FedBully: A Cross-Device Federated Approach for Privacy Enabled Cyber Bullying Detection using Sentence Encoders

Nisha P. Shetty[1], Balachandra Muniyal[1,*], Aman Priyanshu[1] and Vedant Rishi Das[2]

[1]*Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India-576104*
[2]*Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India-576104*
E-mail: nisha.pshetty@manipal.edu; bala.chandra@manipal.edu
[*]*Corresponding Author*

Received 24 May 2022; Accepted 25 March 2023;
Publication 21 June 2023

Abstract

Cyberbullying has become one of the most pressing concerns for online platforms, putting individuals at risk and raising severe public concerns. Recent studies have shown a significant correlation between declining mental health and cyberbullying. Automated detection offers a great solution to this problem; however, the sensitivity of client-data becomes a concern during data collection, and as such, access may be restricted. This paper demonstrates FedBully, a federated approach for cyberbullying detection using sentence encoders for feature extraction. This paper introduces concepts of secure aggregation to ensure client privacy in a cross-device learning system. Optimal hyper-parameters were studied through comprehensive experiments,

Journal of Cyber Security and Mobility, Vol. 12_4, 465–496.
doi: 10.13052/jcsm2245-1439.1242

and a computationally and communicationally inexpensive network is proposed. Experiments reveal promising results with up to 93% classification AUC (Area Under the Curve) using only dense networks to fine-tune sentence embeddings on IID datasets and 91% AUC on non-IID datasets, where IID refers to Independent and Identically Distributed data. The analysis also shows that data independence profoundly impacts network performance, with AUC decreasing by a mean of 5.1% between Non-IID and IID. A rich and extensive study has also been performed on client network size and secure aggregation protocols, which prove the robustness and practicality of the proposed model. The novel approach presented offers an efficient and practical solution to training a cross-device cyberbullying detector while ensuring client-privacy.

Keywords: Federated learning, convolutional neural network, secure aggregation, natural language processing, cyberbullying.

1 Introduction

The advent of social media has led to significant advancement in the field of modern communication. However, such advancements do not come without repercussions, cyberbullying being one of the significant ones. Cyberbullying comprises of bullying behaviours which may present negative connotations on the victim [1]. Victims of bullying have reported multiple symptoms of depressive and suicidal behaviour [2–4]. Therefore, a quick and automated detection method can be an efficient solution to prevent future cyberbullying cases [5–8]. However, samples/data from real-world bullying cases often consist of sensitive material (such as names of individuals/parties involved) and may often be censored or restricted. Hence, a federated approach that protects client privacy can offer a great solution to this.

Federated learning is a framework that allows models to be trained on privacy-sensitive data. In this methodology, raw data is not shared with participating clients/entities, which may or may not include a centralized server. This allows for the protection of sensitive client data, which is desirable for the focused problem statement. However, this class of algorithms is plagued with communication bottlenecks and adversarial central servers.

Therefore, to protect the anonymity of client data, solutions employing secure aggregation has been proposed. Secure aggregation is adopted to protect privacy-sensitive data against an insecure communication channel or a

malicious central server. By utilizing recent advances in natural language processing, the paper employs pre-trained sentence encoders to extract features from the input text data, which reduces the overall size of weights-per-client to be communicated. This paper formalizes a methodology to:

1. Adapt secure aggregation to the federated averaging methodology of aggregation. Implement FedBully for fast cyberbullying detection.
2. Analyze the performance of FedBully with the different protocols of secure aggregation, thereby showing that model performance remains unaffected by it.
3. Extensive experiments regarding sampling rate and dataset imbalance are studied, and sensitivity to both factors is analyzed.

2 Related Work

2.1 Privacy-Enabled Federated Learning

Federated learning is "a framework for privacy securing distributed learning" [9]. A module for on-device training and privacy-preserving machine learning. Numerous toolkits and architectures have been proposed in various areas of implementation such as computer vision [10–12], and next-word prediction in mobile keyboards [13–17]. It gives a real-world solution to the problem of client privacy for on-device sensitive data [18, 19]. For comparison techniques such as anonymization, critical information withholding, to name a few, have been implemented, however, these algorithms tend to reduce the final performance of the model and take enormous computational/manual effort to perform. However, a convenient workaround can be found using federated learning. It offers to solve collaborative learning problems while keeping the privacy of the users. Despite this, it has faced some significant issues, primarily due to:

- Limited resources for communication with all the devices participating in an aggregation round and sporadic dropping out of participating individuals.
- Sensitive content within contributed data about the clients/parties involved.

