Towards Generalized Offensive Language Identification

Alphaeus Dmonte[1], Tejas Arya[2], Tharindu Ranasinghe[3], and Marcos Zampieri[1]

[1] George Mason University, USA
[2] Rochester Institute of Technology, USA
[3] Lancaster University, UK
admonte@gmu.edu

Abstract. The prevalence of offensive content on the internet, encompassing hate speech and cyberbullying, is a pervasive issue worldwide. Consequently, it has garnered significant attention from the machine learning (ML) and natural language processing (NLP) communities. As a result, numerous systems have been developed to automatically identify potentially harmful content and to mitigate its impact. These systems can follow two approaches; (i) Use publicly available models and application endpoints, including prompting large language models (LLMs) (ii) Annotate datasets and train ML models on them. However, both approaches lack an understanding of how generalizable they are. Furthermore, the applicability of these systems is often questioned in off-domain and practical environments. This paper empirically evaluates the generalizability of offensive language detection models and datasets across a novel generalized benchmark: *GenOffense*. We answer three research questions on generalizability. Our findings will be useful in creating robust real-world offensive language detection systems.

Keywords: Offensive Language · Large Language Models · Generalizability.

1 Introduction

The presence of offensive posts on social media platforms leads to various negative consequences for users. Offensive posts have been linked to harmful outcomes such as increased suicide attempts [19, 27] and mental health issues such as depression [3, 8]. To address these serious repercussions, content moderation is typically employed on online platforms. Given the overwhelming volume of posts, however, human moderators alone cannot handle the task effectively, necessitating the development of automatic systems to assist them [41, 51, 48].

A highly effective method for constructing systems that can detect offensive language involves using publicly accessible application endpoints and models in an *unsupervised* fashion. Notably, the development of openly accessible services such as perspective API [24] and models such as toxicBERT have greatly facilitated this approach. Furthermore, a more recent development involves the use of LLMs

in a similar manner, employing specific prompts to identify offensive language [20]. The other most common method for offensive language identification is the *supervised* approach, where a dataset is annotated to serve as training material for ML systems. The datasets can be annotated with different goals in mind depending on the sub-task they address, such as aggression, cyberbullying and hate speech [43] as well as following a more general taxonomy [46].

While both the *unsupervised* and *supervised* approaches have provided excellent results in specific offensive language detection use cases, their generalizability [16, 2] and the ability to perform in unseen use cases [38, 1, 44] have often been questioned. The ability to effectively generalize is consistently highlighted as a fundamental requirement for NLP models [26, 14]. Particularly in a real-world application such as offensive language detection, generalization is crucial to ensure that the system exhibits robust, reliable, and fair behavior when making predictions on data that differs from their training data. However, to the best of our knowledge, no comprehensive evaluation of the generalizability of offensive language detection systems and datasets has been yet carried out. To fill this important gap in the literature, in this paper, we address the question of generalizability in offensive language identification.

Following [21], we define generalizability as the ability to perform consistently among different datasets. First, we construct a generalized offensive language detection benchmark; ***GenOffense***, collecting eight datasets extracted from different social media platforms and mapping them to a general offensive language detection taxonomy. We evaluate publicly available APIs and models, including LLMs in ***GenOffense***, and discuss the results. In the second part, we train various ML models on the training sets of these eight different datasets under different settings such as fully supervised, few-shot and zero-shot and evaluate the results. We answer three research questions as follows:

- **RQ1 - Generalizability:** How well do the publicly available systems and the models trained on different datasets generalize?
- **RQ2 - Dataset Size:** What is the impact of dataset size on generalizability? Does more data always result in better generalizability?
- **RQ3 - Domain Specificity:** What is the overlap and performance carryover between datasets collected from different platforms?

2 Related Work

Offensive Language Detection The problem of offensive language on social media has gained a lot of attention within the ML/NLP community. Researchers and organizations have developed systems to identify multiple types of offensive content such as *aggression*, *cyberbullying*, and *hate speech* [12, 34]. Perspective API [24] is one such free API that was trained on the Toxic Comment Classification dataset [7]. More recently, with the rise of LLMs such as GPT, researchers have used LLMs to detect and identify various forms of offensive language [50]. [20] utilized

ChatGPT for hateful speech detection and showed that ChatGPT provides satisfactory results for certain prompts. In a different study, [25] investigated the potential of using ChatGPT for annotating offensive comments and compared its results with those from crowdsourcing workers and the results show a high agreement. All these systems and APIs can be used in an *unsupervised* way to detect offensive content. However, these systems can induce bias to the task depending on the data they were used to train.

As discussed in the introduction, the most common approach to detect offensive content is the *supervised* approach, where the ML models are trained on annotated datasets. For this purpose, several datasets have been created for English [46, 12, 30]. The popular shared tasks such as OffensEval [47, 49], HatEval [4] and HASOC [39] have also contributed to creating some of these popular English datasets. Researchers have trained various ML models ranging from SVMs [28] to neural transformers [35]. Recent studies have also fine-tuned transformer models on offensive language data and released domain-specific models such as HateBERT [9] and fBERT [37]. These supervised models have provided excellent results over several datasets.

