“Define Your Terms” : Enhancing Efficient Offensive Speech Classification with Definition

Huy Nghiem
University of Maryland, College Park
nghiemh@umd.edu

Umang Gupta and **Fred Morstatter**
USC Information Sciences Institute
umanggup@usc.edu, fred@isi.edu

Abstract

The propagation of offensive content through social media channels has garnered attention of the research community. Multiple works have proposed various semantically related yet subtle distinct categories of offensive speech. In this work, we explore meta-learning approaches to leverage the diversity of offensive speech corpora to enhance their reliable and efficient detection. We propose a joint embedding architecture that incorporates the input's label and definition for classification via Prototypical Network. Our model achieves at least 75% of the maximal F1-score while using less than 10% of the available training data across 4 datasets. Our experimental findings also provide a case study of training strategies valuable to combat resource scarcity.

1 Introduction

While a vital channel for the dissemination of crucial information, social media platforms have also become hotbeds for hateful, and harmful expressions. Such offensive speech not only detracts from the quality of discourse but also poses tangible threats to marginalized and vulnerable groups, escalating existing social tensions. Multiple studies have observed the psychological harms to marginalized communities perpetuated by offensive content in the digital space (Saha et al., 2019; Ștefăniță and Buf, 2021). However, the definition of offensive speech varies between contexts, and across publications that study this problem. A common challenge with offensive speech is the lack of a unifying definition, with conceptually related but definitively distinct categories proposed in literature: *Hate, Abusive, Aggressive, Toxic, Offensive, Cyberbullying etc.* (Poletto et al., 2021; Yin and Zubiaga, 2021). While earlier research focused on binary classification, more current works have explored offensive categories in higher granularity and semantic diversity (Mullah and Zainon, 2021; Caselli et al., 2021; Mozafari et al., 2020; ElSherief et al., 2021a; Yin and Zubiaga, 2021).

Though an active area of research in the Natural Language Processing (NLP) community, accurate and reliable detection of offensive speech often requires significant amount of training data (Vidgen and Derczynski, 2020; Goodfellow et al., 2016). For these tasks, the typical pipeline of data collection involves gathering a candidate corpus based on a set of relevant keywords, then soliciting task-specific labels for them via crowdsourcing or expert annotation (Vidgen and Derczynski, 2020; Paullada et al., 2021). Demographics of annotators may be different from one dataset to the next, including platforms (e.g. Amazon Mechanic Turk, Prolific), payments, levels of education, languages and cultural backgrounds (Founta et al., 2018).

As offensive content is frequently linked to real world events, there exists a need for appropriately tailored datasets. Nevertheless, constructing a sufficient amount of labelled data often proves a resource-intensive challenge (Poletto et al., 2021; Founta et al., 2018; Toraman et al., 2022). On the other hand, there exists a plethora of available data on similar yet categorically distinct areas of offensive speech. We set out with the objective to discover suitable techniques capable of leveraging existing datasets to *efficiently and reliably adapt* to new domains of offense content.

To this end, we compile from literature a collection of 14 relevant datasets, which allow us to perform a battery of testing on various pre-training and meta-learning approaches to assess their efficacy and robustness in classification of offensive content. We also experiment with different model architectures to incorporate label information to enhance knowledge transference at multiple levels of data availability. We introduce **JE_ProtoNet**, a *joint embedding* based on Prototypical Network which utilizes **definition of label categories** and exhibit competitive performance across 4 test sets while

Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1293–1309

using a fraction of the available training data. To the best of our knowledge, this work is the first in literature that harnesses label definition in offensive speech detection. Our experiments provide a case study on the trade-off between sample efficiency and performance, with findings potentially applicable to any classification task where categories entail more nuanced expression beyond simple labels. Based on empirical findings, we provide a set of recommendations to leverage our approach to enhance the efficiency of offensive speech classification.

2 Related Work

2.1 Annotation with Instructions

High-quality annotation is crucial to the development of offensive speech classifiers. Annotators' implicit biases and disagreements could be propagated and even magnified by downstream models (Waseem, 2016; Vidgen and Derczynski, 2020; Davani et al., 2023; Akhtar et al., 2020). Explicitly priming annotators with clear instructions and definitions have been shown to reduce biases and enhance inter-annotator agreements (Sap et al., 2019a; Waseem, 2016; Parmar et al., 2023).

2.2 Cross-Dataset Transference

The diversity of datasets on offensive speech has prompted researchers to investigate their generalizability. Models' performance tend to significantly drop when applied to out-of-domain dataset (Bansal and Villavicencio, 2019; Yin and Zubiaga, 2021). Fortuna et al. (2021)'s extensive study revealed that cross-dataset transference is highly influenced by their semantic similarity. Some works, such as HateBERT and fBERT, pre-trained the base model on specialized corpora to allow better adaptation to new datasets (Caselli et al., 2021; Sarkar et al., 2021).

Model architecture and fine-tuning strategy could also enhance transferrability. Mozafari et al. (2022) applied Model-Agnostic Meta-Learning (MAML) and Proto-MAML to BERT-based (Devlin et al. (2018)) models and observed improvements in few-show cross-lingual hate speech detection. Kim et al. (2022) used contrastive learning to enhance detection of implicit hate speech detection across three benchmarks. Tran et al. (2020) constructed HABETOR with fewer parameters but still demonstrated good generalizable performance across 2 out-of-domain datasets.

2.3 Label-Aware Classification

The idea of constructing label embedding was pioneered by Tang et al. (2015) in their work on Predictive Text Embedding. Wang et al. (2018) followed up with Label-Embedding Attentive Model (LEAM), a joint embedding of words and labels downstream classification task. More recently, Xiong et al. (2021a) leveraged BERT's self-attention mechanism for classification by concatenating labels' tokens directly into their respective inputs. Luo et al. (2021) took this idea further in their method Label-semantic Augmented Meta-Learner (LaSAML) via Prototypical Network, a framework capable of few-shot text classification.

3 Data Collection and Processing

General Criteria

Our goal is to leverage existing datasets to adapt to new domains of offensive content in a reliable and label-efficient manner. To this end, we survey literature[1] to identify relevant existing datasets on offensive speech and related topics. We filter our options based on the following criteria: size (> 10,000 samples), diversity of label categories (or, the nature of offensive text these labels capture), availability of definitions and instructions, along with method of annotation. We strive to incorporate a sufficient number of categories related to offensive speech with distinct levels of granularity. When definitions of label categories are unavailable in the original work, we solicit this content from their authors. Detailed definitions for the labels are included in Tables 5 and 6 of the Appendix. Ultimately 14 datasets are chosen (Table 1), with the 4 below held out for final testing of the models and are not used for any pre-training.

ToxiGen is a large-scale machine-generated dataset by demonstration-based prompting. Hartvigsen et al. (2022a) controlled machine generation to create a corpus of *Benign* and *Toxic* texts that cover 13 identity groups. In addition to its unique nature of construction, this dataset is included as a representative for binary classification tasks.

HateXplain is constructed by Mathew et al. (2021) with an emphasis on explanability. The authors asked annotators to highlight the span of tokens, called *rationales*, that contribute to their selection of the labels. This dataset shares the same label space with Davidson et al. (2017), yet with differ-

[1] https://hatespeechdata.com

ent definitions for each term.

Implicit_hate is developed by ElSherief et al. (2021b) to fill the gap in the literature with respect to negative sentiments expressed in coded or indirect language. As the corresponding authors posited, detecting implicit hate speech is regarded as more challenging than its overt counterpart.

Covid focuses on the rise of anti-Asian sentiments fueled by the COVID-19 pandemic (Vidgen et al., 2020). Among other COVID-related hate speech corpora (Nghiem and Morstatter, 2021; He et al., 2021), this dataset arguably considers the most nuanced categories of East Asian entities.

