# Generalizable Multilingual Hate Speech Detection on Low Resource Indian Languages using Fair Selection in Federated Learning

**Akshay Singh**
Computer Science & Engineering
Indian Institute of Technology Roorkee
akshay_s@cs.iitr.ac.in

**Rahul Thakur**
Computer Science & Engineering
Indian Institute of Technology Roorkee
rahulthakur@ieee.org

## Abstract

Social media, originally meant for peaceful communication, now faces issues with hate speech. Detecting hate speech from social media in Indian languages with linguistic diversity and cultural nuances presents a complex and challenging task. Furthermore, traditional methods involve sharing of users' sensitive data with a server for model training making it undesirable and involving potential risk to their privacy remained under-studied. In this paper, we combined various low-resource language datasets and propose MultiFED, a federated approach that performs effectively to detect hate speech. MultiFED utilizes continuous adaptation and fine-tuning to aid generalization using subsets of multilingual data overcoming the limitations of data scarcity. Extensive experiments are conducted on 13 Indic datasets across five different pre-trained models. The results show that MultiFED outperforms the state-of-the-art baselines by 8% (approx.) in terms of Accuracy and by 12% (approx.) in terms of F-Score.

## 1 Introduction

In the age of social computing utilizing social media as a communication tool motivated by anonymity and ease has grown significantly in size and significance (Kar and Debbarma, 2023). Microblogs provide instantaneous worldwide idea sharing, information acquisition, company promotion, etc. This encourages free speech and makes the world more connected. Unfortunately, it has also led to an increase in abusive interactions, such as cyberbullying, profanity, hate speech, etc., directed at specific people and groups (Wachs et al., 2022). In addition to detracting from important conversations, the increasing amount of inappropriate information on the internet has the potential to instigate aggressiveness, which can lead to violent real-world situations(Arango et al., 2022; Mathew et al., 2019). Social media platforms are forced to carefully monitor user postings and actions in order to halt the spread of hate speech (Chatterjee et al., 2023). Therefore, there exists a pressing necessity to establish effective automated frameworks for identifying hate speech, ensuring a proactive approach to its mitigation.

The majority of existing work in hate speech detection is based on European languages or languages with a wider international user base (Nozza et al., 2022; Demus et al., 2022; Battistelli et al., 2020). Whereas only a limited amount of work is done on other languages due to limited resources and linguistic nuances. Specifically, in India, we have 22 official languages and people like to communicate in their regional languages (Kalra and Dutt, 2020). This makes social media interactions lack formal structure and frequently involve grammatical errors, emoticons, spelling issues, etc. Moreover, the mixing of one or more languages that can even be code-mixed makes hate speech detection extremely challenging (Jayanthi et al., 2021; Agarwal et al., 2021). As a result, we have a limited digital presence of training data. Hence, a very limited amount of work focuses on hate speech detection in Indian languages. This motivated us to work on hate speech detection for Indian languages.

To address data scarcity for hate speech detection in Indian languages, various works sourced datasets from social media platforms (Ramesh et al., 2022; Gupta et al., 2022). Also, competitions such as HASOC [1] and IndoML [2] have been launched to promote research in Indian languages by providing high-quality datasets. However, the scale and linguistic coverage of these datasets is sparse (Gupta et al., 2022). Therefore, most of the works in hate speech are either based on monolingual datasets or utilize only a few languages. Consequently, the model trained in fewer languages might not yield accurate results in other languages (Röttger et al.,

[1] https://hasocfire.github.io/hasoc/2023/index.html
[2] https://indoml.in/



*Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7208–7218

2022). To solve this, a few recent studies attempted to widen the coverage of training data by combining samples from multiple languages (Ranasinghe and Zampieri, 2021; Röttger et al., 2021; Saha et al., 2021). However combining various smaller datasets into one large-scale dataset may introduce inconsistency due to differences in annotation guidelines, data source platform, class imbalance, etc (Risch et al., 2021). In such scenarios, the performance of the centrally trained model may be highly impacted and biased. Therefore, adapting a single model to give a generalized performance in diverse domains while maintaining its effectiveness in each domain is challenging and complex.

Further, the success of a centralized computational framework relies on public datasets. But sensitive topics demand caution even with publicly available social media texts. The texts become private if a user makes their account private, impacting prior posts. Such privacy can be preserved if the model training utilizing these sensitive data happens at the user devices without the need to send the data to a common server. This is the motivation to use Federated Learning, suitable for sensitive data tasks emphasizing user privacy (Basu et al., 2022; Lin et al., 2022; Nagy et al., 2023). Apart from privacy risks, FL addresses user volume, info loss, and label imbalance (Gala et al., 2023; Gandhi et al., 2022). The objective of this paper is to design a generalized model using federated learning capable of detecting hate speech in multiple Indian languages. For this, we consider datasets from 13 Indian languages collected from various social media platforms. The publicly available different state-of-the-art BERT models such as XLM-RoBERTa (Conneau et al., 2020), multilingual-BERT (Devlin et al., 2019), MuRIL, Indic-BERT (Kakwani et al., 2020) similar to (Saha et al., 2021), and XLNet constitutes our centralized baselines for multilingual hate speech detection task. In summary, we make the following contributions:

- We aim to train and evaluate a classifier that effectively recognizes hate speech in a multilingual setting and gives generalized performance over unseen datasets.
- We utilize a fair selection approach for clients aiming for optimal performance across diverse datasets striking a balance between the personalized and generalized performance of the final model.
- We generate IID and non-IID dataset partitions across clients. These non-IID partitions replicate different types of distribution shifts (such as changes in labels, features, quantities, etc.) among clients, mirroring scenarios frequently encountered in real-world applications.
- We provide a comprehensive comparison between popular FL methods and state-of-the-art centralized methods for multilingual hate speech detection.
- We show that the model trained using federated learning gives better performance in diverse data environments which could be further scaled with acceptable performance variation along with preserving privacy of the user.