Generalized Machine Learning Good generalization, defined as the ability to successfully transfer representations, knowledge, and strategies from past experiences to new experiences, is a primary requisite for NLP/ ML models [21]. Generalization has been widely investigated on different NLP tasks, including machine translation [31], language modeling [10], and semantic parsing [22] and is crucial to ensure robustness, reliability, and fairness [40]. While the aforementioned offensive language detection methods have provided good results on the datasets they are evaluated, several studies have questioned their ability to perform on unseen use cases. [38] showed that hate speech classifiers often misclassify chess discussions as racist. [44] evaluate nine different offensive language detectors on political discussions and show that they have a low agreement. Furthermore, offensive language detection systems have been evaluated for geographic biases [17] and vulnerability to adversarial attacks [18]. Finally, [16] tested multiple intra- and cross-dataset offensive language identification scenarios. However, the study is limited to a few datasets and models. To the best of our knowledge, no work exists on a comprehensive evaluation of the generalizability of offensive language detection systems, which we address in this research.

3 GenOffense: A Generalized Offensive Language Detection Benchmark

The root cause for the lack of generalization research on offensive language detection is that no standard benchmark exists for the domain. While there are several popular datasets for offensive language identification, each of them has been annotated using different annotation guidelines and taxonomies. This, in theory, limits the possibility of combining existing datasets when training and evaluating robust offensive language identification models. To address this

we construct the first Generalized Offensive Language Detection Benchmark; *GenOffense*.[4]

	Training		Testing			
Dataset	**Inst.**	**OFF %**	**Inst.**	**OFF %**	**Data Sources**	**Reference**
AHSD	19,822	0.83	4,956	0.82	Twitter	[12]
HASOC	5,604	0.36	1,401	0.35	Twitter, Facebook	[29]
HatE	9,000	0.42	1,434	0.42	Twitter	[4]
HateX	11,535	0.59	3,844	0.58	Twitter, Gab	[30]
OHS	8,285	0.21	2,090	0.20	Reddit	[32]
OLID	13,240	0.33	860	0.27	Twitter	[46]
TCC	12,000	0.09	2,500	0.10	Wikipedia Talk	URL[1]
TRAC	4,263	0.20	1,200	0.42	Facebook, Twitter, YouTube	[5]

Table 1: The eight datasets used for *GenOffense*, including the number of instances (Inst.) in the training and testing sets, the OFF % in each set, the data source, and the reference.

3.1 *GenOffense* Construction

We use eight popular publicly available datasets containing English data summarized in Table 1 to construct *GenOffense*. As the datasets were annotated using different guidelines and labels, following the methodology described in [33], we map all labels to OLID level A [46], which is offensive (OFF) and not offensive (NOT). We choose OLID due to the flexibility provided by its general three-level hierarchical taxonomy below, where the OFF class contains all types of offensive content, from general profanity to hate speech, while the NOT class contains non-offensive examples.

- **Level A:** Offensive (OFF) vs. Non-offensive (NOT).
- **Level B:** Classification of the type of offensive (OFF) tweet - Targeted (TIN) vs. Untargeted (UNT).
- **Level C:** Classification of the target of a targeted (TIN) tweet - Individual (IND) vs. Group (GRP) vs. Other (OTH).

In the OLID taxonomy, offensive (OFF) posts targeted (TIN) at an individual are often cyberbullying, whereas offensive (OFF) posts targeted (TIN) at a group are often hate speech.

AHSD is one of the most popular hate speech datasets available. The dataset contains data retrieved from Twitter, which was annotated using crowdsourcing. The annotation taxonomy contains three classes: Offensive, Hate, and Neither. We conflate Offensive and Hate under a class OFF while neither class corresponds to OLID's NOT class.

[4] `https://github.com/TharinduDR/GeneralOffense.git`

HASOC is the dataset used in the HASOC shared task 2020. It contains posts retrieved from Twitter and Facebook. The upper level of the annotation taxonomy used in HASOC is hate-offensive vs non Non hate-offensive, which is the same as OLID's. This allows us to directly map hate-offensive to OLID's OFF class and non hate-offensive to NOT class.

HatE is the official dataset at SemEval-2019 Task 5 (HatEval), which focuses on hate speech against migrants and women. The first level of annotation contains two classes, hate speech or not, which can be mapped directly to OLID's OFF and NOT categories.

HateX is a dataset collected for the explainability of hate speech. It contains both token- and post-level annotation of Twitter and Gab posts. Post-level annotations have three classes: Hateful, Offensive, and Normal. We map Hateful and Offensive classes to OFF class and Normal to NOT class.

OHS is a dataset collected from Reddit with the goal of studying interventions in conversations containing hate speech. Full conversations/threads have been retrieved and annotated at the post-level as hateful or not hateful, which we map to OFF and NOT classes correspondingly.

OLID is the official dataset of the SemEval-2019 Task 6 (OffensEval) [47]. It contains data from Twitter annotated with a three-level hierarchical annotation which we described before. We adopt the labels in OLID level A as our classification labels.

TCC is the Toxic Comment Classification dataset. TCC was created for the Kaggle competition with the same name. The dataset contains Wikipedia comments with various classes such as toxic, obscene, insult, and threat merged in the OLID OFF class. The rest of the instances were mapped to the NOT class.