Train Set Sampling

The remaining 10 datasets are reserved for meta-training. Data "in the wild" tends to have considerably different distributions with very low representation of the offensive classes (Poletto et al., 2021). Further, the offensive content is frequently deleted from the platforms, making retrieval for research even more challenging (Poletto et al., 2021; Vidgen et al., 2021). Some datasets in our collections contain classes that suffer from extremely low prevalence. To alleviate these problems, we only select label classes that have clear definitions and significant samples relative to their respective set. Then, we employ stratified sampling to create a subset while maintaining as close to an equal distribution between classes as feasible. These sample sizes are reflected in Table 1, with a final tally of 82,000. Finally, we perform pre-processing steps to standardize texts (details in Appendix A).

4 Experimental Setup

From here on, we refer to the datasets as *domains*. With the goal of investigating the potential benefits of learning from semantically related but distinct data, our general experiment pipeline consists of first pretraining a model on the 10 reserved domains using different techniques [2]. Then, the model is fine-tuned and evaluated on *each* of the 4 aforementioned test domains. More specifically, we hold out a portion of the test domains using the ratio described in their original publications (Table 1). We then perform K-shot sampling of the remaining data to fine-tune the pre-trained models where K $\in \{16, 32, 64, 128, 256\}$. We used $K = 64$ from the leftover data to select hyper-parameters (details in Table 4 of the Appendix). Finally, the fine-tuned model is tested on the held-out dataset. The following sections describe our pretraining approaches.

4.1 Baselines

We select base RoBERTa (Robustly Optimized BERT approach) as implemented by the Huggingface library to be the main structure of our model due to its strong performance on related sentiment classification tasks (Liu et al., 2019; ElSherief et al., 2021a; Poletto et al., 2021). Already pretrained on a large English corpus in an unsupervised fashion, this version of RoBERTa contains 12 layers of transformer blocks, 12 attention heads, and approximately 125 million trainable parameters.

The baseline models[3] all use the [CLS] token from the embedding as input to the classification head – a fully connected layer – to produce logit scores for each label. The model seeks to minimize the Cross-Entropy loss, with parameters updated via `AdamW` Optimizer. The simplest baseline, **RoBERTa_untrained** refers to training with only K samples from the test domains (K-shot learning), then evaluating on the held-out portion without using any form of pretraining.

Inspired by Gururangan et al. (2020), the next variant, **RoBERTa_retrained**, trains the model on each of the test domain's entire (non-sampled) training set using the Mask Language Model's objective in a self-supervised manner, before being further fine-tuned through supervised learning with the K-shot samples.

Finally, **RoBERTa_binary**, incorporates the 82,000 samples in a simple fashion. We unify the different domains by collapsing the disparate label spaces into a binary mapping: all non-neutral categories into *Offensive*, and the rest into *Not Offensive*. The model is pretrained on this unified dataset using the supervised learning objective before being fine-tuned in a K-shot way on in-domain samples. Additionally, we also K-shot fine-tune then evaluate out-of-the-box **HateBERT** (Caselli et al., 2021) for comparison.

4.2 Meta-Learning Settings

In this section, we explore various meta-learning frameworks as a means of pre-training. The following frameworks all simulate N-way K-shot learning, where N is the number of classes (labels) in a

[2] Our code repository is available at: https://github.com/hnghiem-usc/define_your_terms

[3] We use Huggingface's RobertaForSequenceClassification implementation

Dataset	Total Size	Sample Size	Platform	Annotation Method	Selected Labels
Waseem and Hovy, 2016	16,914	3,000	Twitter	E	Offensive, Not Offensive
Golbeck et al., 2017	20,360	10,000	Twitter	E	Harrassment, Not Harrassment
Davidson et al., 2017	24,800	5,000	Twitter	C	Hate Speech, Offensive, Normal
Kumar et al., 2018	15,000	10,000	Facebook	-	Overly Aggressive, Covertly Aggressive, Non-Aggressive
Founta et al., 2018	80,000	10,000	Twitter	C	Normal, Abusive Language, Hate Speech
Zampieri et al., 2019	14,100	6,000	Twitter	C	Targeted Insult, Untargeted Insult, Not Offensive
Basile et al., 2019	13,000	8,000	Twitter	C	Hate Speech, Not Hate Speech
Sap et al., 2019b	44,671	10,000	Reddit, Twitter, Gab, Stormfront	C	Offensive, Not Offensive
Vidgen et al., 2021	41,255	10,000	-	C	Derogation, Animosity, Threatening, Support for Hateful Entities, Dehumanization, Neutral
Toraman et al., 2022	100,000	10,000	Twitter	E	Offensive, Hate, Normal
ToxiGen	274,186	*2,740*	Synthetic	-	Toxic, Benign
HateXplain	20,148	*2,000*	Gab, Twitter	C	Hate Speech, Offensive, Normal
Implicit_hate	6,346	*1,340*	Twitter	E, C	White Grievance, Incitement to Violence, Inferiority Language, Irony, Stereotypes and Misinformation, Threatening and Intimidation
Covid	20,000	*2,000*	Twitter	E	Hostility against an East Asian Entity, Criticism of an East Asian Entity, Discussion of East Asian Prejudice, None of the Above
Total	688,587	82,000			

Table 1: General information about the compiled data sources. For *Annotation Method*, E stands for *Expert*, where trained annotators are selected for labeling, and C for *Crowdsource*, a setting that employs a larger, typically non-specialized pool of workers. The last 4 datasets are reserved for eventual evaluation. The sample size of test sets (italicized) refers to values used for the final evaluation and is not included in the Total Size column.

domain, and K is the number of samples per class. Each learner (model) f is parameterized by θ, of which we seek to optimize over the classification tasks using the 10 reserved domains.

At each training episode, a *support* and *query* set of the same size is sampled from a domain $\mathcal{D}_i$, where $i \in \{1, 2, ..., 10\}$ for each of the reserved domains. For meta-training, K is restricted to $\{16, 32, 64, 128\}$ shots to accommodate domains with high number of categories. Since meta-training is computationally demanding, we train models on a single fixed seed and report aggregate results by K-shot fine-tuning the meta-trained model with 5 random seeds.

Training Without Label Information

In this standard setting, the learner's inputs do not incorporate any label information.

Prototypical Network, or ProtoNet, is a metric-based meta-learning framework (Snell et al., 2017). We use RoBERTa's [CLS] token as the encoded representation of each input. For each class $c \in \mathcal{C}$ in domain $\mathcal{D}_i$, a prototype $\mathbf{v}_c$ is constructed by taking the mean of all K samples:

$$\mathbf{v}_c = \frac{1}{|\mathcal{S}_c|} * \sum_{(\mathbf{x}_i, y_i) \in \mathcal{S}_c} f_\theta(\mathbf{x}_i) \quad (1)$$

where $\mathcal{S}_c$ denotes the support set for which $y_i = c$. Distribution over the classes is calculated by taking the softmax over the inverse distances d_φ (Eucliean in our work) between inputs' embedding and the prototypes.

$$p(y = c|\mathbf{x}) = \frac{exp(-d_\varphi(f_\theta(\mathbf{x}), \mathbf{v_c}))}{\sum_{c' \in \mathcal{C}} exp(-d_\varphi(f_\theta(\mathbf{x}), \mathbf{v_{c'}}))} \quad (2)$$

Input $\mathbf{x}$ is assigned the label of the nearest prototype.

ProtoMAML, an optimization-based framework, extends Model-Agnostic Meta-Learning

(MAML, Finn et al. (2017)), which aims to learn a good initialization of the learner's base parameters θ that can quickly adapt to new tasks with limited data. During meta-training, MAML optimizes the model virtually using the support set, then evaluates the gradients on the query set with respect to the original parameters. Designing the classification layer with MAML is challenging when tasks have different label spaces. To circumvent this problem, Triantafillou et al., 2019 proposed ProtoMAML, which incorporates Prototypical Network's strengths by reformulating the softmax over Euclidean distances as a linear layer with with softmax. By setting the weights of the linear layer to twice the prototypes, and the biases to the to the negative of the prototypes, we obtain a classification layer that would be compatible with any domain. We implement First-Order ProtoMAML in this work to avoid the computational cost of obtaining second-order derivatives as in the original MAML algorithm.