## 2 Related Work

**Centralized Learning** Previously, due to limited datasets, more focus was on non-Indian languages. Recently, more studies have attempted hate speech in Indian languages like Hindi (Shukla et al., 2022), Marathi (Patil et al., 2022a), Bengali (Das et al., 2022), etc. However, users often mix English with their native language (Code-Mixed data) for communication ease, complicating hate speech detection. Despite progress, research in bilingual, code-mixed, or multilingual tasks is nascent.

For example, (Fortuna et al., 2021) proposed standardization classes across publicly available datasets and studied the generalization capabilities of BERT, fastText, and SVM models. (Corazza et al., 2020) uses datasets for 3 different languages (English, Italian, and German) and trains different models such as LSTMs, GRUs, Bidirectional LSTMs, etc. The work claims to have a robust neural architecture for hate speech detection across different languages. Our work develops models that are far more generalizable and trained on a much larger dataset of languages. (Huang et al., 2020) constructed a multilingual Twitter hate speech corpus from 5 languages that they augmented with demographic information to study the demographic bias in hate speech classification. (Aluru et al., 2020) use datasets from 8 different languages and obtain their embeddings using LASER [3] and MUSE [4]. Though performance is decent across languages, fine-tuned models work best only for the

[3] https://github.com/facebookresearch/LASER
[4] https://github.com/facebookresearch/MUSE

8 languages. The Study focuses on low-resource performance, ignoring external impact. To address poor generalization of models due to biases in hate speech datasets, a suite of functional tests (HATECHECK) is introduced. HATECHECK offers insights into hate speech detection models but only tests English and modalities in text (Röttger et al., 2021). The extension of the English HATECHECK functional test suite is Multilingual HATECHECK (MHC), identifying model weaknesses for monolingual and cross-lingual applications, providing insights for the development of better multilingual hate speech detection models (Röttger et al., 2022).

**Federated Learning** Prior work on hate speech detection has primarily focused on privacy-agnostic machine learning paradigms, using centralized models for classification. In this work, we use a privacy-centric paradigm of machine learning, i.e., Federated Learning (FL). FL is a decentralized training strategy of machine learning models, a strategy reminiscent of parameter servers across a group of clients (McMahan et al., 2017). Clients train the model locally keeping data private from the server and peers. The parallel nature of training reduces the overall training time and enables training of models on a large corpus. This prevents the system from computational bottlenecks and minimizes the effect of heterogeneity in data on the global model performance (Zhu et al., 2021).
Recently several studies applied FL in many NLP tasks (Liu et al., 2021; Singh et al., 2022; Lin et al., 2022). The authors of (Basu et al., 2022) discussed training of NLP models and acknowledged the potential of methods like differential privacy, federated learning, and homomorphic encryption to handle sensitive data, e.g., medical records. However, the paper does not provide insights into the scalability or efficiency of the proposed framework for training NLP models on privacy-protected data. It is experimentally proved that FL ensures faster convergence when the number of clients is increased (Nagy et al., 2023). But the study lacks a discussion on framework limits in handling bigger datasets. In a similar study, the FL method is utilized as a privacy-preserving training paradigm for hate speech detection on eight datasets, surpassing centralized models aiming to address the lack of privacy in current approaches (Gala et al., 2023). Another study applied FL for multilingual emoji prediction in clean and attack situations, emphasizing its privacy and distributed advantages. However, it overlooks discussing how data distributions and biases could affect FL model performance (Gamal et al., 2023). Similar work is done in emoji detection for Hindi texts collected from Twitter considering FedProx (Li et al., 2020) and a modified version of CausalFedGSD (Francis et al., 2021) using federated approach (Gandhi et al., 2022).

**How is Our Approach Different?**

We explore cross-lingual meta-training in the domain of hate speech for both iid and non-iid configurations. Our proposed method of fair selection using federated learning, MultiFED, is a novel idea that can be further adapted for languages with no availability or limited availability of labeled data. We focus on resource maximization and domain generalization while transferring task-specific knowledge to low-resource languages. We carry out a large-scale study in multilingual hate speech detection across diverse domains on available hate speech datasets.

## 3 Methodology

### 3.1 Dataset Selection

This section provides a detailed description of the dataset used for experimentation. We created our multilingual dataset (*i.e. multicomb*) by combining datasets from 12 Indian languages and English, collected from various sources. Table 1 provides a detailed description of all the datasets.

Table 1: Detailed Description of Multicomb Dataset

| Dataset | Language | IID | | | Non-IID | | |
|---|---|---|---|---|---|---|---|
| | | Hate | Non Hate | Total | Hate | Non Hate | Total |
| (Gupta et al., 2022) | Hindi | 16152 | 16456 | 32608 | 8660 | 16456 | 25116 |
| (Gupta et al., 2022) | Tamil | 14429 | 13921 | 28350 | 7952 | 15571 | 23523 |
| (Gupta et al., 2022) | Telugu | 14397 | 14761 | 29158 | 7365 | 14761 | 22126 |
| (Gupta et al., 2022) | Kannada | 16045 | 15910 | 31955 | 8514 | 16890 | 25404 |
| (Gupta et al., 2022) | Malayalam | 12077 | 11629 | 23706 | 7622 | 13771 | 21393 |
| (Romim et al., 2021) | Bengali | 11000 | 11000 | 22000 | 7016 | 20000 | 27016 |
| (Patil et al., 2022b) | Marathi | 18750 | 18750 | 37500 | 9768 | 18750 | 28518 |
| IndoML | Bhojpuri | 8814 | 10304 | 19118 | 6831 | 16354 | 23185 |
| IndoML | Gujarati | 3948 | 5574 | 9522 | 3948 | 11994 | 15942 |
| IndoML | Haryanvi | 2658 | 3320 | 5978 | 2658 | 6103 | 8761 |
| IndoML | Odia | 6688 | 9518 | 16206 | 6688 | 20430 | 27118 |
| IndoML | Punjabi | 13000 | 12926 | 25926 | 6493 | 13000 | 19493 |
| TRAC-1 | English | 7588 | 8331 | 15919 | 4792 | 8415 | 13207 |

Our *multicomb* dataset consists of 300K (approx.) texts consisting of two classes namely *hate* and *non-hate*. A brief description of the source of utilized datasets:

- The first set of datasets is taken from baseline paper (Gupta et al., 2022). In their work, they provided datasets in **Hindi, Tamil, Telugu, Kannada, and Malayalam** languages. The samples have been scraped from sharechat [5].