TRAC is the dataset used in the TRAC shared task 2020 [23]. It focuses on aggression detection with three classes: overtly aggressive and covertly aggressive merged as OFF and non-aggressive which corresponds to the NOT class used in OLID. Finally, TRAC is the most heterogeneous dataset we used in terms of data sources containing posts from Facebook, Twitter, and YouTube.

3.2 *GenOffense* Properties

We highlight the following generalization types that *GenOffense* benchmark tests. These are shown as crucial generalization types by [21].

Platform Shift *GenOffense* benchmarks contains datasets from six different social media platforms. While most of the datasets are based on Twitter, *GenOffense* has datasets that are based on other social media platforms such as Facebook and Reddit. Therefore, *GenOffense* benchmark evaluates how the models can handle different platforms.

Language Shift The datasets included in *GenOffense* range from 2017 to 2021. The language that was used to convey offense can be different from 2017 to 2021. Therefore, *GenOffense* benchmark tests how the models can handle language shift.

Task Shift As we mentioned before, these datasets contained different tasks such as aggression detection, hate speech detection and offensive language detection. As a result, *GenOffense* reflects these tasks and a model that can perform well in *GenOffense* will generalize well across different sub-tasks.

Topic Shift Different datasets have been collected with different goals in mind depending on the 'offensive language detection sub-task' they address. Therefore, each dataset in *GenOffense* has different topics, and the models will be evaluated on how well they can handle different topics in the offensive language domain.

Finally, upon acceptance of this paper, *GenOffense* will be made available as an online platform where researchers can submit the model predictions and evaluate how the model generalizes over different datasets.

4 Unsupervised Offensive Language Detection Models

The following public models and APIs are evaluated in the test sets of *GenOffense* without any training or fine-tuning.

	Models	AHSD	HASOC	HatE	HateX	OHS	OLID	TCC	TRAC	Avg
I	Perspective	0.8603	0.6487	0.5340	0.6688	0.5578	0.7691	0.9228	0.6847	0.7058
	ToxicBERT	0.7430	0.6522	0.5283	0.6361	0.5416	0.7765	0.9606	0.6906	0.6911
II	BERT	0.1473	0.3951	0.4002	0.2986	0.4456	0.4328	0.3741	0.3961	0.3612
	fBERT	0.4589	0.3149	0.4075	0.3807	0.2403	0.3357	0.4178	0.3230	0.3599
	HateBERT	0.5335	0.4733	0.4968	0.5405	0.4466	0.4984	0.5945	0.3467	0.4913
III	Davinci-003	0.8152	0.5909	0.4881	0.6075	0.4780	0.7401	0.7617	0.7454	0.6534
	Falcon-7B	0.7406	0.6049	0.6033	0.6106	0.5291	0.7456	0.6178	0.7152	0.6458
	T0	0.6972	0.5005	0.4195	0.5631	0.5160	0.4907	0.6008	0.7126	0.5625
	MPT-7B	0.5313	0.3571	0.3621	0.5240	0.2832	0.3703	0.2998	0.7466	0.4343

Table 2: Macro F1 score of the publicly available offensive language detection models. **Row I** shows public APIs/ models, **row II** shows the results of adapting transformers and **row III** shows the results for LLMs. The **Average** column shows the average score of all the experiments.

4.1 Methods

Public APIs/ Models We evaluate **Perspective API** [24] and **ToxicBERT** [11][5]. **Perspective API** is a free API developed by Google Jigsaw, that leverages machine learning to identify toxic comments. This API was first trained using a BERT [13] model, which is then distilled into monolingual CNN based models. The model was mainly trained on the TCC dataset, which we also included in *GenOffense*. The model has six attributes, toxicity, severe toxicity, identity attack, insult, profanity, and threat. The model generates a score between 0 and 1 for each of these attributes. For each test dataset, we get all the attribute scores for each instance. If any of the attributes have a value greater than 0.5, we classify that instance as OFF, else it is classified as NOT.

We also evaluate **ToxicBERT** on *GenOffense*. ToxicBERT is a BERT model trained primarily on the TCC dataset. The model is a multi-label classification model with six labels similar to Perspective API. We follow a similar approach to Perspective API to convert the ToxicBERT outputs into OFF and NOT classes.

Adapting Transformers We evaluate different general-purpose transformer models; BERT, and two domain-specific transformer models; fBERT [37] and HateBERT [9] on offensive language identification using an unsupervised approach. We classify a test sentence as positive or negative, where the positive label represents the NOT class and the negative represents the OFF class. We concatenate the last four hidden states returned by the model as the representative embeddings for the test sentence and the labels. We then find the cosine similarity between the representative embeddings of the labels and that of the test sentence. Finally, the sentence is assigned the label with the highest cosine similarity score.

Prompting LLMs Finally, we evaluate how LLMs perform in *GenOffense* benchmark, a recent trend as we discussed before. We use the following prompt to get a response from LLMs.

> *Comments containing any form of non-acceptable language (profanity) or a targeted offense, which can be veiled or direct, are offensive comments. This includes insults, threats, and posts containing profane language or swear words. Comments that do not contain offense or profanity are not offensive. Is this comment offensive or not? Comment:*

We use several LLMs for prompting. We first use Davinci-003 through OpenAI API. Additionally, we use MPT-7B-Instruct, Falcon-7B-Instruct and T0-3B [36]. All of these models are available in HuggingFace[6] [45], and we use the LangChain implementation.