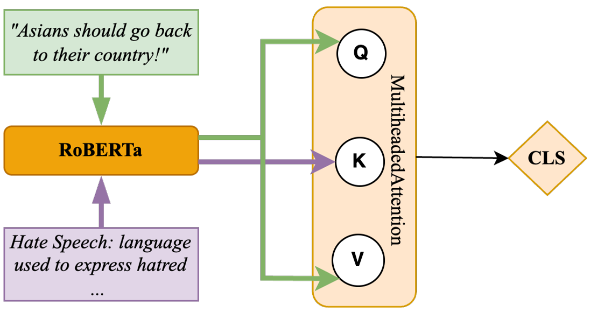


Figure 1: General architecture of JE_ProtoNet. Hidden states of text input and its corresponding label and definition are obtained from RoBERTa, and then passed as Query, Value, and Key (color coded for traceability) to the Multiheaded Attention module.

MLDG (Meta-Learning for Domain Generalization) was proposed by Li et al. (2018). To learn a good initialization suited for generalization, $\mathcal{S}$ domains are split into disjoint sets $\overline{\mathcal{S}}$ and $\mathcal{S}'$. During meta-training, MLDG updates the model's parameters virtually on tasks drawn from $\overline{\mathcal{S}}$ to using gradients $\nabla_\theta = \mathcal{F}'_\theta(\overline{\mathcal{S}}, \theta)$. During meta-training, the model is virtually evaluated on tasks drawn from $\mathcal{S}'$ to obtain loss $\mathcal{G}(\mathcal{S}'; \theta')$. The base model is optimized using both losses:

$$\theta = \theta - \gamma \frac{\partial(\mathcal{F}(\overline{\mathcal{S}}, \theta) + \beta\mathcal{G}(\mathcal{S}'; \theta - \alpha\nabla_\theta)}{\partial\theta}) \quad (3)$$

Inspired by Ye and Chao (2021); Kao et al. (2022), the classification head takes the [CLS] token as input to pass through a linear layer of the same dimension (768), whose output is further connected to a final fully connected layer of shape (768,1). At the beginning of each training episode, this layer is duplicated accordingly to the required number of classes for each domain, with the parameters' weights set to 0.

4.3 Training with Label Information

In this setting, label information is directly incorporated into the training inputs in various configurations. ProtoNet is the sole chosen architecture because its metric-based nature limits overfitting on labels compared to other methods. Label incorporation only happens during meta-training and fine-tuning. During test time, no label information is available to the model.

ProtoNet_Token For each domain $\mathcal{D}_i$, we convert the label L_j into new token E_{Lj} for all j in the label space. For labels that consist of multiple subwords, we construct E_{Lj} by averaging their token embeddings. Inspired by Xiong et al. (2021b) and Si et al. (2020), we concatenate the token embedding E_T of input T with its corresponding label token E_{Lj}, separated by the [SEP] token. Labels from different domains but share identical textual representation would also share their token embeddings.

ProtoNet_Label In contrast, this setting concatenate the corresponding label L_j directly to the end of each text input T, all of which are passed together to the model. This approach simplifies the label fusing process to create more discriminate representation of inputs (Luo et al., 2021).

ProtoNet_Full In this approach, we also utilize the definition associated with each label. Specifically, we construct the input to the model using the format [CLS] T [SEP] L_j : D_j [SEP], where D_j is the full definition of the corresponding label.

JE_ProtoNet We construct an architecture that takes into consideration the compatibility between the text inputs and the labels' definitions via a joint embedding (illustrated in Figure 1). Each input T is fed into the RoBERTa's backbone to obtain the hidden state representation H_T. Similarly, we obtain the hidden state H_D of the the corresponding label and definition sequence of the format $L_j : D_j$ using the same model. We then pass H_T as the Query and Value input, and H_D as the Key into the attention module [4] (Vaswani et al., 2017), which consists of 3 attention heads. In contrast to the

[4]We use Huggingface's MultiHeadAttention module

native self-attention mechanism seen in previous configurations, this setup allows the model to focus on certain aspects or parts of the input text semantically relevant to the given label definition. Finally, we extract the [CLS] token from the output of joint embedding for downstream classification as in other ProtoNet settings. During testing, a blank string is passed in lieu of the definition.

5 Results

Macro F-1 score is chosen as the evaluation metric. We first discuss the performance of models relative to each other in their respective setting, then provide a top-down analysis. Figure 2 illustrates the performance for each setting. Detailed numerical results are displayed in Table 7 of the Appendix. We also provide Figure 3 as an alternative illustration to facilitate comparison between models.

5.1 For Baseline Settings

Results for Baseline experiments are displayed in Figure 2a. Unsurprisingly, macro F1-scores improve with less variance as the number of K training shots increases. With the exception of ToxiGen's binary classification, models tend not to attain most of their classifying capability until K=128. *RoBERTA_untrained*, which uses no pre-training, displays a consistently improvement in performance with more data across all 4 test domains. In contrast, pre-training on in-domain data with the Mask Language Model objective, causes underperformance in some domains (HateXplain, Implicit_hate), while provides a boost for others at various K's (ToxiGen, Covid). Pre-training on binary collapsed data allows *Roberta_binary* to attain better F-1 scores in low-resource cases (K < 64), suggesting beneficial initialization from exposure to general data on offensive speech. Nevertheless, this method does not guarantee peak performance when more in-domain training data is available. This finding aligns with prior cross-domain hate speech experiments (Fortuna et al., 2021; Toraman et al., 2022), suggesting that the binary mapping scheme might overlook specific nuances unique to each domain, hindering generalization to new domains. Overall, these simple pre-training methods offer inconsistent performance.

5.2 For Meta-Learning Settings

5.2.1 Without Label

Optimization-based models require more training data ($K \geq 64$) to exhibit competitive performance. *Proto_MAML*'s F1-scores are inferior to those of *MLDG* in every setting. Furthermore, this model displays considerably more dispersed results between seeds than the others, especially at higher K values for ToxiGen and Covid. These factors suggest that relying on the inductive bias using prototype-based initialization of the classifier may not enhance generalization between domains. On the other hand, *MLDG*, specifically designed for domain generalization, appears to perform comparably to *Robeta_binary* at the extremes of K values, with similar trajectory in between. Nevertheless, this model's performance also displays higher standard deviation for $K \in \{128, 256\}$ for HateXplain and Covid.

ProtoNet has the distinction of offering the best F1-scores at K=16 for all test domains and consistently stable results for different seeds, thanks to its metric-based nature. However, this feature also appears to hamper its classifying power even when exposed to more in-domain training data, showing little improvement at maximum value of K.

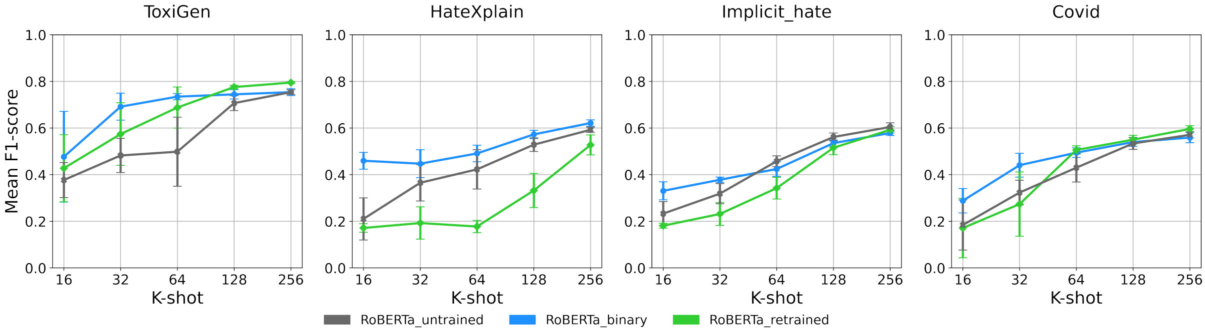
5.2.2 With Label

Though all *ProtoNet*-based models exhibit relatively stable performance across evaluation seeds as in previous setting, their trajectories differ when trained on more in-domain data (Figure 2c). Interestingly, *ProtoNET_Label*'s performance deteriorates as K increases, along with higher standard deviation in comparison to other variants.