[5]https://sharechat.com/

- The second set of datasets has been taken from a competition, i.e., "Indian Symposium on Machine Learning (IndoML) [6]". In this, they provided datasets in various Indian languages for multiple tasks. We consider datasets in **Bhojpuri, Gujarati, Haryanvi, Odia, and Punjabi** languages.

- The dataset in **Bengali** has been taken from (Romim et al., 2021). The samples are comments collected from YouTube and Facebook comment sections.The **Marathi** dataset has been curated from Twitter and annotated manually (Patil et al., 2022b). And the **English** dataset has been taken from TRAC-1 [7].

  **Note:** For more details, readers are requested to follow the respective citations.

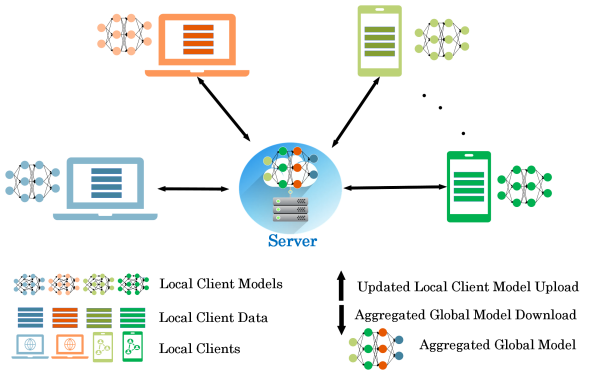


Figure 1: Federated Learning: An initial model shared by the server is fine-tuned locally on each client device and updated model is shared back to server for knowledge aggregation

### 3.2 Dataset-Client Partitioning

Three different training setups were carried out in this paper: traditional centralized training with no FL, FL with IID data, and FL with non-IID data.

In the centralized setup, we follow the dataset splits mentioned in (Gupta et al., 2022), i.e., balanced and unbalanced. In the balanced scenario, the number of training samples in both classes are equally distributed. However, in an unbalanced scenario, the number of samples are distributed in a 1:3 and 1:5 ratio (i.e., hate and non-hate). We randomly split *multicomb* in 80:10:10 ratio to form the training, validation, and test set and use this as the default split for all our experimental setups.

To create clients for FL experiments, we utilized the centralized split dataset as mentioned above. The balanced dataset is used to create FL clients with iid data (i.e., similar distribution of samples in both classes). Similarly, the unbalanced dataset is used to create FL clients with non-IID data (i.e., the distribution of samples in both classes follows a 1:3 and 1:5 ratio, hate and non-hate). It is worth mentioning that in both FL IID and non-IID, we carry out experiments while assuming that each client has samples from one language to follow real-world scenarios. Also, the validation and test splits are the same in both centralized and FL settings to make a fair comparison of results.

### 3.3 Models and Baselines

We evaluated our *multicomb* dataset by performing a series of experiments using 5 pre-trained transformer models in both centralized and federated settings.

**A. Models** Specifically, we employed state-of-the-art BERT models for this purpose: (1) XLM-RoBERTa (Conneau et al., 2020) is a multilingual variant of RoBERTa. (2) Multilingual-BERT (Devlin et al., 2019) was pre-trained on the top 104 languages with the largest Wikipedia, utilizing a masked language modeling (MLM) objective; (3) MuRIL (Khanuja et al., 2021) is a BERT model pre-trained on 16 Indian languages and their transliterated versions, using publicly available Wikipedia corpora; (4) IndicBERT (Kakwani et al., 2020) is a multilingual ALBERT model that received exclusive pre-training on 12 major Indian languages. (5) XLNet is an extension of the Transformer-XL model, trained using an auto-regressive approach to learn bidirectional contexts.

**B. Baselines** The state-of-the-art techniques that have been tested on our dataset are as follows.

1) Pre-Trained: All the state-of-the-art pre-trained BERT models.

2) Centralized: All the state-of-the-art BERT models in a centralized setting.

3) Finetuned (Gupta et al., 2022): In this, the authors randomly sample 5M comments out of the complete corpora and use these sampled comments for continued pretraining of the XLM-R model using masked language modeling (MLM) loss. We use the same strategy to finetune all BERT models.

For all the BERT models we use the tokenizers provided with each model. We measure their performance using weighted Accuracy, and weighted

[6] https://indoml.in/

[7] https://sites.google.com/view/trac1/home, https://github.com/kmi-linguistics/trac-1/tree/master/english

F1 scores.

