervision and tuning procedures. Finally, we should seek to develop weak superivsion techniques that are better suited to the evolving nature of hate speech online by forgoing fixed lexicons in favor of more flexible methods.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
COMPARISON OF WEAK SUPERVISED ENSEMBLES EVALUATED ON THE *Gab* DATASET. RESULTS ARE REPORTED BEFORE AND AFTER TUNING IS APPLIED TO THE MODELS. TUNED ENSEMBLES SHOWED AN AVERAGE IMPROVEMENT OF 12% ON HATE RECALL

Weak Supervised Ensembles - Gab

		Pre-tuning		Post-tuning	
Train Data	**Topology**	**F1-Macro**	**Hate Recall**	**F1-Macro**	**Hate Recall**
OLID	*GloVe-Twitter*+CNN+GRU	0.31	0.20	0.37	0.59
	***GloVe-CC*+CNN+GRU**	**0.35**	**0.25**	**0.40**	**0.67**
	FastText-CC+CNN+GRU	0.33	0.13	0.40	0.49
	FastText-Wiki+CNN+GRU	0.35	0.23	0.39	0.54
Combined	***GloVe-Twitter*+CNN+GRU**	0.25	0.26	0.31	0.50
	GloVe-CC+CNN+GRU	0.42	0.27	0.32	0.42
	FastText-CC+CNN+GRU	0.27	0.26	0.34	0.39
	FastText-Wiki+CNN+GRU	0.25	0.24	0.32	0.37
Gab	*GloVe-Twitter*+CNN+GRU	0.31	0.51	0.36	0.53
	GloVe-CC+CNN+GRU	0.33	0.49	0.37	0.55
	***FastText-CC*+CNN+GRU**	0.36	0.54	0.41	0.62
	FastText-Wiki+CNN+GRU	0.32	0.61	0.38	0.60

ACKNOWLEDGMENTS

The authors thank Michael Ridenhour and the rest of NASCL at UNC-Charlotte for their support in the execution of this work.

REFERENCES

[1] Z. Zhang, D. Robinson, and J. Tepper, "Detecting hate speech on twitter using a convolution-gru based deep neural network," 03 2018.

[2] H. Watanabe, M. Bouazizi, and T. Ohtsuki, "Hate speech on twitter: A pragmatic approach to collect hateful and offensive expressions and perform hate speech detection," *IEEE Access*, vol. 6, pp. 13 825–13 835, 2018.

[3] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ser. ICWSM '17, 2017, pp. 512–515.

[4] G. Rizos, K. Hemker, and B. Schuller, "Augment to prevent: Short-text data augmentation in deep learning for hate-speech classification," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 991–1000. [Online]. Available: https://doi.org/10.1145/3357384.3358040

[5] S. Zannettou, B. Bradlyn, E. De Cristofaro, H. Kwak, M. Sirivianos, G. Stringini, and J. Blackburn, "What is gab: A bastion of free speech or an alt-right echo chamber," in *ACM WWW*, 2018, pp. 1007–1014.

[6] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Hate speech detection: Challenges and solutions," *PloS one*, vol. 14, no. 8, p. e0221152, 2019.

[7] A. Vaidya, F. Mai, and Y. Ning, "Empirical analysis of multi-task learning for reducing identity bias in toxic comment detection," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, 2020, pp. 683–693.

[8] K. Garimella, G. De Francisci Morales, A. Gionis, and M. Mathioudakis, "Political discourse on social media: Echo chambers, gatekeepers, and the price of bipartisanship," in *Proceedings of the 2018 World Wide Web Conference*, 2018, pp. 913–922.

[9] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in *Proceedings of the 26th International Conference on World Wide Web Companion*, 2017, pp. 759–760.

[10] A. Arango, J. Pérez, and B. Poblete, "Hate speech detection is not as easy as you may think: A closer look at model validation," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 45–54. [Online]. Available: https://doi.org/10.1145/3331184.3331262

[11] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the Type and Target of Offensive Posts in Social Media," in *Proceedings of NAACL*, 2019.

[12] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *J. Mach. Learn. Res.*, vol. 11, p. 625–660, Mar. 2010.

[13] M. Dehghani, H. Zamani, A. Severyn, J. Kamps, and W. B. Croft, "Neural ranking models with weak supervision," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 65–74. [Online]. Available: https://doi.org/10.1145/3077136.3080832

[14] E. Raisi and B. Huang, "Weakly supervised cyberbullying detection using co-trained ensembles of embedding models," in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018, pp. 479–486.

[15] T. Gröndahl, L. Pajola, M. Juuti, M. Conti, and N. Asokan, "All you need is "love": Evading hate-speech detection," *CoRR*, vol. abs/1808.09115, 2018. [Online]. Available: http://arxiv.org/abs/1808.09115

[16] L. Lima, J. C. S. Reis, P. Melo, F. Murai, L. Araujo, P. Vikatos, and F. Benevenuto, "Inside the right-leaning echo chambers: Characterizing gab, an unmoderated social system," in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018, pp. 515–522.

[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[18] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: http://www.aclweb.org/anthology/D14-1162

[19] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," 2016.

[20] S. Zimmerman, C. Fox, and U. Kruschwitz, "Improving hate speech detection with deep learning ensembles," 05 2018.

[21] T. Semwal, P. Yenigalla, G. Mathur, and S. B. Nair, *A Practitioners' Guide to Transfer Learning for Text Classification using Convolutional Neural Networks*, pp. 513–521. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611975321.58

[22] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," 2018.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on March 09,2021 at 18:14:58 UTC from IEEE Xplore. Restrictions apply.