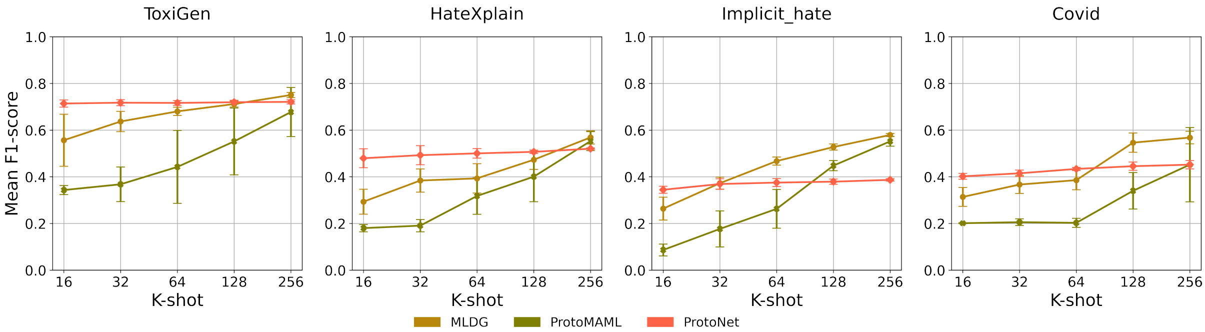
Appending the entire definition to the input also does not appear to be viable, as *ProtoNet_Full* yields the second least favorable F-1 scores for HateXPlain, Implicit_hate, and Covid domains. *ProtoNet_token*, though yielding more favorable results compared to the 2 previous variants, do not demonstrate significant difference in performance compared to the base *ProtoNet* setting in 5.2.1.

JE_ProtoNet is the only model whose performance appreciatively scales with increment of K values. This architecture achieves competitive F1-scores at $K \in \{16, 32\}$, with notably higher result for Implicit_hate. More importantly, *JE_ProtoNet* outperforms other with-label variants across all test domains, demonstrating its robustness.

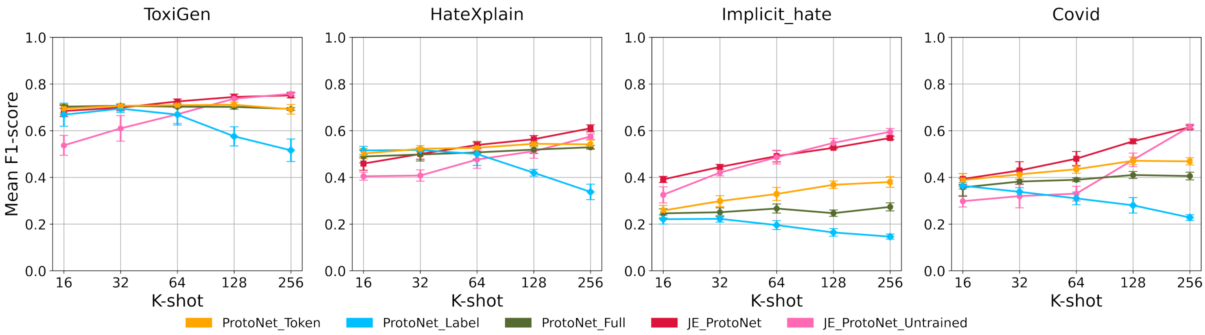
Does pre-training help initialize Joint



(a) Results for Baseline RoBERTa models



(b) Results for Meta-Learning models without labels



(c) Results for ProtoNet-based models with labels

Figure 2: Illustration of Macro F1-scores of models for various K-shot settings. Vertical bars denote standard deviation of results over 5 seeds. HateBERT is ommited to simplify comparison.

Embedding? We perform testing on the *JE_ProtoNet* model, whose Attention module's weights are randomly initialized, without any pre-training on the 10 datasets, denoted as *JE_ProtoNet_Untrained*. In Figure 2c, we observe that *JE_ProtoNet_Untrained*'s F1-scores are inferior to that of its counterpart *JE_ProtoNet* across domains for K$\leq$128, except for *Implicit_hate* domain at K=64. Additionally, the former generally exhibits higher variance among results compared to the latter model for K$\leq$64. These notions suggest that our pre-training approach via meta-learning provides advantageous initialization when in-domain resource is scarce.

Leveraging JE_ProtoNet's features Observing the discrepancy in improvement with more resources (higher K-shots) of Baseline models, and ProtoNet-based models' good performance in low-resource setting, we hypothesize that it is possible to enhance *JE_ProtoNet* to overcome its limited inductive bias while fully utilizing its learned discriminate features. We thus equip *JE_ProtoNet* with a classification head, a feed forward neural network that takes the [CLS] token form the joint embedding as input. This model, **JE_ProtoNet_CLS**, is discussed in the next section.

5.2.3 Global Assessment

We restrict our analysis here to K=256. From Table 2, we observe that all models perform respectably on *ToxiGen*'s classification task. This finding is in line with the dataset's conditional

	ToxiGen		HateXplain		Implicit_hate		Covid	
	F1	σ	**F1**	σ	**F1**	σ	**F1**	σ
HateBERT	0.731	0.007	0.571	0.007	0.523	0.010	0.542	0.020
RoBERTa_untrained	0.753	0.010	0.592	0.011	**0.604**	0.018	0.571	0.014
RoBERTa_binary	0.754	0.015	*0.621*	0.014	0.578	0.011	0.559	0.022
RoBERTa_retrained	**0.795**	0.004	0.527	0.043	0.592	0.013	0.596	0.014
ProtoNet	0.721	0.009	0.520	0.003	0.387	0.005	0.452	0.018
ProtoMAML	0.677	0.105	0.553	0.041	0.552	0.021	0.452	0.159
MLDG	0.750	0.011	0.569	0.028	0.579	0.006	0.568	0.027
ProtoNet_Token	0.691	0.021	0.541	0.018	0.380	0.023	0.469	0.015
ProtoNet_Label	0.516	0.048	0.338	0.033	0.146	0.011	0.228	0.012
ProtoNet_Full	0.693	0.006	0.529	0.008	0.274	0.017	0.406	0.016
JE_ProtoNet	0.751	0.010	0.610	0.015	0.569	0.007	0.615	0.008
JE_ProtoNet_Untrained	*0.758*	0.008	0.575	0.011	*0.595*	0.014	*0.617*	0.011
JE_ProtoNet_CLS	*0.758*	0.005	**0.628**	0.007	*0.595*	0.018	**0.636**	0.031

Table 2: Macro F1-scores and their standard deviation (σ) for K = 256. Highest and second-highest F1-scores in each test domain are **bolded** and *italicized*, respectively.

	Train size	No. class	F1 Max	64		128		256	
				F1%	Size %	F1%	Size%	F1%	Size%
ToxiGen	512*	2	0.795	91	25	93	50	95	100
HateXplain	16,118	3	0.687	79	1	84	2	91	5
Implicit_hate	3,807	6	0.586	88	10	96	20	102	39
Covid	16,000	4	0.832	53	2	66	3	76	6

Table 3: Comparison of JE_ProtoNet_CLS performance and size across K values {64, 128, 256}. *F1 %* represents the model's F1-score relative to the highest F1-score (*F1 max*) reported by the original authors using the corresponding training data size (*Train size*). *Size %* indicates the sample size percentage based on the K-value relative to the *Train size*. *Best F1-score attained by *RoBERTa_binary* at K=256 chosen (statistics not reported by original authors)

machine-generation of its binary labels. *HateBERT*'s performance generally trails behind our baseline models, indicating this model's struggle to adapt to new domains in few-shot settings. Interestingly, pre-training on in-domain data allows RoBERTa to leading F1-score of 0.795. On the other hand, *RoBERTa_untrained* achieves the leading score of 0.604 on *Implicit_hate*. Nevertheless, none of the baseline models obtain consistently good performance across the board. Meta-learning models without labels also do not produce competitive results. This group's top performer, *MLDG*, attains only decent results across test domains.