### 3.4 Fair Selection

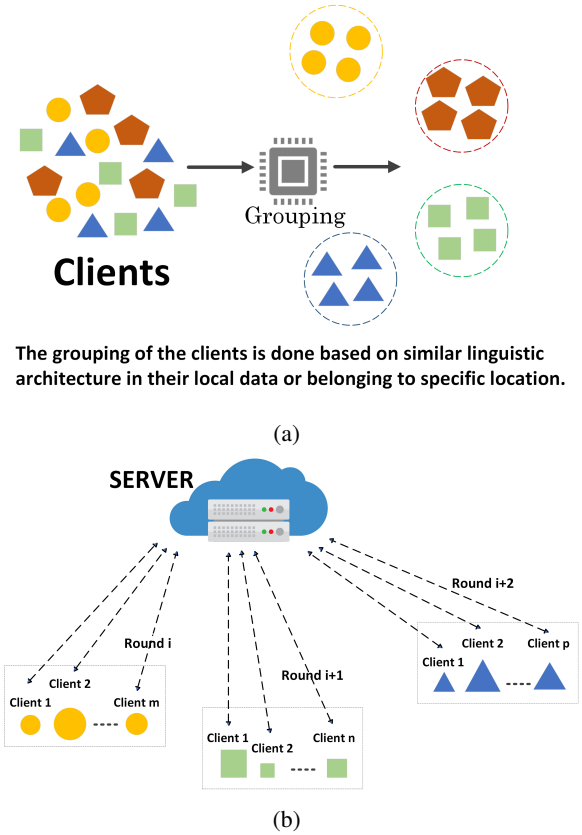


Figure 2: Fair Selection of Clients with Grouping

In FL setup, the clients are created by partitioning the *multicomb* dataset (3.2), which contains texts from various languages. To build a generalized model that performs optimally well on each language dataset is very complex. The generalization capability of any hate speech classifier is also affected when the training data is biased or contains samples where the definition of hate speech varies. To address this, we perform a grouping of clients based on their lexical similarity. Before the training phase, all the connected clients share a few local samples with the server. The server calculates the correlation among the clients using cosine similarity. Based on the similarity score, the group of clients is created (see Figure 2(a)). However, it may happen that the number of clients in few groups is too small or too high. Also, creating similar client groups may end up creating a large number of groups. To address this, we fixed the minimum number of clients in a group should not be less than 35 and more than 50.

After the groups are created, in each communication round, the server selects a group, and among the group, a set of clients are randomly selected to perform local training (see Figure 2(b)). These formed groups of similar clients can be considered as a set of independent and identically distributed (iid) clients. Hence, local training among similar clients minimizes the model weight divergence, which makes the model learn faster and doesn't hurt the accuracy much (Cao et al., 2022).

Additionally, we emphasize the selection of clients to be fair, diverse and increase the participation count of each participating client in the learning process. Adoption of the above approach for selection increases the coverage to diverse clients containing datasets from various languages. In our fair selection approach, all the groups are selected in sequence giving equal weightage to each data distribution, increasing coverage to all. We argue that without fair selection, the performance of the model may be biased towards the dataset of highly selected clients, and the idea of a generalized model will not be met.

### 3.5 Federated Training

Our proposed experimental investigation uses FL as a decentralized privacy-preserving approach instead of a centralized approach. This follows from the work (Nobata et al., 2016), which demonstrated that partitioning data into smaller segments can improve the classification performance overall. Following this we partitioned the entire dataset into several smaller segments as clients (described in section 3.2). In our work, we conceptualize client devices as users who witness and report hate speech. Before the experiments, we perform hyper-parameter tuning for the client learning rate, server-side learning rate, and proximal term (see Appendix B.1).

To initiate the process, a fraction 10%, 20%, or 30% of clients are selected randomly from 200 client devices in between two communication rounds. Then the locally computed model parameters by each client are aggregated by the server to compute new updated parameters. The FedAvg suffers from weight divergence and statistical heterogeneity. Therefore, to address this we use FedProx and FedOpt algorithms for the aggregation of parameters on the server side. FedProx adds a regularisation constant as the proximal term and FedOpt introduces a separate optimizer for the server-side model to account for data heterogeneity in an effort

to provide more reliable models.

#### 3.5.1 Why Proposed FL Setup Outperforms?

In each iteration, a random set of clients are selected among the selected group and perform local training. In the next iteration, a different group is selected, and the local training continues on another set of random clients (see Figure 2(b)). The training of the hate speech detection model on a set of similar clients resembles the domain adaptation phenomenon because hate speech from common datasets can be considered as separate domains. Now, aggregation of updated parameters received from the clients after each iteration build a new global model to be given to a set of clients from different domain. This is similar to transfer learning that involves leveraging knowledge gained while solving one problem and applying it to a different but related problem. Similar analogy is used for pre-trained models where transfer learning can significantly improve model performance, especially when the pre-trained model has been trained on a large and diverse dataset. In a similar manner, we are initializing the model with parameters learned while trained on one domain and finetuning it on another domain in each iteration. This helps the model to build generalizability over diverse datasets and improves overall performance.

### 3.6 Implementation Details

We implement our proposed model on the Python-based Pytorch, deep learning library. As the evaluation metric, we employ Weighted Accuracy and Weighted F1-score (F1) for hate speech detection. We use Adam as an optimizer, softmax as a classifier for hate speech classifier, and the categorical cross-entropy as a loss function, We used learning rate 2e-5 and batch size 16, epochs 1, 5, 10, and carried out experiments for 100 rounds.

Table 2: Values of Hyperparameters in Various Algorithms (lr: Learning Rate, BS: Batch Size)

| **Algorithm** | **FedAvg and FedProx** | | | **FedOpt** | | |
|---|---|---|---|---|---|---|
| **Model** | **Client_lr** | **BS** | $\mu$ | **Client_lr** | **BS** | **Server_lr** |
| XLMR | 2e-5 | 16 | 0.01 | 2e-5 | 16 | 0.01 |
| mBERT | 2e-5 | 16 | 0.01 | 2e-5 | 16 | 0.01 |
| MuRIL | 2e-5 | 16 | 0.01 | 2e-5 | 16 | 0.001 |
| IndicBERT | 2e-4 | 32 | 0.01 | 2e-4 | 32 | 0.001 |
| XLNet | 2e-6 | 32 | 0.001 | 2e-6 | 32 | 0.001 |

## 4 Results and Discussion

We assess the performance of our proposed MultiFED approach on five pre-trained BERT-based models under iid and non-iid data distribution settings. Firstly, we compare the generalized performance over the low-resource indic languages (see Table 3). Secondly, the trained models are also assessed on test data from each language individually giving the personalized performance (see Table 4). Please note that we reported the best results obtained in various experiments carried out for MultiFED.