[5] ToxicBERT is available at `https://huggingface.co/unitary/toxic-bert`

[6] MPT-7B-Instruct is available at `https://huggingface.co/mosaicml/mpt-7b-instruct`, Falcon-7B-Instruct is available at `https://huggingface.co/tiiuae/falcon-7b-instruct` and T0-3B is available at `https://huggingface.co/bigscience/T0_3B`

4.2 Results

The results of the aforementioned models are shown in Table 2. Public APIs/models generally performed well on *GenOffense* compared to the other two methods. However, LLMs also provide competitive results. From the LLMs, Davinci-003 performs best, closely followed by Falcon-7B. It is clear that recent LLMs produce better results on *GenOffense*. Overall, Perspective API performed best on the *GenOffense* benchmark. It provided the best results for six datasets out of eight and had the highest overall average.

Most of the models show inconsistent results on the datasets. Particularly, all the models do not perform well on HatE and OHS datasets which indicates that these models do not generalize well across different tasks and platforms.

5 Training Offensive Language Detection Models

In this section, we evaluate the *supervised* ML models on *GenOffense* benchmarks. We train the following ML models under different settings on the training sets in *GenOffense* benchmark and evaluate on the test sets.

LSTM We experiment with a bidirectional Long Short-Term-Memory (BiLSTM) model, which we adapted from the baseline in OffensEval 2019 [47]. The model consists of *(i)* an input embedding layer with fasttext embedding [6], *(ii)* a bidirectional LSTM layer, and *(iii)* an average pooling layer of input features. The concatenation of the LSTM layer and the average pooling layer is further passed through a dense layer, whose output is ultimately passed through a *softmax* to produce the final prediction. We used updatable embeddings learned by the model during training as the input.

Transformers We also use transformers as a classification model, which have achieved state-of-the-art on a variety of offensive language identification tasks. From an input sentence, transformers compute a feature vector $\boldsymbol{h} \in \mathbb{R}^d$, upon which we build a classifier for the task. For this task, we implemented a softmax layer, i.e., the predicted probabilities are $\boldsymbol{y}^{(B)} = \text{softmax}(W\boldsymbol{h})$, where $W \in \mathbb{R}^{k \times d}$ is the softmax weight matrix and k is the number of labels. For the experiments, we use the bert-large-cased and domain-specific fBERT [37] and HateBERT [9] available in HuggingFace [45].

5.1 Model Configuration

For LSTM, we used a Nvidia Tesla k80 to train the models. We divided the dataset into a training set and a validation set using 0.8:0.2 split. We performed *early stopping* if the validation loss did not improve over 10 evaluation steps. For the LSTM model we used the same set of configurations mentioned in Table 3 in all the experiments. All the experiments were conducted for three times and the mean value is taken as the final reported result.

Parameter	Value
batch size	64
epochs	3
first dense layer units	256
learning rate	1e-4
LSTM units	64
max seq. length	256

Table 3: LSTM Parameter Specifications.

For transformers models, we used a GeForce RTX 3090 GPU to train the models. We divided the dataset into a training set and a validation set using a 0.8:0.2 split. For transformer models, we used the same set of configurations mentioned in Table 4 in all the experiments. We performed *early stopping* if the validation loss did not improve over 10 evaluation steps. All the experiments were conducted three times and the mean value is taken as the final reported result.

Parameter	Value
adam epsilon	1e-8
batch size	64
epochs	3
learning rate	1e-5
warmup ratio	0.1
warmup steps	0
max grad norm	1.0
max seq. length	256
gradient accumulation steps	1

Table 4: BERT Parameter Specifications.

5.2 Results

We use multiple strategies to answer the three **RQ**s considering generalizability with respect to training and testing data.

We address training set variation by training the three models in the following settings:

1 to 1 We train a separate machine learning model on each of the eight training sets. We then evaluate the trained model on each of the eight test sets in isolation.

All -1 We concatenate all training sets except one and train a single machine learning model. We then evaluate the model on the test set of that particular dataset that was left out.