As discussed in 5.2.2, *ProtoNet* with various methods to incorporate label information do not yield improvement over their non-label counterpart, and may even exhibit degradation (*ProtoNet_Full*). Using joint embedding that incorporates label definitions, however, achieves both strong and consistent F-1 scores, as shown by all configurations of *JE_ProtoNet* models. In fact, *JE_ProtoNet_CLS* attains best or second best results in all 4 test domains, especially the 0.636 F1-score for *Covid*, arguably the most semantically distinctive domain.

6 Discussion

Definition matters While many works in offensive speech literature have focused on standard classification techniques, ours is the first to leverage the definition of associated labels. Our proposed framework to incorporate definition via the joint embedding is beneficial to boost classification performance over other models, given the same amount of training data. In addition to enhancing annotation quality, this factor is yet another signal to encourage researchers to pay more attention to their terminologies to both enhance downstream tasks and facilitate cross-task studies.

More data is not always needed Our experiments provide a case study on how much data is necessary to achieve certain results in the area of offensive speech detection. While having more labeled data is always preferable, the annotation process can be expensive, and thus constituting a barrier to researchers not equipped with abundant resources. Table 2 describes the percentages of *JE_ProtoNet_CLS*'s F1-scores for K=64 to 256 relative to the F1-scores reported by the original authors using the entire training data. Most notably, our balanced data sampling and model design achieve 79% of max F1-score with 1% training data for *HateXplain*, 76% with 6% training data for *Covid*, and even bested the max F1-score with only 39% training data for *Implicit_Hate*.

Recommendations for Low-Resource Settings This observation suggests that tailoring the data annotation process for class balance may allow offensive speech classifiers to attain better performance with less training resource. For instance, practitioners may opt to iterate over collecting, annotating, testing and increasing the quantity of data using classification metrics as guiding criteria. Experimental results suggest that setting K = 64 may be a good starting point. As our technique does not incur significant technical overhead compared to baseline architectures, researchers may implement both to mutually juxtapose during this iterative data collection process, and stop when the models' performances plateau or reach a satisfactory threshold. This approach has the advantage of being both data-efficient and empirically driven.

7 Conclusion

While we also leverage the existing rich corpora, our work explores a different setting of offensive speech detection compared to other works, such as HateBERT or fBERT (Caselli et al., 2021; Sarkar et al., 2021). The proposed joint-embedding may be adapted to complement other existing architectures. Our approach can also be applied to other NLP tasks, such as sentiment analysis and stance detection, where labels extend beyond compact phrases. We invite researchers to explore definitions and the extent of their usefulness in other tasks.

Offensive content is ever-evolving in today's world. We hope that our findings provide useful pointers for NLP practitioners to more efficiently explore diverse topics in this field.

Limitations

Our pre-processing step that removes special characters and casts inputs into lower-case is chosen for efficiency and to facilitate fair comparisons between the various experimental configurations. It is possible that these characters provide additional predictive signal, and could be used to enhance the models' performance.

This work uses RoBERTa as the sole backbone architecture for our models. In recent years, a plethora of new, potentially more powerful architectures have been proposed and may obtain better performance on our tasks. Furthermore, our corpora all focus on English, which does not reflect the diversity of languages, cultural norms and expressions that can express offensive sentiment. Our classification tasks only explore label categories, while other works also explicitly predict the targets of offensive content. Finally, definition for label is not always available for all offensive speech datasets. It remains an open research question if our method will transfer to other domains, not limited to offensive speech. We invite interested researchers to explore these venues.

Ethics Statement

This research aims to reduce the spread of offensive content by means of more reliably detecting them. Our compiled datasets do not violate privacy as they are extracted from published works, whose authors have taken steps to uphold confidentiality. We acknowledge that, due to the open nature of this data, they might contain references to real life personnel. There exists a risk that nefarious parties may leverage the ideas proposed in this work in the opposite of the authors' intention to propagate more offensive speech instead.

References

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2020. Modeling annotator perspective and polarized opinions to improve hate speech detection. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 8, pages 151–154.

Mohit Bansal and Aline Villavicencio. 2019. Proceedings of the 23rd conference on computational natural language learning (conll). In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 54–63.

Tommaso Caselli, Valerio Basile, Mitrovic Jelena, Granitzer Michael, et al. 2021. Hatebert: Retraining bert for abusive language detection in english. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25. Association for Computational Linguistics.

Aida Mostafazadeh Davani, Mohammad Atari, Brendan Kennedy, and Morteza Dehghani. 2023. Hate speech classifiers learn normative social stereotypes. *Transactions of the Association for Computational Linguistics*, 11:300–319.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. 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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. 2021a. Latent hatred: A benchmark for understanding implicit hate speech. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 345–363.

Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. 2021b. Latent hatred: A benchmark for understanding implicit hate speech. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 345–363, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR.

Paula Fortuna, Juan Soler-Company, and Leo Wanner. 2021. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? *Information Processing & Management*, 58(3):102524.

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

Jennifer Golbeck, Zahra Ashktorab, Rashad O Banjo, Alexandra Berlinger, Siddharth Bhagwan, Cody Buntain, Paul Cheakalos, Alicia A Geller, Rajesh Kumar Gnanasekaran, Raja Rajan Gunasekaran, et al. 2017. A large labeled corpus for online harassment research. In *Proceedings of the 2017 ACM on web science conference*, pages 229–233.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022a. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326, Dublin, Ireland. Association for Computational Linguistics.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022b. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326.

Bing He, Caleb Ziems, Sandeep Soni, Naren Ramakrishnan, Diyi Yang, and Srijan Kumar. 2021. Racism is a virus: Anti-asian hate and counterspeech in social media during the covid-19 crisis. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 90–94.

Chia-Hsiang Kao, Wei-Chen Chiu, and Pin-Yu Chen. 2022. Maml is a noisy contrastive learner in classification. In *International Conference on Learning Representations*.

Youngwook Kim, Shinwoo Park, and Yo-Sub Han. 2022. Generalizable implicit hate speech detection using contrastive learning. In *Proceedings of the 29th International Conference on Computational Linguistics*, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Ritesh Kumar, Aishwarya N Reganti, Akshit Bhatia, and Tushar Maheshwari. 2018. Aggression-annotated corpus of hindi-english code-mixed data. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. 2018. Learning to generalize: Metalearning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

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

Qiaoyang Luo, Lingqiao Liu, Yuhao Lin, and Wei Zhang. 2021. Don't miss the labels: Label-semantic augmented meta-learner for few-shot text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2773–2782.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14867–14875.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. 2020. A bert-based transfer learning approach for hate speech detection in online social media. In *Complex Networks and Their Applications VIII: Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019 8*, pages 928–940. Springer.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. 2022. Cross-lingual few-shot hate speech and offensive language detection using meta learning. *IEEE Access*, 10:14880–14896.

Nanlir Sallau Mullah and Wan Mohd Nazmee Wan Zainon. 2021. Advances in machine learning algorithms for hate speech detection in social media: a review. *IEEE Access*, 9:88364–88376.

Huy Nghiem and Fred Morstatter. 2021. " stop asian hate!": refining detection of anti-asian hate speech during the covid-19 pandemic. *arXiv preprint arXiv:2112.02265*.

Mihir Parmar, Swaroop Mishra, Mor Geva, and Chitta Baral. 2023. Don't blame the annotator: Bias already starts in the annotation instructions. In *17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023*, pages 1771–1781. Association for Computational Linguistics (ACL).

Amandalynne Paullada, Inioluwa Deborah Raji, Emily M Bender, Emily Denton, and Alex Hanna. 2021. Data and its (dis) contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11).

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. 2021. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, 55:477–523.