Considering the performance for iid and non-iid settings from Table 3, we observe that the reported accuracy and f1-score vary by a small margin. This is due to the fact that the iid partitioning strategy divides an equal number of samples in each class while the non-iid partition carries unequal number of samples in each class. However, the total sample statistics in each client are similar in either setting. This is also evident from Table 1; the utilized datasets have a sufficient number of samples in each class. This is the reason for the robust performance of *Centralized* and *Finetuned* baselines in non-iid settings.

After the preliminary exploration, it is clearly visible that our proposed MultiFED framework outperformed the baseline by a margin of 7.78% & 11.58% in terms of accuracy and by 9.59% & 11.91% in terms of f1-score under iid and non-iid settings, respectively. We see that XLM-Roberta achieves the best performance in all experimental settings while XLNet gives suboptimal performance. It is interesting to observe the performance of baseline models that are pre-trained on various indic languages (such as **Muril** (Khanuja et al., 2021) and **Indicbert** (Kakwani et al., 2020)). Although these models are pre-trained on similar indic languages, XLMR and mBert beat them in most of the cases. This suggests the importance of continued training for domain adaptation. As evident from Table 8, FedProx gives better performance than FedAvg and FedOpt. Therefore, the results reported are experimented with using FedProx as the aggregation algorithm.

We also tested MultiFed on the centralized test data from each language separately to quantify the personalized performance of the proposed model (see Table 4). We observe that our proposed MultiFed model outperforms each baseline for most of the tested languages highlighting the importance of pretraining on domain-aligned partitioned data fractions. The linguistic patterns of few languages in our dataset are similar to each other. For exam-

Table 3: Results Comparison of Proposed Federated Setting with Centralized Baselines Settings

| Model | Pre-Trained | | | | Centralized | | | | Finetuned | | | | Federated | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | IID | | Non-IID | | IID | | Non-IID | | IID | | Non-IID | | IID | | Non-IID | |
| | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 |
| XLMR | 58.35 | 36.85 | 58.34 | 36.82 | 85.76 | 85.28 | 84.78 | 83.77 | 87.86 | 87.71 | 86.29 | 85.81 | 88.43 | 88.15 | 87.37 | 86.85 |
| mBERT | 58.31 | 36.86 | 41.47 | 31.48 | 84.67 | 84.11 | 84.50 | 83.39 | 86.75 | 86.59 | 86.17 | 85.53 | 87.79 | 87.47 | 86.88 | 86.14 |
| MuRIL | 41.65 | 29.41 | 41.65 | 29.40 | 85.25 | 84.50 | 83.82 | 82.78 | 87.03 | 86.87 | 86.44 | 85.79 | 87.74 | 87.42 | 87.12 | 86.46 |
| IndicBERT | 57.95 | 45.51 | 41.67 | 29.48 | 77.15 | 74.93 | 72.07 | 70.63 | 83.30 | 83.09 | 82.75 | 82.90 | 84.93 | 84.52 | 83.65 | 82.54 |
| XLNet | 58.18 | 37.05 | 41.78 | 30.96 | 65.59 | 65.43 | 66.76 | 60.24 | 67.67 | 67.58 | 66.52 | 63.65 | 69.75 | 68.62 | 68.75 | 64.48 |

Table 4: Results Comparison on Each Language Test Dataset

| Language | Pre-Trained | | | | Centralized | | | | Finetuned | | | | Federated | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | IID | | Non-IID | | IID | | Non-IID | | IID | | Non-IID | | IID | | Non-IID | |
| | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 | Acc | F1 |
| English | 58.62 | 36.95 | 58.53 | 38.78 | 57.96 | 57.01 | 51.70 | 48.56 | 74.17 | 73.85 | 66.23 | 66.14 | 67.44 | 67.29 | 66.10 | 66.00 |
| Haryanvi | 69.17 | 53.18 | 67.92 | 40.44 | 88.69 | 87.11 | 87.67 | 85.16 | 87.21 | 85.18 | 89.78 | 89.42 | 90.41 | 89.21 | 91.86 | 90.73 |
| Hindi | 52.05 | 34.23 | 51.21 | 37.38 | 83.70 | 83.65 | 78.18 | 77.69 | 83.70 | 83.69 | 85.07 | 85.04 | 86.26 | 86.25 | 85.41 | 85.33 |
| Tamil | 53.66 | 35.58 | 53.00 | 34.64 | 87.30 | 87.27 | 86.43 | 86.25 | 87.86 | 87.83 | 87.43 | 87.24 | 88.60 | 88.58 | 88.26 | 87.33 |
| Telugu | 52.13 | 34.26 | 52.12 | 35.06 | 90.20 | 90.19 | 89.53 | 89.52 | 90.66 | 90.65 | 87.20 | 86.69 | 90.67 | 90.65 | 88.27 | 88.26 |
| Kannada | 50.06 | 33.36 | 50.18 | 33.73 | 85.34 | 85.33 | 83.15 | 83.03 | 87.91 | 87.90 | 86.36 | 86.30 | 88.03 | 88.02 | 86.97 | 86.14 |
| Malayalam | 54.31 | 35.27 | 54.25 | 35.17 | 85.76 | 85.56 | 81.73 | 80.75 | 88.23 | 88.11 | 87.65 | 87.40 | 87.93 | 87.79 | 85.56 | 85.17 |
| Bengali | 66.60 | 40.07 | 66.67 | 40.00 | 84.03 | 82.25 | 81.30 | 78.48 | 87.23 | 86.17 | 86.80 | 85.56 | 87.94 | 86.03 | 87.66 | 86.12 |
| Marathi | 50.00 | 33.52 | 50.61 | 38.28 | 86.05 | 86.01 | 86.74 | 86.70 | 90.10 | 90.10 | 88.96 | 88.93 | 90.96 | 90.94 | 87.62 | 87.61 |
| Bhojpuri | 64.99 | 47.02 | 64.54 | 39.22 | 92.25 | 91.51 | 91.05 | 90.18 | 87.20 | 85.92 | 91.47 | 90.92 | 94.75 | 94.31 | 94.19 | 93.64 |
| Gujarati | 75.15 | 42.90 | 75.14 | 42.90 | 87.70 | 84.01 | 88.14 | 83.41 | 86.57 | 83.18 | 88.65 | 86.43 | 90.46 | 87.65 | 90.27 | 86.78 |
| Odia | 75.84 | 43.28 | 75.84 | 43.13 | 90.22 | 86.61 | 91.48 | 87.74 | 91.99 | 89.44 | 92.65 | 90.26 | 92.25 | 89.69 | 93.43 | 90.76 |
| Punjabi | 53.42 | 46.67 | 51.53 | 34.01 | 89.07 | 89.06 | 87.27 | 87.26 | 89.34 | 89.33 | 87.50 | 87.49 | 90.46 | 90.43 | 88.23 | 88.23 |