This paper looks at the above two problems and attempts to balance communication complexity, client-level privacy, and model performance. The following literature introduces a novel practical approach to the above complexities while ensuring time efficiency and high performance.

2.2 Cyberbullying Detection

Automatic cyberbullying detection offers a quick way to curb this toxic environment, and recent research has been invested in developing high-performing detectors [20–24]. Various deep learning approaches have been explored, including but not limited to using attention-based Bi-LSTM approach [25–28] and sentence encoders for the task [29]. J. Yadav et al. proposes BERT augmented with a linear fully connected dense layer as a classifier [30]. These studies provide insight and reinforce that sentence embeddings can give high performance in cyberbullying detection.

However, these papers do not critically analyze privacy issues and lack practical deployment due to the extensive pre-processing required to anonymize user datasets [31, 32]. FedBully offers a solution to this and provides client level anonymity by employing federated learning.

Author's Note: A recent contribution by Zhu et al. [33] proposes TextCNN, which utilizes institutionalized federated learning for intent classification. However, they employed word vectors such as word2vec. However, with rigorous development in the field of sentence encoders such as SBERT [34], Universal Sentence Encoders [35], and InferSent [36], their performance and applications have not been incorporated with federated learning. The following proposal integrates both sentence encoders and federated learning.

2.3 Secure Aggregation

The secure aggregation [37–39] framework proposes five baseline protocols to solve each threat level in a refining manner.

2.3.1 Protocol 0

Each protocol is developed in a series of refinements, looking at the proposed protocol 0 or masking with one-time pads. The proposed methodology calls for pair-wise secure communication channels for each client participating in the aggregation. It then asks each pair to select a matched pair of input perturbations, i.e., for a task of calculating $x = \sum_{u \in U} x_u$, the selected user, u samples a vector $s_{(u,v)}$ uniformly from a random distribution $[0, R)$ for every other user, v. Once selected every pair computed their perturbations using $p_{u,v} = s_{u,v} - s_{v,u}(mod\ R)$, thereby having the relation $p_{u,v} = p_{v,u}$ [37, 65]. Accurate aggregation is guaranteed as perturbations cancel each other, as validated by:

$$x = \sum_{u \in U} x_u + \sum_{u \in U} \sum_{v \in U} p_{u,v}$$

$$= \sum_{u \in U} x_u + \sum_{u \in U} \sum_{v \in U} s_{u,v} - \sum_{u \in U} \sum_{v \in U} s_{v,u}$$
$$= \sum_{u \in U} x_u (mod R)$$

Here, x_u is user data for user u, $\overline{x}$ is the securely computed mean, $s_{(u,v)}$ is a sampled vector from a random distribution $[0, R)$ for every other user, $p_{u,v} = s_{u,v} - s_{v,u}(mod R)$ represents the computed perturbations.

The proposed methodology guarantees perfect privacy for the users; however, it severely lacks robustness. If any user *u* drops out during communication, this protocol will be unable to return accurate results, thereby sabotaging the central model. At the same time, it requires the availability of highly secure pairwise communication channels, which can be expensive.

2.3.2 Protocol 1

Protocol 1 attempts to make protocol 0 more robust, by incorporating public-key cryptography, thereby reducing the complexity of key-sharing and secure communication channels required. At the same time, it allows flexibility by performing aggregation once a fixed number of users join the aggregation step. Once a required number of clients is achieved their data is communicated amongst the participating groups, allowing for unmasked aggregation [37, 67].

However, this methodology can be used to maliciously compute a target user's data x_k. By implying all other users to share their respective perturbations $S_{u_k,v}$, where u_k is the target client. Therefore, risking client privacy.

2.3.3 Protocol 2

This protocol refers to double masking as a methodology to thwart a malicious server, here each user performs double-masking to protect x_u by adding a random value, b_u. It is given by the equation:

$$x'_u = x_u b_u + \sum_u \sum_v S_{u,v}, \quad \textit{where } v \in U - \{u\}$$

During the unmasking round, the server makes a choice with respect to each user, $u \in U1$, from each surviving member $v \in U2$. The choice made either requests a share of sum $s_{u,v}$ or the perturbations associated, b_u. After gathering at least *t* shares of sum $s_{u,v}$ for all $u \in U1$ and *t* shares of b_u for

all $u \in U2$, allowing the server to reconstruct the secrets and thereafter the aggregate value [37, 66].