	Train Dataset(s)	AHSD	HASOC	HatE	HateX	OHS	OLID	TCC	TRAC	Avg
LSTM	AHSD	0.8872	0.5465	0.3735	0.4903	0.3757	0.4005	0.4598	0.5809	0.5143
	HASOC	0.4336	0.6539	0.5388	0.5339	0.5503	0.5832	0.5756	0.4056	0.5343
	HatEval	0.6605	0.5200	0.5825	0.5266	0.5479	0.5413	0.5212	0.4991	0.5498
	HateX	0.5531	0.4623	0.3976	0.7091	0.4943	0.5193	0.3710	0.4710	0.4927
	OHS	0.1487	0.3936	0.5234	0.5309	0.6984	0.4670	0.2604	0.8117	0.4793
	OLID	0.6391	0.6224	0.5283	0.5477	0.5636	0.7124	0.6366	0.7473	0.6247
	TCC	0.5432	0.4756	0.5581	0.5497	0.5841	0.5711	0.7930	0.5587	0.5791
	TRAC	0.1800	0.4058	0.5105	0.4868	0.5376	0.5457	0.5460	0.6853	0.4872
	All	0.8689	0.6134	0.4849	0.6775	0.6236	0.6754	0.6537	0.7490	0.6681
	All-1	0.8675	0.5745	0.4539	0.5842	0.4957	0.5569	0.6491	0.6231	0.6006
BERT	AHSD	0.9268	0.6300	0.5279	0.5867	0.5179	0.6991	0.8188	0.6278	0.6657
	HASOC	0.6203	0.7585	0.5850	0.5550	0.5798	0.4925	0.6541	0.5495	0.5993
	HatEval	0.6122	0.4418	0.5880	0.4966	0.5795	0.5795	0.6240	0.6884	0.6012
	HateX	0.5690	0.6049	0.6322	0.7829	0.6167	0.5049	0.7214	0.5382	0.6212
	OHS	0.1960	0.4100	0.4567	0.3875	0.7745	0.4225	0.5048	0.3920	0.4430
	OLID	0.6857	0.6366	0.5296	0.6206	0.5725	0.8074	0.8451	0.7402	0.6797
	TCC	0.7210	0.6448	0.5241	0.6297	0.5677	0.7453	0.8805	0.6678	0.6726
	TRAC	0.6225	0.6260	0.5757	0.6122	0.5579	0.6916	0.7692	0.8596	0.6643
	All	0.9257	0.7506	0.7412	0.7718	0.7263	0.7449	0.8578	0.7793	0.7872
	All-1	0.3805	0.5346	0.5557	0.5771	0.5652	0.6680	0.7829	0.6529	0.5896
HateBERT	AHSD	0.9299	0.6248	0.5367	0.6051	0.5311	0.6425	0.7313	0.5813	0.6478
	HASOC	0.5704	0.6529	0.5852	0.5873	0.5563	0.6666	0.6101	0.7247	0.6192
	HatEval	0.7033	0.4974	0.4748	0.5852	0.5392	0.4814	0.6336	0.5421	0.5571
	HateX	0.5276	0.5954	0.5765	0.7724	0.5805	0.4981	0.6547	0.5609	0.5958
	OHS	0.2149	0.4024	0.3785	0.3102	0.7591	0.4189	0.4982	0.3651	0.4184
	OLID	0.7610	0.6239	0.5465	0.5971	0.4855	0.7811	0.7822	0.6317	0.6511
	TCC	0.7885	0.6286	0.5376	0.6386	0.5502	0.7107	0.8408	0.6493	0.6680
	TRAC	0.2597	0.5083	0.5164	0.5614	0.5715	0.5838	0.6392	0.8239	0.5580
	All	0.9174	0.6180	0.6076	0.7803	0.7009	0.7278	0.7955	0.6789	0.7283
	All-1	0.4844	0.5487	0.5760	0.6073	0.5751	0.5257	0.7861	0.6625	0.5957
fBERT	AHSD	0.9241	0.6365	0.5318	0.6246	0.5096	0.6918	0.8032	0.5482	0.6587
	HASOC	0.6912	0.6753	0.5386	0.6343	0.5510	0.7778	0.8226	0.7443	0.6794
	HatEval	0.6810	0.5332	0.4917	0.5724	0.5693	0.5599	0.6893	0.6714	0.5960
	HateX	0.5276	0.5954	0.6263	0.7840	0.5784	0.5252	0.7156	0.5991	0.6189
	OHS	0.1615	0.3935	0.4649	0.5720	0.7558	0.5572	0.6905	0.5382	0.5167
	OLID	0.7239	0.6572	0.5474	0.6217	0.5217	0.7838	0.8234	0.7524	0.6789
	TCC	0.7497	0.6545	0.5243	0.6303	0.5452	0.7458	0.8486	0.6753	0.6717
	TRAC	0.5757	0.5975	0.5565	0.6277	0.5389	0.7049	0.8290	0.8416	0.6589
	All	0.9201	0.6338	0.5727	0.7768	0.7102	0.7350	0.8389	0.7677	0.7444
	All-1	0.3516	0.5590	0.5588	0.6369	0.5685	0.5854	0.7889	0.6601	0.5887

Table 5: Macro F1 score of the offensive language detection models. The **Training Dataset(s)** shows the training dataset while the subsequent columns show the results for each test set. The **Average** column shows the average score of all the experiments.

All We concatenate the training sets of all the datasets and trained a single machine learning model. We then evaluate the model on each testing set of all eight datasets in *GenOffense*.

Few to 1 We also perform progress tests. We randomly selected 1000, 2000, 3000 etc. instances from each of the eight training sets and train separate machine learning models. We then evaluate the trained model on each of the eight test sets in isolation.