ple, Hindi, Bhojpuri, Haryanvi, and Punjabi share the same linguistic space. Hence, the model trained may be better at handling data sparsity issues. It can generalize well to other languages, even if the other dataset of the same domain has limited samples. This is the reason for poor performance on English dataset. Similarly, we can relate the reason for higher performance on Bhojpuri, Haryanvi, Gujarati, and Odia despite of having lesser number of samples. In federated learning, the globally aggregated model contains the collective intelligence from each participating client. Therefore it is expected to give optimal performance on each client which is reported in Table 3. However, the performance of the federated model may not perform equally good on each dataset belonging to different domains (e.g., language) compared to the centrally trained model (e.g., Malayalam and Marathi ). The reason for this biased performance lies in the imbalance of aggregated knowledge from each client due to the linguistic diversity of our *multicomb* dataset.

To understand this better, it is worth discussing the behavior of these utilized pre-trained models in our FL setup with different linguistic variability. All these models are pre-trained on large-scale diverse corpus of multilingual texts, including various Indian languages. Their capability to capture semantic similarity and differences across the languages tend to give significant performance across wide-range of languages without any extensive fine-tuning. However, the performance is dependent on the lexical similarity to the languages it was pre-trained. In this work, we fine-tuned these models using FL after creating groups of similar clients. Thus, these models leverage knowledge gained from one language to improve the performance in other facilitating the transferable characteristics across languages.

Table 5: Results Comparison of Proposed Federated Setting with Different Participation Ratio

| Federated | Non IID ($\alpha = 0.33$) | | | | | |
|---|---|---|---|---|---|---|
| | C= 10% | | C= 20% | | C= 30% | |
| Model | Acc | F1 | Acc | F1 | Acc | F1 |
| XLMR | 85.80 | 85.38 | 86.87 | 86.44 | 87.02 | 86.80 |
| mBERT | 82.74 | 81.86 | 85.67 | 84.88 | 86.88 | 86.14 |
| MuRIL | 84.17 | 83.73 | 86.98 | 86.37 | 87.07 | 86.36 |
| IndicBERT | 82.65 | 80.47 | 83.63 | 82.44 | 83.50 | 82.61 |
| XLNet | 66.36 | 63.22 | 68.41 | 64.54 | 68.63 | 64.83 |

Table 6: Results Comparison of Proposed Federated Setting with Different Local Epoch

| Federated | Non IID ($\alpha = 0.33$) | | | | | |
|---|---|---|---|---|---|---|
| | E = 1 | | E = 5 | | E = 10 | |
| Model | Acc | F1 | Acc | F1 | Acc | F1 |
| XLMR | 85.80 | 85.38 | 86.66 | 86.12 | 86.99 | 86.56 |
| mBERT | 82.74 | 81.86 | 83.29 | 82.18 | 84.00 | 83.11 |
| MuRIL | 84.17 | 83.73 | 86.25 | 85.59 | 86.60 | 86.03 |
| IndicBERT | 82.65 | 80.47 | 81.94 | 80.61 | 83.19 | 82.25 |
| XLNet | 66.36 | 63.22 | 68.17 | 66.59 | 67.40 | 64.34 |

### 4.1 Sensitivity Analysis

In this section, we analyze the sensitivity of MultiFED under various hyperparameter values.

Table 5 shows the performance under different participation ratios $C$ among {10%, 20%, 30%} for various model architectures on local epoch $E$=1 and FedProx. It is concluded that the performance increases on increasing $C$ regardless of the model architecture. When $C$ rises from 10% to 30%, the accuracy and f-score increases because the number of training rounds are increased for each client. However, few models report a drop in performance when $C$ increases from 20% to 30%. This drop may be the result of overfitting occurring in some of the clients. Similarly, in Table 6 (reported for $C = 10\%$), we can relate the performance rise with the increasing number of local training epochs.

Table 7: Results Comparison of Proposed Federated Setting with Different Data Heterogeneity

| **Federated** | $\alpha$ **= 0.5** | | $\alpha$ **= 0.33** | | $\alpha$ **= 0.2** | |
|---|---|---|---|---|---|---|
| **Model** | **Acc** | **F1** | **Acc** | **F1** | **Acc** | **F1** |
| XLMR | 87.26 | 87.51 | 86.87 | 86.44 | 87.39 | 72.83 |
| mBERT | 87.24 | 86.92 | 85.67 | 84.88 | 87.66 | 72.88 |
| MuRIL | 86.92 | 86.20 | 86.98 | 86.37 | 88.01 | 74.59 |
| IndicBERT | 84.27 | 83.98 | 83.63 | 82.44 | 79.89 | 65.54 |
| XLNet | 68.20 | 68.00 | 68.41 | 64.54 | 62.76 | 58.72 |