This system clearly increases privacy protection even during user droppage, it also incorporates double masking and inherently has stronger security. However, it is computationally expensive, thereby reducing the efficiency of the final aggregator model.

2.3.4 Protocol 3

Protocol 3 works towards efficient secret exchange, attempting to reduce computation and communication complexity. Understanding that the use of secret values could be contained as a vector of pseudorandom values derived from a cryptographically secure pseudorandom generator (PRG). Thus, only the seeds for $s_{u,v}$ and $\mathrm{b_u}$ must be communicated. Reducing the complexity of communication from sending a *k*-dimensional vector to a seed value. This protocol also incorporates the Diffie-Hellman secret key sharing protocol, where each user $u \in U$ broadcasts their public key at the beginning of the entire process.

The secrets are shared through their seed values, as inverse perturbations given by, $s_{u,v} = PRG(seed)$ and $s_{v,u} = -PRG(seed)$. Having learned the secret key, the server can reconstruct all the perturbations of user, *u* during the secret sharing round. The aggregation is done as before. This methodology drastically reduces computational and communication complexity [37, 65]. However, it lacks pairwise secure communication thus causing a trade-off between communication efficiency and security.

2.3.5 Protocol 4

Now exploring protocol 4 or minimizing trust in practice, as described above, the authors propose a server-mediated key agreement [37]. It also derives the concept of double masking from protocol 2 generating y_u, additional mask is also added given by,

$$y_u = x_u + b_u + \sum_{v \in U1} p_{u,v}(modR)$$

This allows for more efficient communication by reducing the complexity of the data to be transferred. On a final note, protocol 4 offers an efficient and practical means for aggregation and, therefore, is presented as the proposed methodology for aggregation [37].

In this paper, secure aggregation is implemented for the federated learning process to protect client privacy from the malicious central server. Protocols 0 and 4 are implemented to produce successful results. The FedBully

implementation of the methodology allows easier communication while reducing the computational load on client devices.

3 Proposed Methodology

3.1 Cross-Device Federated Learning for Sentence Embeddings with Secure Aggregation

FedBully employs federated averaging to classify toxic/bullying content from different clients involved in such incidents. The objective of the proposal:

- To produce a decentralized learning model.
- Make the model to apply SOTA NLP techniques, namely sentence encoders.

Unlike word embeddings, sentence encoders generate a single embedding for the entire sentence. This embedding can be taken as an input for the client models. For weight aggregation, the weight updates are computed and sent to the central server. This allows for faster convergence of the central model. However, the setting discussed in the proposal is cross-device federated learning which is riddled with lots of issues [13]. Secure aggregation is employed, which takes into account client dropouts and communication failure. To enable this, masking with one-time pads and minimizing trust in practice from practical secure aggregation for federated learning on user-held data has been implemented [37].

The algorithm presented in 1 is an integration of federated learning with sentence encoders for future extraction. To enable additional security, secure aggregation is introduced. For protocol 0 as described in Section 2.3, masking with one-time pads, pair-wise secure communication channels are implemented for each client participating in the aggregation. These connections are made across all clients, and perturbations are shared. Each client u then updates its weight with the sum of all the perturbations from every other client, then sends them to the central server for aggregation.

In contrast to protocol 0, protocol 4 or minimizing trust in practice includes double masking and key sharing and a solution to user dropouts [37] descripted in Section 2.3. A Diffie-Hellman secret key agreement protocol is followed to produce security in the practical set-up; at the same time, double masking (b_u) is employed to produce better security. A cryptographically secure pseudorandom generator (PRG) is used to seed the generation of perturbations, significantly reducing communication size. This is finally shared with the centralized server.

The pseudo-codes for the FedBully training procedure are portrayed in Algorithm 1, which takes into account protocol 4, minimizing trust in practice, which is more practical and substantial than masking with one-time pads.