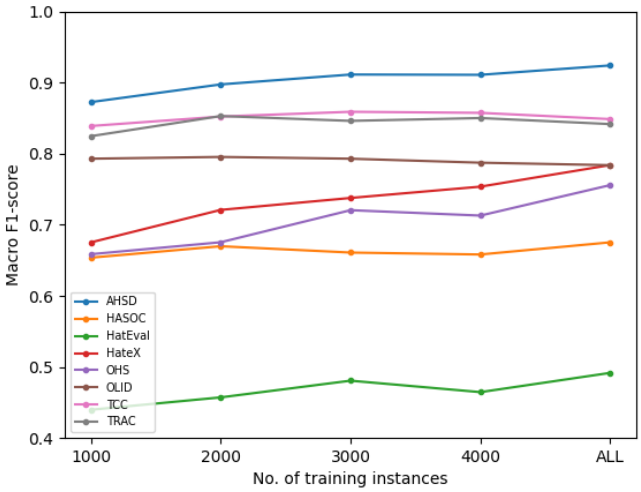

Fig. 1: Few-shot Learning Results for BERT

We present the results of the aforementioned strategies in Table 5 and Figure 1 in terms of Macro F1 score. The transformer models outperform the LSTM for all tested dataset combinations. This is in line with the findings of popular competitions such as HatEval and OffensEval. However, domain-specific models such as fBERT and HateBERT did not outperform BERT in average scores of *GenOffense*. This can be because both of these models are fine-tuned on platform-specific data. Unsurprisingly the **all** strategy, achieves the best results in all four classifiers. However, few-shot results in Figure 1 suggest and more training instances do not improve the average Macro F1 score of *GenOffense*. Furthermore, the **all -1** strategy was outperformed by many of the individual datasets suggesting that simply using a large dataset does not always result in better generalizability.

In terms of the individual dataset performance, models trained on OLID yielded the highest generalization followed by TCC. This is due to the general nature of these two datasets covering multiple types of offensive content rather than focusing on a particular type of offensive content (e.g. hate speech). AHSD also provided good generalization, likely due to the presence of both hate speech

and general offensive language in the dataset. On the other hand, models trained on OHS yielded the worst performance. This can be explained by the platform-specificity of the dataset, as OHS is the only Reddit dataset in this collection.

5.3 Test Set Combination

We also look at the performance of the models on a single test set combining all individual test sets in *GenOffense*. We use a separate BERT model on each of the eight training sets and tested them on the concatenated test set. We present the results obtained on the consolidated test set in terms of Macro and Weighted F1 Table 6.

Train Dataset	Macro F1	Weighted F1
AHSD	0.7348	0.7348
HASOC	0.6722	0.6743
HatE	0.6210	0.6239
HateX	0.6879	0.6899
OHS	0.4247	0.4348
OLID	**0.7543**	**0.7551**
TCC	0.7064	0.7060
TRAC	0.6467	0.6492

Table 6: BERT results for the combined test set in terms of Macro F1 and Weighted F1. Best results in bold.

The results indicate that models trained on OLID offers the best performance on the combined test set, followed by AHSD, and TCC while OHS delivers the lowest performance by a very large margin. This is in line with the results obtained using individual test sets.

6 Conclusion

This paper introduced the first generalization benchmark for offensive language detection; *GenOffense*. We also presented a comprehensive evaluation of the generalizability of different computational models, including recently released LLMs. We hope that our findings motivate the community to further explore the question of generalizability as argued by other recent studies [15, 16].

We revisit the research questions posed in the introduction:

– **RQ1 - Generalizability:** Despite being popular, LLMs did not perform well in the *GenOffense* benchmark. APIs, such as Perspective, showed better generalizability. In the supervised setting, models trained on OLID, AHSD,

and TCC provided the best generalizability to other datasets. This can be explained by their focus on general offensive language (in the case of OLID and TCC) and the presence of both hate speech and general offensive (in the case of AHSD), which is reflected in their annotation models. More specific datasets, such as HatEval, which focuses on women and migrants, displayed lower results. Finally, OHS, the only Reddit dataset, achieved the lowest performance, suggesting that the domain has a substantial impact on performance (see **RQ3**).

- **RQ2 - Dataset Size:** We observed that more data does not always result in better generalizability. The few-shot experiments showed that adding more training instances did not provide better generalizability. Even though the "All" strategy achieved the best performance for all datasets, the "All-1" strategy achieved performance lower than most datasets in isolation. Therefore we have not found a direct correlation between generalizability and training dataset size in our experiments. The question of dataset size requires further investigation.
- **RQ3 - Domain Specificity:** Models trained on OHS, the only Reddit dataset in the collection, achieved the lowest performance of all datasets, suggesting that the domain plays an important role in generalizability. OHS is not the smallest dataset tested in our experiments, therefore we believe that the low performance is due to the specificity of their source material (Reddit) rather than its size. We would like to further investigate this by running more dataset ablation experiments.

In future work, we would like to extend *GenOffense* benchmark to adversarial test sets using popular augmentation techniques such as random insertion and random deletion. This will provide the opportunity for the researchers to explore probing in offensive language detection models. We believe this would provide us with even more insights into the generalizability of the datasets and the robustness of the models. Finally, we would like to extend *GenOffense* to support multilingual offensive language datasets and replicate these experiments for different languages. Such multilingual benchmarks will be useful for many real-world applications.

Acknowledgements

We would like to thank the anonymous reviewers for their positive and valuable feedback. We further thank the creators of the datasets used in this paper for making the datasets publicly available for our research.

The experiments in this paper were conducted on the High End Computing (HEC) Cluster at Lancaster University, which is funded through a combination of central funding and contributions from individual research grants. The experiments were designed in UCREL-HEX[42], which is a collection of GPU equipped hosts at the School of Computing and Communications, Lancaster University.