Table 8: Results Comparison of Proposed Federated Setting with Different Algorithm

| **Federated** | **FedOpt** | | **FedAvg** | | **FedProx** | |
|---|---|---|---|---|---|---|
| **Model** | **Acc** | **F1** | **Acc** | **F1** | **Acc** | **F1** |
| XLMR | 85.33 | 84.72 | 86.35 | 85.86 | 86.87 | 86.44 |
| mBERT | 84.13 | 83.10 | 85.65 | 84.77 | 85.67 | 84.88 |
| MuRIL | 86.54 | 85.87 | 86.93 | 86.25 | 86.98 | 86.37 |
| IndicBERT | 83.45 | 82.55 | 83.32 | 82.28 | 83.63 | 82.44 |
| XLNet | 67.31 | 60.50 | 68.24 | 61.69 | 68.41 | 64.54 |

We also analyze MultiFED on different levels of data heterogeneity ($\alpha$) on each language dataset, reported for $C = 20\%$ and $E = 1$. Here, $\alpha$ controls the degree of heterogeneity from 0.5 (or iid) to 0.33, and 0.2. Smaller $\alpha$ means a higher imbalance in the number of samples in each class. As shown in Table 7, the performance of all the models decreases as the imbalance in class is increased. It is worth mentioning, the accuracy for XLMR, mBert, and Muril when $\alpha = 0.33$ is lower than when $\alpha = 0.2$. However, if we closely look into the values of f-score, we conclude that the performance of all the models is better in the case of $\alpha = 0.33$. Similarly, Table 8 compares the performance of MultiFED in different aggregation scenarios. The experiments show that FedProx performs the best among the other algorithms as FedOpt is Well-suited for scenarios where the imbalance is not the primary concern (Ye et al., 2023). This is because FedProx employs a "proximal term" in the learning objective, penalizing deviations between local and global model parameters to promote the proximity of local models to the global one.

## 5 Conclusion

In this work, we propose a federated approach to hate speech detection, coined MultiFED, which aims to localize users' data and prevent it from being exposed while training models for hate speech detection. It also tackles the challenge of diverse linguistic data by personalizing the performance of clients. Through empirical experimentations, we find that Federated Learning along with preserving privacy achieves a higher performance level than centralized baselines in various scenarios. In future work, we intend to personalize the performance, improve fairness in the selection of clients, and reduce the resource consumption of each client.

## Limitations

Federated Learning preserves the privacy of users and improves communication efficiency in the learning process. However, multiple iterations involved during federated training may pose significant challenges due to issues such as limited bandwidth, high latency, and high communication overhead. Also, the participating user devices may not have sufficient resources to complete the training process. How good the device may be, it may struggle to reach convergence and may limit its performance.

We train our models using Federated Learning in both iid and non-iid settings, reported accuracy, and f-score to quantify the bias introduced due to data heterogeneity. However, due to limited resources and the high clock time taken for its experimentation, we limited the number of rounds to 100. The performance may improve if more iterations may have been added. Additionally, we carried out experiments under the assumption that each participating client has sufficient resources to train and infer these heavy BERT models. Additionally, if the data in each client is imbalanced in terms of labels as well as in total dataset size, federated learning may fail to give performance better than its centralized counterpart.

## Ethics Statement

Although Federated Learning can be a good choice to ensure anonymity for a user and their data, it still has many potential risks. In particular, a malignant user may initiate model threat attacks, which might compromise the integrity of the FL system and result in erroneous predictions. Furthermore, misclassification of the FL system is caused by attacks on model availability, which is more widespread and harmful than integrity breaches. One of the main privacy concerns for getting sensitive information unlawfully is the revelation of privacy data, which is caused by data leakage attacks that breach the confidentiality of training data. Additional benefits of federated learning include edge intelligence and personalization, which may enhance user experience and mitigate hate speech. However, in practical situations, it requires serious consideration to ensure that online conversations are healthy and that users may enjoy complete anonymity without worrying about data breaches.

## References


Vibhav Agarwal, Pooja Rao, and Dinesh Babu Jayagopi. 2021. Towards code-mixed Hinglish dialogue generation. In *Proceedings of the Student Research Workshop Associated with RANLP 2021*, pages 7–15, Online. INCOMA Ltd.

Sai Saketh Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. 2020. A deep dive into multilingual hate speech classification. In *Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track: European Conference*, page 423–439. Springer-Verlag.

Ayme Arango, Jesus Perez-Martin, and Arniel Labrada. 2022. HateU at SemEval-2022 task 5: Multimedia automatic misogyny identification. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 581–584, Seattle, United States. Association for Computational Linguistics.

Priyam Basu, Tiasa Singha Roy, Rakshit Naidu, Zumrut Muftuoglu, Sahib Singh, and Fatemehsadat Mireshghallah. 2022. Benchmarking differential privacy and federated learning for bert models.

Delphine Battistelli, Cyril Bruneau, and Valentina Dragos. 2020. Building a formal model for hate detection in french corpora. *Procedia Computer Science*, 176:2358–2365.

Mei Cao, Yujie Zhang, Zezhong Ma, and Mengying Zhao. 2022. C2s: Class-aware client selection for effective aggregation in federated learning. *High-Confidence Computing*, 2(3):100068.

Sankhadeep Chatterjee, Kushankur Ghosh, Arghasree Banerjee, and Soumen Banerjee. 2023. Forecasting covid-19 outbreak through fusion of internet search, social media, and air quality data: A retrospective study in indian context. *IEEE Transactions on Computational Social Systems*, 10(3):1017–1028.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Michele Corazza, Stefano Menini, Elena Cabrio, Sara Tonelli, and Serena Villata. 2020. A multilingual evaluation for online hate speech detection. *ACM Trans. Internet Technol.*, 20(2).

Mithun Das, Somnath Banerjee, Punyajoy Saha, and Animesh Mukherjee. 2022. Hate speech and offensive language detection in Bengali. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 286–296. Association for Computational Linguistics.

Christoph Demus, Jonas Pitz, Mina Schütz, Nadine Probol, Melanie Siegel, and Dirk Labudde. 2022. Detox: A comprehensive dataset for German offensive language and conversation analysis. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 143–153, Seattle, Washington (Hybrid). Association for Computational Linguistics.

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

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.