Koustuv Saha, Eshwar Chandrasekharan, and Munmun De Choudhury. 2019. Prevalence and psychological effects of hateful speech in online college communities. In *Proceedings of the 10th ACM conference on web science*, pages 255–264.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A Smith. 2019a. The risk of racial bias in hate speech detection. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 1668–1678.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A Smith, and Yejin Choi. 2019b. Social bias frames: Reasoning about social and power implications of language. *arXiv preprint arXiv:1911.03891*.

Diptanu Sarkar, Marcos Zampieri, Tharindu Ranasinghe, and Alexander Ororbia. 2021. fbert: A neural transformer for identifying offensive content. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1792–1798.

Shijing Si, Rui Wang, Jedrek Wosik, Hao Zhang, David Dov, Guoyin Wang, and Lawrence Carin. 2020. Students need more attention: Bert-based attention model for small data with application to automatic patient message triage. In *Machine Learning for Healthcare Conference*, pages 436–456. PMLR.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1165–1174.

Cagri Toraman, Furkan Şahinuç, and Eyup Halit Yilmaz. 2022. Large-scale hate speech detection with cross-domain transfer. *arXiv preprint arXiv:2203.01111*.

Thanh Tran, Yifan Hu, Changwei Hu, Kevin Yen, Fei Tan, Kyumin Lee, and Se Rim Park. 2020. Habertor: An efficient and effective deep hatespeech detector. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7486–7502.

Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. 2019. Meta-dataset: A dataset of datasets for learning to learn from few examples. In *International Conference on Learning Representations*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Bertie Vidgen and Leon Derczynski. 2020. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *PLoS ONE*, 15(12).

Bertie Vidgen, Scott Hale, Ella Guest, Helen Margetts, David Broniatowski, Zeerak Waseem, Austin Botelho, Matthew Hall, and Rebekah Tromble. 2020. Detecting East Asian prejudice on social media. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 162–172, Online. Association for Computational Linguistics.

Bertie Vidgen, Tristan Thrush, Zeerak Waseem, and Douwe Kiela. 2021. Learning from the worst: Dynamically generated datasets to improve online hate detection. 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 1667–1682.

Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint embedding of words and labels for text classification. *arXiv preprint arXiv:1805.04174*.

Zeerak Waseem. 2016. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science*, pages 138–142.

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

Yijin Xiong, Yukun Feng, Hao Wu, Hidetaka Kamigaito, and Manabu Okumura. 2021a. Fusing label embedding into bert: An efficient improvement for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1743–1750.

Yijin Xiong, Yukun Feng, Hao Wu, Hidetaka Kamigaito, and Manabu Okumura. 2021b. Fusing label embedding into BERT: An efficient improvement for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1743–1750, Online. Association for Computational Linguistics.

Han-Jia Ye and Wei-Lun Chao. 2021. How to train your maml to excel in few-shot classification. In *International Conference on Learning Representations*.

Wenjie Yin and Arkaitz Zubiaga. 2021. Towards generalisable hate speech detection: a review on obstacles and solutions. *PeerJ Computer Science*, 7:e598.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics (ACL).

Oana Ştefăniţă and Diana-Maria Buf. 2021. Hate speech in social media and its effects on the lgbt community: A review of the current research. *Romanian Journal of Communication and Public Relations*, 23(1):47–55.

A Appendix

Preprocessing

We perform standard preprocessing steps on our data. First, we remove non-ASCII characters from the inputs and convert them to lower case. Special platform-specific characters are removed, with certain exceptions (e.g. hyperlinks replaced with <*url*>, user-mentions with <*user*>, hashtags are segmented into separate tokens by using the Ekphrasis Python library [5]). We also replace repetitive patterns with a single representative (e.g. *"b b b "* to *"b"*).

Technological Details

All models are trained using single NVIDIA P100 GPU, with the exception on JE models, which were trained on NVIDIA A100 GPU. Our system also posses 20GB of RAM memory.

To select hyperparameters for the K-shot fine-tuning process on test domains, we use a sample of size K=64 from the left over data after the initial K-shot training samples. To select hyperparameters during meta-training, we monitor the average losses and F1-score during meta-testing. Meta-learning models are trained over a number of meta epochs, where each consists of 300 tasks randomly chosen from the 10 training domains. For fine-tuning, the learning rate is equipped with the Cosine Annealing Learning Rate scheduler [6] with minimum rate set to 1e-5. *Roberta_retrained* models are pre-trained using MLM objective for 5 epochs. Learning rates are chosen from the following pool of candidates: {1e-5, 2e-5, 5e-5, 7e-5, 1e-4, 5e-4, 7e-4, 1e-3}. Fine tuning and meta epochs are chosen from {2,3,4,5}. Batch sizes are set to 16. Table 4 shows the final values of hyperparameters.