Marcos Zampieri is partially supported by a grant from the Virginia Commonwealth Cyber Initiative (CCI) award number N-4Q24-009 .

References

1. Aggarwal, P., Chawla, P., Das, M., Saha, P., Mathew, B., Zesch, T., Mukherjee, A.: HateProof: Are hateful meme detection systems really robust? In: Proceedings of TheWebConf (2023)
2. Arango, A., Pérez, J., Poblete, B.: Hate speech detection is not as easy as you may think: A closer look at model validation. In: Proceedings of SIGIR. pp. 45–54 (2019)
3. Bannink, R., Broeren, S., van de Looij-Jansen, P.M., de Waart, F.G., Raat, H.: Cyber and Traditional Bullying Victimization as a Risk Factor for Mental Health Problems and Suicidal Ideation in Adolescents. PloS one **9**(4) (2014)
4. Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Pardo, F.M.R., Rosso, P., Sanguinetti, M.: Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In: Proceedings of SemEval (2019)
5. Bhattacharya, S., Singh, S., Kumar, R., Bansal, A., Bhagat, A., Dawer, Y., Lahiri, B., Ojha, A.K.: Developing a multilingual annotated corpus of misogyny and aggression. In: Proceedings of TRAC (2020)
6. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
7. Borkan, D., Dixon, L., Sorensen, J., Thain, N., Vasserman, L.: Nuanced metrics for measuring unintended bias with real data for text classification. In: Companion Proceedings of WWW. p. 491–500 (2019)
8. Bucur, A.M., Zampieri, M., Dinu, L.P.: An exploratory analysis of the relation between offensive language and mental health. In: Findings of the ACL (2021)
9. Caselli, T., Basile, V., Mitrović, J., Granitzer, M.: Hatebert: Retraining bert for abusive language detection in english. arXiv preprint arXiv:2010.12472 (2020)
10. Chronopoulou, A., Peters, M., Dodge, J.: Efficient hierarchical domain adaptation for pretrained language models. In: Proceedings of NAACL (2022)
11. Davidson, T., Bhattacharya, D., Weber, I.: Racial bias in hate speech and abusive language detection datasets. In: Proceedings of ALW (2019)
12. Davidson, T., Warmsley, D., Macy, M.W., Weber, I.: Automated hate speech detection and the problem of offensive language. In: Proceedings of ICWSM (2017)
13. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of NAACL (2019)
14. Elangovan, A., He, J., Verspoor, K.: Memorization vs. generalization : Quantifying data leakage in NLP performance evaluation. In: Proceedings of EACL (2021)
15. Fortuna, P., Soler, J., Wanner, L.: Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In: Proceedings of the 12th language resources and evaluation conference. pp. 6786–6794 (2020)
16. Fortuna, P., Soler-Company, J., Wanner, L.: How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? Information Processing & Management **58**(3), 102524 (2021)
17. Ghosh, S., Baker, D., Jurgens, D., Prabhakaran, V.: Detecting cross-geographic biases in toxicity modeling on social media. In: Proceedings of W-NUT (2021)
18. Gröndahl, T., Pajola, L., Juuti, M., Conti, M., Asokan, N.: All you need is "love": Evading hate speech detection. In: Proceedings of AISec (2018)
19. Hamm, M.P., Newton, A.S., Chisholm, A., Shulhan, J., Milne, A., Sundar, P., Ennis, H., Scott, S.D., Hartling, L.: Prevalence and Effect of Cyberbullying on Children