Sreya Francis, Irene Tenison, and Irina Rish. 2021. Towards causal federated learning for enhanced robustness and privacy.

Jay Gala, Deep Gandhi, Jash Mehta, and Zeerak Talat. 2023. A federated approach for hate speech detection. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3248–3259.

Karim Gamal, Ahmed Gaber, and Hossam Amer. 2023. Federated learning based multilingual emoji prediction in clean and attack scenarios.

Deep Gandhi, Jash Mehta, Nirali Parekh, Karan Waghela, Lynette D'Mello, and Zeerak Talat. 2022. A federated approach to predicting emojis in Hindi tweets. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11951–11961.

Vikram Gupta, Sumegh Roychowdhury, Mithun Das, Somnath Banerjee, Punyajoy Saha, Binny Mathew, hastagiri prakash vanchinathan, and Animesh Mukherjee. 2022. Multilingual abusive comment detection at scale for indic languages. In *Advances in Neural Information Processing Systems*, volume 35, pages 26176–26191.

Xiaolei Huang, Linzi Xing, Franck Dernoncourt, and Michael J. Paul. 2020. Multilingual Twitter corpus and baselines for evaluating demographic bias in hate speech recognition. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1440–1448, Marseille, France. European Language Resources Association.

Sai Muralidhar Jayanthi, Kavya Nerella, Khyathi Raghavi Chandu, and Alan W Black. 2021. CodemixedNLP: An extensible and open NLP toolkit for code-mixing. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 113–118, Online. Association for Computational Linguistics.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics*, pages 4948–4961.

Rajrani Kalra and Ashok K. Dutt. 2020. *Exploring Linguistic Diversity in India: A Spatial Analysis*, pages 391–403. Springer International Publishing.

Purbani Kar and Swapan Debbarma. 2023. Multilingual hate speech detection sentimental analysis on social media platforms using optimal feature extraction and hybrid diagonal gated recurrent neural network. *The Journal of Supercomputing*, pages 1–32.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks.

Bill Yuchen Lin, Chaoyang He, Zihang Ze, Hulin Wang, Yufen Hua, Christophe Dupuy, Rahul Gupta, Mahdi Soltanolkotabi, Xiang Ren, and Salman Avestimehr. 2022. FedNLP: Benchmarking federated learning methods for natural language processing tasks. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 157–175.

Ming Liu, Stella Ho, Mengqi Wang, Longxiang Gao, Yuan Jin, and He Zhang. 2021. Federated learning meets natural language processing: A survey.

Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. 2019. Spread of hate speech in online social media. In *Proceedings of the 10th ACM conference on web science*, pages 173–182.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Balázs Nagy, István Hegedűs, Noémi Sándor, Balázs Egedi, Haaris Mehmood, Karthikeyan Saravanan, Gábor Lóki, and Ákos Kiss. 2023. Privacy-preserving federated learning and its application to natural language processing. *Knowledge-Based Systems*, 268:110475.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the International Conference on World Wide Web*, page 145–153, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Debora Nozza, Federico Bianchi, and Giuseppe Attanasio. 2022. HATE-ITA: Hate speech detection in Italian social media text. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 252–260.

Hrushikesh Patil, Abhishek Velankar, and Raviraj Joshi. 2022a. L3Cube-MahaHate: A tweet-based Marathi hate speech detection dataset and BERT models. In *Proceedings of the Third Workshop on Threat, Aggression and Cyberbullying (TRAC 2022)*, pages 1–9.

Hrushikesh Patil, Abhishek Velankar, and Raviraj Joshi. 2022b. L3cube-mahahate: A tweet-based marathi hate speech detection dataset and bert models. In *Proceedings of the Third Workshop on Threat, Aggression and Cyberbullying (TRAC 2022)*, pages 1–9.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Tharindu Ranasinghe and Marcos Zampieri. 2021. An evaluation of multilingual offensive language identification methods for the languages of india. *Information*, 12(8).

Julian Risch, Philipp Schmidt, and Ralf Krestel. 2021. Data integration for toxic comment classification: Making more than 40 datasets easily accessible in one unified format. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 157–163.

Nauros Romim, Mosahed Ahmed, Hriteshwar Talukder, and Md. Saiful Islam. 2021. Hate speech detection in the bengali language: A dataset and its baseline evaluation. In *Proceedings of International Joint Conference on Advances in Computational Intelligence*, pages 457–468.

Paul Röttger, Haitham Seelawi, Debora Nozza, Zeerak Talat, and Bertie Vidgen. 2022. Multilingual HateCheck: Functional tests for multilingual hate speech detection models. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 154–169. Association for Computational Linguistics.

Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, and Janet Pierrehumbert. 2021. HateCheck: Functional tests for hate speech detection models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 41–58, Online. Association for Computational Linguistics.

Debjoy Saha, Naman Paharia, Debajit Chakraborty, Punyajoy Saha, and Animesh Mukherjee. 2021. Hatealert@DravidianLangTech-EACL2021: Ensembling strategies for transformer-based offensive language detection. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 270–276.

Shubham Shukla, Sushama Nagpal, and Sangeeta Sabharwal. 2022. Hate speech detection in hindi language using bert and convolution neural network. In *2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pages 642–647.

Pushpa Singh, Murari Kumar Singh, Rajnesh Singh, and Narendra Singh. 2022. *Federated Learning: Challenges, Methods, and Future Directions*, pages 199–214. Springer International Publishing.

Sebastian Wachs, Manuel Gámez-Guadix, and Michelle F Wright. 2022. Online hate speech victimization and depressive symptoms among adolescents: The protective role of resilience. *Cyberpsychology, Behavior, and Social Networking*, 25(7):416–423.

Mang Ye, Xiuwen Fang, Bo Du, Pong C. Yuen, and Dacheng Tao. 2023. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Comput. Surv.*, 56(3).

Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. 2021. Federated learning on non-iid data: A survey. *Neurocomput.*, 465(C):371–390.