[5] Available at https://github.com/cbaziotis/ekphrasis

[6] As implemented by Pytorch library

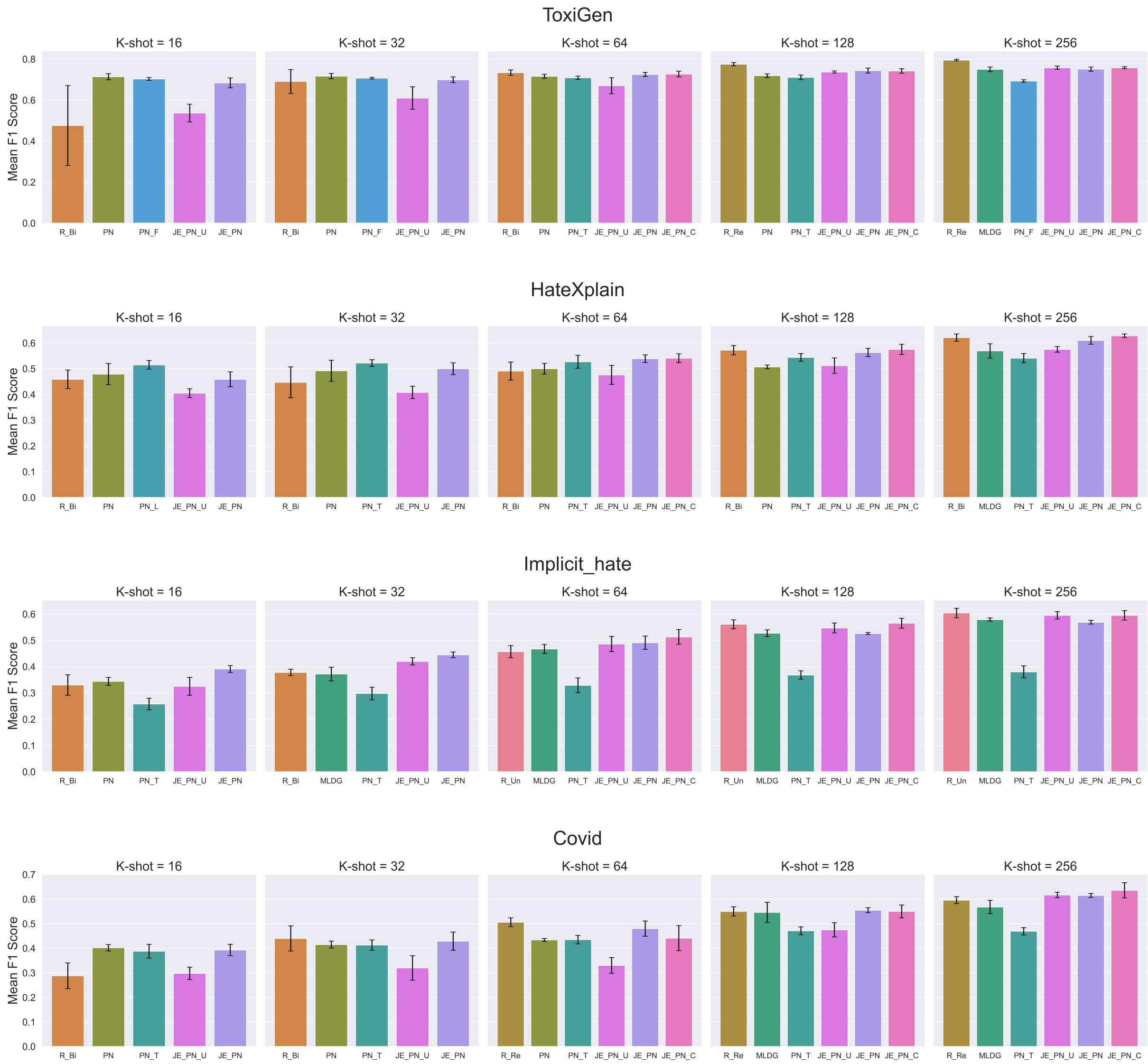


Figure 3: Mean macro F-1 scores of various models on 4 test sets at different K-shot settings, with error bars representing standard deviation over 5 seeds . For each K, the first bar shows the best performer among the *Baseline* models, the second bar shows the best among the models *Without Label*, and the third among models *With Label*. The rest includes all applicable *Joint-Embedding* models. *R_Bi*: RoBERTa_binary, *R_Re*: RoBERTa_retrained, *R_Un*: RoBERTa_untrained, *PN*: ProtoNet, *PN_F*: ProtoNET_Full, *PN_T*: ProtoNet_Token, *PN_L*: ProtoNet_Label, *PM*: ProtoMAML, *JE_PN*: JE_ProtoNet, *JE_PN_U*: JE_ProtoNet_Untrained, *JE_PN_C*: JE_ProtoNet_CLS.

Model	Meta Epoch	Meta Learning Rates	Finetune Epoch	Finetune Learning Rates
Baselines	-	-	3	2e-5
MLDG	5	E:5e-5; C:1e-4	3	E:2e-5 ; C:{-1:5e-3, 16:5e-3, 32:5e-3, 64:7e-3, 128:7e-3, 256:5e-4}
ProtoMAML	5	E:5e-5; C:1e-4	4	E:2e-5 ; C:{-1:1e-3, 16:1e-4, 32:1e-4}
ProtoNet (all variants)	5	E:2e-5; C:1e-4	2	E:1e-5
JE_ProtoNet	5	E:5e-5; A:2e-5; C:1e-4	3	E:2e-5, A:2e-5
JE_ProtoNet_CLS	-	-	3	E:2e-5, A:2e-5; C:1e-4

Table 4: Hyperparameter values chosen for reported runs. For learning rates, E stands for *Word Embedding*, (RoBERTa), A for *Attention* module, C for *Classification* head. If no letter specififed, then learning rate applies to all components. Learning rates in bracketed dictionaries are tied to the corresponding component, with the key represents the corresponding K-shot value it is applied to. -1 denotes the default rate.

Dataset	Definition
Waseem and Hovy, 2016	**Overtly Aggressive** : any text in which aggression is overtly expressed either through the use of specific kind of lexical items or lexical features which is considered aggressive and or certain syntactic structures is overt aggression. **Covertly Aggressive** : any speech in which aggression is overtly expressed either through the use of specific kind of lexical items or lexical features which is considered aggressive and or certain syntactic structures is overt aggression. **Non Aggressive** : any text that does not fall into the other two categories.
Golbeck et al., 2017	**Offensive** : uses a sexist or racial slur, attacks a minority, seeks to silence a minority, criticizes a minority without a well founded argument , promotes, but does not directly use, hate speech or violent crime, criticizes a minority and uses a straw man argument, blatantly misrepresents truth or seeks to distort views on a minority with unfounded claims, shows support of problematic hash tags, negatively stereotypes a minority, defends xenophobia or sexism, contains a screen name that is offensive, as per the previous criteria, the tweet is ambiguous at best , and the tweet is on a topic that satisfies any of the above criteria. **Not Offensive** : does not into any other categories.
Davidson et al., 2017	**Targeted Insult** : posts containing insult threat to an individual, a group, or others. **Untargeted Insult** : posts containing non targeted profanity and swearing. posts with general profanity are not targeted, but they contain non acceptable language. **Not Offensive** : posts that do not contain offense or profanity
Kumar et al., 2018	**Harrassment** : deeply racist, misogynistic or homophobic, or otherwise bigoted. the use of shocking language primarily to upset the person who is reading. unapologetically or intentionally offensive this could be someone saying something with the intent of upsetting a group, or an extreme account e.g. neo nazis using language that they approve of but they know the general public would disapprove of. have language intended to make the target or a broader group fearful or to feel unsafe. express hate or extreme bias to a particular group. could be based on religion, race, gender, sexual orientation. language directed at a particular person or group designed to upset them. this language may be milder than in other cases but should be part of the campaign by one person or a group to make the target feel threatened or intimidated. **Not Harrassment** : anything that does not rise to the level of clearly and unambiguously fitting into the other categories.
Founta et al., 2018	**Hate Speech** : targeting immigrants; content must have immigrants refugees as main target, or even a single individual, but considered for his her membership in that category and not for the individual characteristics ; must deal with a message that spreads, incites, promotes or justifies hatred or violence against target, or a message that aims at dehumanizing, hurting or intimidating the target. or expresses hating towards women in particular in the form of insulting, sexual harassment, threats of violence, stereotype, objectification and negation of male responsibility. **Not Hate Speech** : the followings are not considered hate speech, against other target, offensive language, blasphemy, historical denial, over incitement to terrorism, offense towards public servant, defamation.
Zampieri et al., 2019	**Abusive Language** : any strongly impolite, rude or hurtful language using profanity, that can show a debasement of someone or something, or show intense emotion. **Hate Speech** : language used to express hatred towards a targeted individual or group, or is intended to be derogatory, to humiliate, or to insult the members of the group, on the basis of attributes such as race, religion, ethnic origin, sexual orientation, disability, or gender. **Normal** : tweets that do not fall in any of the other categories
Basile et al., 2019	**Hate Speech** : language that is used to expresses hatred towards a targeted group or is intended to be derogatory, to humiliate,or to insult the members of the group. may also be language that threatens or incites violence. **Offensive Language** : may contain offensive terms but targets disadvantaged social groups in a manner that is potentially harmful to them. **Neither** : language that does not all into either of the other categories .
Sap et al., 2019b	**Offensive** : denotes the overall rudeness, disrespect, or toxicity of a post. whether a post could be considered offensive to anyone. **Not Offensive** : not offensive to anyone.
Toraman et al., 2022	**Hate** : target, incite violence against, threaten, or call for physical damage for an individual or a group of people because of some identifying trait or characteristic. **Offensive** : humiliate, taunt, discriminate, or insult an individual or a group of people in any form, including textual. **Normal** : does not fall into any of the other categories .
Vidgen et al., 2021	**Derogation** : content which explicitly attacks, demonizes, demeans or insults a group. **Animosity** : content which expresses abuse against a group in an implicit or subtle manner. *Threatening* : content which expresses intention to, support for, or encourages inflicting harm on a group, or identified members of the group. **Support For Hateful Entities** : content which explicitly glorifies, justifies or supports hateful actions, events, organizations, tropes and individuals collectively, entities. **Dehumanization** : content which perceives or treats people as less than human. **Not Hate** : content that falls into none of the other categories .