and Young People: A Scoping Review of Social Media Studies. JAMA Pediatrics **169**(8), 770–777 (08 2015)
20. Huang, F., Kwak, H., An, J.: Is chatgpt better than human annotators? potential and limitations of chatgpt in explaining implicit hate speech. In: Proceedings of WWW (2023)
21. Hupkes, D., Giulianelli, M., Dankers, V., Artetxe, M., Elazar, Y., Pimentel, T., Christodoulopoulos, C., Lasri, K., Saphra, N., Sinclair, A., et al.: State-of-the-art generalisation research in nlp: a taxonomy and review. arXiv preprint arXiv:2210.03050 (2022)
22. Jambor, D., Bahdanau, D.: LAGr: Label aligned graphs for better systematic generalization in semantic parsing. In: Proceedings of ACL) (2022)
23. Kumar, R., Ojha, A.K., Malmasi, S., Zampieri, M.: Evaluating aggression identification in social media. In: Proceedings of TRAC (2020)
24. Lees, A., Tran, V.Q., Tay, Y., Sorensen, J., Gupta, J., Metzler, D., Vasserman, L.: A new generation of perspective api: Efficient multilingual character-level transformers. In: Proceedings of KDD (2022)
25. Li, L., Fan, L., Atreja, S., Hemphill, L.: " hot" chatgpt: The promise of chatgpt in detecting and discriminating hateful, offensive, and toxic comments on social media. arXiv preprint arXiv:2304.10619 (2023)
26. Linzen, T.: How can we accelerate progress towards human-like linguistic generalization? In: Proceedings ACL (2020)
27. López-Meneses, E., Vázquez-Cano, E., González-Zamar, M.D., Abad-Segura, E.: Socioeconomic effects in cyberbullying: Global research trends in the educational context. International Journal of Environmental Research and Public Health **17**(12) (2020)
28. Malmasi, S., Zampieri, M.: Detecting Hate Speech in Social Media. In: Proceedings of RANLP (2017)
29. Mandl, T., Modha, S., Kumar M, A., Chakravarthi, B.R.: Overview of the hasoc track at fire 2020: Hate speech and offensive language identification in tamil, malayalam, hindi, english and german. In: Proceedings of FIRE (2020)
30. Mathew, B., Saha, P., Yimam, S.M., Biemann, C., Goyal, P., Mukherjee, A.: HateXplain: A Benchmark Dataset for Explainable Hate Speech Detection. In: Proceedings of AAAI (2021)
31. Moisio, A., Creutz, M., Kurimo, M.: Evaluating morphological generalisation in machine translation by distribution-based compositionality assessment. In: Proceedings of NoDaLiDa (2023)
32. Qian, J., Bethke, A., Liu, Y., Belding, E., Wang, W.Y.: A benchmark dataset for learning to intervene in online hate speech. In: Proceedings of EMNLP (2019)
33. Ranasinghe, T., Zampieri, M.: Multilingual Offensive Language Identification with Cross-lingual Embeddings. In: Proceedings of EMNLP (2020)
34. Ranasinghe, T., Zampieri, M.: Mudes: Multilingual detection of offensive spans. In: Proceedings of NAACL (2021)
35. Ranasinghe, T., Zampieri, M., Hettiarachchi, H.: BRUMS at HASOC 2019: Deep Learning Models for Multilingual Hate Speech and Offensive Language Identification. In: Proceedings of FIRE (2019)
36. Sanh, V., Webson, A., Raffel, C., Bach, S., Sutawika, L., Alyafeai, Z., Chaffin, A., Stiegler, A., Raja, A., Dey, M., Bari, M.S., Xu, C., Thakker, U., Sharma, S.S., Szczechla, E., Kim, T., Chhablani, G., Nayak, N., Datta, D., Chang, J., Jiang, M.T.J., Wang, H., Manica, M., Shen, S., Yong, Z.X., Pandey, H., Bawden, R., Wang, T., Neeraj, T., Rozen, J., Sharma, A., Santilli, A., Fevry, T., Fries, J.A.,

Teehan, R., Scao, T.L., Biderman, S., Gao, L., Wolf, T., Rush, A.M.: Multitask prompted training enables zero-shot task generalization. In: Proceedings of ICLR (2022)
37. Sarkar, D., Zampieri, M., Ranasinghe, T., Ororbia, A.: fbert: A neural transformer for identifying offensive content. In: Findings of EMNLP (2021)
38. Sarkar, R., KhudaBukhsh, A.R.: Are chess discussions racist? an adversarial hate speech data set. In: Proceedings of AAAI (2021)
39. Satapara, S., Majumder, P., Mandl, T., Modha, S., Madhu, H., Ranasinghe, T., Zampieri, M., North, K., Premasiri, D.: Overview of the hasoc subtrack at fire 2022: Hate speech and offensive content identification in english and indo-aryan languages. In: Proceedings of FIRE (2023)
40. Sharma, D., Buduru, A.B.: FAtNet: Cost-effective approach towards mitigating the linguistic bias in speaker verification systems. In: Findings of ACL: NAACL (2022)
41. Vidgen, B., Nguyen, D., Margetts, H., Rossini, P., Tromble, R.: Introducing CAD: the contextual abuse dataset. In: Proceedings of NAACL (2021)
42. Vidler, J., Rayson, P.: UCREL - Hex; a shared, hybrid multiprocessor system. `https://github.com/UCREL/hex`, accessed: 2024
43. Waseem, Z., Davidson, T., Warmsley, D., Weber, I.: Understanding Abuse: A Typology of Abusive Language Detection Subtasks. In: Proceedings of ALW (2017)
44. Weerasooriya, T.C., Dutta, S., Ranasinghe, T., Zampieri, M., Homan, C.M., KhudaBukhsh, A.R.: Vicarious offense and noise audit of offensive speech classifiers (2023)
45. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A.: Transformers: State-of-the-art natural language processing. In: Proceedings of EMNLP (2020)
46. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: Predicting the type and target of offensive posts in social media. In: Proceedings of NAACL (2019)
47. Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., Kumar, R.: SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In: Proceedings of SemEval (2019)
48. Zampieri, M., Morgan, S., North, K., Ranasinghe, T., Simmmons, A., Khandelwal, P., Rosenthal, S., Nakov, P.: Target-based offensive language identification. In: Proceedings of ACL (2023)
49. Zampieri, M., Nakov, P., Rosenthal, S., Atanasova, P., Karadzhov, G., Mubarak, H., Derczynski, L., Pitenis, Z., Çöltekin, c.: SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In: Proceedings of SemEval (2020)
50. Zampieri, M., Rosenthal, S., Nakov, P., Dmonte, A., Ranasinghe, T.: Offenseval 2023: Offensive language identification in the age of large language models. Natural Language Engineering **29**(6), 1416–1435 (2023)
51. Zia, H.B., Castro, I., Zubiaga, A., Tyson, G.: Improving zero-shot cross-lingual hate speech detection with pseudo-label fine-tuning of transformer language models. In: Proceedings of ICWSM (2022)