Table 5: Full definition for train datasets

Dataset	Definition
Covid	**Hostility Against An East Asian Entity** : express abuse or intense negativity against an east asian entity, primarily by derogating attacking them. also includes conspiracy theories, claiming east asians are a threat, and expressing negative emotions about them. **Criticism Of An East Asian Entity** : make a negative judgement assessment of an east asian entity, without being abusive. this includes commenting on perceived social, economic and political faults, including questioning their response to the pandemic and how they are governed. **Discussion Of East Asian Prejudice** : tweets that discuss prejudice related to east asians but do not engage in, or counter, that prejudice. includes content which discusses whether east asian prejudice has increased during covid 19, the supposed media focus on prejudice, and or free speech. also includes explicitly challenge or condemn abuse against an east asian entity. includes rejecting the premise of abuse, describing content as hateful or prejudicial, or expressing solidarity with target entities. **None Of The Above** : do not fall into any of the other categories. could be abusive in other ways, such as expressing misogyny.
ToxiGen	**Toxic** : text that could be seen as harmful to anyone, may contain offensive rude, humor, insults, personal attacks, profanity, aggression, may refer to targeted group with harmful intent that is expressed in stereotypes or lewd manners. minority groups include blacks, asians, native americans, muslims, latinos, jewish, chinese, lgbtq , mentally disabled, physically disabled, mexican, women and middle eastern. **Benign** : text that could not be seen as harmful or intended to be harmful to anyone.
Implicit_hate	**White Grievance** : includes frustration over a minority group s perceived privilege and casting majority groups as the real victims of racism. this language is linked to extremist behavior and support for violence. **Incitement To Violence** : includes flaunting in group unity and power or elevating known hate groups and ideologies. **Inferiority Language** : implies one group or individual is inferior to another, and it can include dehumanization denial of a person s humanity , and toxification language that compares the target with disease, insects, animals . related to assaults on human dignity, dominance, and declarations of superiority of the in group. **Irony** : refers to the use of sarcasm , humor, and satire to attack or demean a protected class or individual. **Stereotypes And Misinformation** : associate a protected class with negative attributes such as crime, or terrorism. includes misinformation that feeds stereotypes and vice versa, like holocaust denial and other forms of historical negationism. **Threatening And Intimidation** : conveys a speaker's commitment to a target s pain, injury, damage, loss, or violation of rights, threats related to implicit violation of rights and freedoms, removal of opportunities, and more subtle forms of intimidation.
HateXplain	**Hate Speech** : language which attacks, demeans, offends, threatens, or insults a group based on race, ethnic origin, religion, disability, gender, age, sexual orientation, or other traits. it is not the presence of certain words that makes the text hate speech, rather you should look the context the word is used in the text. **Offensive Language** : usage of rude, hurtful, derogatory, obscene or insulting language to upset or embarasse people. **Normal** : neither hate speech nor offensive .

Table 6: Full definition for test datasets

K	Model	ToxiGen F1	ToxiGen σ	HateXplain F1	HateXplain σ	Implicit_hate F1	Implicit_hate σ	Covid F1	Covid σ
16	**HateBERT**	0.462	0.095	0.326	0.032	0.233	0.047	0.368	0.037
	RoBERTa_untrained	0.377	0.075	0.210	0.090	0.232	0.052	0.184	0.108
	RoBERTa_binary	0.476	0.195	0.459	0.036	0.330	0.039	0.288	0.052
	RoBERTa_retrained	0.427	0.144	0.171	0.018	0.181	0.010	0.171	0.127
	ProtoNet	**0.714**	0.015	0.479	0.041	*0.344*	0.015	**0.402**	0.013
	ProtoMAML	0.343	0.020	0.180	0.016	0.086	0.025	0.201	0.002
	MLDG	0.557	0.112	0.293	0.053	0.263	0.049	0.314	0.040
	ProtoNet_Token	*0.695*	0.019	*0.502*	0.022	0.258	0.022	0.388	0.028
	ProtoNet_Label	0.668	0.049	**0.515**	0.017	0.221	0.021	0.364	0.014
	ProtoNet_Full	0.703	0.007	0.489	0.027	0.246	0.020	0.357	0.038
	JE_ProtoNet	0.684	0.024	0.459	0.029	**0.391**	0.013	*0.393*	0.023
	JE_ProtoNet_Untrained	0.537	0.043	0.405	0.017	0.325	0.034	0.298	0.025
	JE_ProtoNet_CLS	–	–	–	–	–	–	–	–
32	**HateBERT**	0.498	0.092	0.373	0.044	0.359	0.010	*0.429*	0.028
	RoBERTa_untrained	0.482	0.073	0.365	0.078	0.318	0.043	0.323	0.053
	RoBERTa_binary	0.691	0.058	0.447	0.060	0.378	0.012	**0.440**	0.051
	RoBERTa_retrained	0.574	0.134	0.192	0.069	0.231	0.049	0.273	0.138
	ProtoNet	0.717	0.013	0.492	0.041	0.369	0.023	0.415	0.014
	ProtoMAML	0.368	0.074	0.191	0.026	0.177	0.077	0.205	0.014
	MLDG	0.637	0.044	0.384	0.050	0.372	0.026	0.367	0.038
	ProtoNet_Token	*0.706*	0.009	**0.522**	0.013	0.298	0.024	0.413	0.021
	ProtoNet_Label	0.694	0.016	*0.517*	0.006	0.222	0.014	0.338	0.018
	ProtoNet_Full	**0.707**	0.004	0.498	0.028	0.251	0.019	0.382	0.011
	JE_ProtoNet	0.699	0.014	0.500	0.023	**0.445**	0.011	*0.429*	0.037
	JE_ProtoNet_Untrained	0.610	0.055	0.408	0.024	*0.420*	0.014	0.320	0.050
	JE_ProtoNet_CLS	–	–	–	–	–	–	–	–
64	**HateBERT**	0.558	0.109	0.466	0.017	0.420	0.011	0.455	0.031
	RoBERTa_untrained	0.498	0.148	0.422	0.084	0.457	0.023	0.430	0.061
	RoBERTa_binary	**0.734**	0.013	0.491	0.035	0.424	0.032	*0.494*	0.021
	RoBERTa_retrained	0.688	0.088	0.177	0.026	0.342	0.047	**0.506**	0.018
	ProtoNet	0.716	0.010	0.500	0.021	0.375	0.017	0.434	0.006
	ProtoMAML	0.442	0.156	0.318	0.078	0.263	0.083	0.202	0.019
	MLDG	0.680	0.018	0.393	0.063	0.467	0.017	0.385	0.041
	ProtoNet_Token	0.709	0.008	0.527	0.025	0.329	0.028	0.435	0.017
	ProtoNet_Label	0.669	0.045	0.500	0.049	0.196	0.019	0.310	0.027
	ProtoNet_Full	0.702	0.004	0.507	0.029	0.266	0.019	0.390	0.008
	JE_ProtoNet	*0.725*	0.010	*0.539*	0.015	*0.491*	0.025	0.480	0.031
	JE_ProtoNet_Untrained	0.670	0.039	0.476	0.037	0.486	0.029	0.330	0.032
	JE_ProtoNet_CLS	0.727	0.014	**0.541**	0.017	**0.513**	0.028	0.441	0.051
128	**HateBERT**	0.687	0.012	0.537	0.009	0.497	0.010	0.490	0.027
	RoBERTa_untrained	0.707	0.032	0.528	0.029	*0.561*	0.017	0.534	0.026
	RoBERTa_binary	0.744	0.020	**0.572**	0.018	0.535	0.013	0.539	0.019
	RoBERTa_retrained	**0.775**	0.008	0.332	0.073	0.515	0.030	*0.550*	0.019
	ProtoNet	0.719	0.009	0.507	0.007	0.379	0.011	0.445	0.018
	ProtoMAML	0.551	0.143	0.400	0.107	0.448	0.022	0.340	0.078
	MLDG	0.712	0.013	0.473	0.041	0.527	0.013	0.546	0.041
	ProtoNet_Token	0.711	0.011	0.544	0.015	0.368	0.016	0.471	0.016
	ProtoNet_Label	0.576	0.041	0.420	0.014	0.164	0.016	0.280	0.033
	ProtoNet_Full	0.702	0.010	0.519	0.016	0.247	0.014	0.410	0.015
	JE_ProtoNet	*0.744*	0.012	*0.563*	0.016	0.526	0.004	**0.555**	0.010
	JE_ProtoNet_Untrained	0.737	0.005	0.512	0.030	0.547	0.019	0.475	0.029
	JE_ProtoNet_CLS	0.742	0.011	0.575	0.020	**0.565**	0.019	*0.550*	0.026

Table 7: Macro F1-scores of models on 4 test domains with K=16 to 128. Best performance for each K per dataset in **bold**, second best *italicized*.