The proposed algorithm differs from previous implementations of cyberbullying detection and federated intent classification in the following aspects:

- The proposed methodology protects client privacy and enables distributed learning, thereby allowing a more significant portion of the community to collaborate in its training. FedBully introduces federated learning to secure said methodology compared to previous implementations that have used only sentence encoders for cyberbullying detection. The proposed method is coherent with cross-device federated learning, where the number of participating clients is huge while permitting sampling.
- Additional security is ensured by using secure aggregation, minimizing trust in practice, which provides a secure method of communication while ensuring solutions to important issues like communication failures and drop-outs for clients. Compared to implementations of institutional federated learning for intent classification, the proposed methodology allows cross-device federated learning with secure aggregation, allowing client-dropouts, communication efficiency, and resource constraints, which are not handled by the prior.
- The proposed algorithm employs secure aggregation, which uses shared perturbations. In contrast to this, when differential privacy is involved, the training module on each dataset must conform to the predefined privacy limit. Therefore the number of samples that can be drawn from a dataset without violating the privacy limit is co-variant to the $E, q, \ \& \ \sigma$ which are privacy limit, sampling ratio, and noise multiplier respectively. FedBully breaks this barrier by using secure aggregation, giving a practical solution to a rather sensitive topic.

3.2 Handling IID/non-IID and Imbalanced Data Load

As described in cross-device federated learning for sentence embeddings with secure aggregation in Algorithm 1, this paper is aimed to introduce a practical and secure model for cyberbullying detection. Empirical data does not conform with any standards, including cross-silo federated learning standards [13]; therefore, practices must be developed according to

Algorithm 1 FedBully

Require: $\boldsymbol{K}$: K clients indexed by k
m: sample ratio required to begin aggregation
l: ratio required to complete aggregation
w: trainable parameters or weights
E: the number of epochs for local training
η: learning rate for training
D_k: text data of client k
FEDERATED TRAIN
Initialize $w^{(0)}$
for $t \in 1, \dots T$ **do**
  $U1 \leftarrow$ (Sampled m ratio of clients)
  C^{PK} declared for ever $u \in U1^{(t)}$
  for all $D_k \in U1^{(t)}$ in parallel **do**
    $w_k^{(t)} \leftarrow$ ClientUpdate($w^{(t-1)}$, D_k)
  end for
  $U2^{(t)} \leftarrow$ (ratio of clients remaining)
  for $client_v \in U2^{(t)}$ **do**
    store: either $p_{u,v} : u \in U1, v \in U2$ or $b_u : u \in U1$
    if stored 1 share of $p_{u,v} : u \in U1 \backslash U2$ and $b_u : u \in U2$ **then**
      break
    end if
  end for
  $w^{(t)} \leftarrow \sum_{u \in U2} w_u^{t+1} - \sum_{u \in U2} b_u - \sum_{u \in U2} \sum_{v \in U1 \backslash U2} p_{u,v}$
end for
function **ClientUpdate**($\boldsymbol{w_0}, \boldsymbol{d}$)
embedded $\leftarrow$ *sentenceEncoder*(d)
for $i \in 1, \dots, E$ **do**
  for $b \in$ batches B embedded **do**
    $w \leftarrow w - \eta \nabla l(w : b)$
  end for
end for
$\boldsymbol{w'} \leftarrow \boldsymbol{w}$
for $\boldsymbol{client_i} \in \boldsymbol{U1^{(t)} - client_k}$ **do**
  if $\boldsymbol{s_{k,i}}$ is not determined **then**
    $\boldsymbol{s_{k,i}} \leftarrow$ sampled from $[0, R)$
    $s_{i,k} \leftarrow$ sampled from $[0, R)$
  end if
  $\boldsymbol{p_{k,i}} \leftarrow \boldsymbol{PRG(s_{k,i}) - PRG(s_{i,k})(mod R)}$
  $\boldsymbol{w'} \leftarrow \boldsymbol{w'} + \boldsymbol{p_{k,i}}$
end for
  return $\boldsymbol{w'}$

cross-device federated learning. This procedure includes handling non-IID data and imbalanced data loads [40]. FedBully uses secure aggregation, which is not limited to differential privacy restrictions, E, q, and σ as mentioned above, which allows more freedom in the sets of data FedBully can handle. At the same time, secure aggregation allows for the dropping of users during the aggregation step.

Now discussing real-world datasets, one must understand that a single victim of cyberbullying will not contribute a significant number of samples for their case. Therefore, it becomes crucial that such standards for the training data be discussed. It must be understood that datasets may have bias, and the number of samples per client may be drastically different. An additive methodology is proposed alongside,

- Simulated non-IID for FedBully: In this methodology, the algorithm asks contributors only to submit cases of cyberbullying, while the central server simulates an equal number of entities trained on non-cyber bullying samples. This allows us to create an artificial balance within the dataset, allowing both sides to contribute only what is required. It also reduces the load of pre-processing required for annotating client-side data.

Practical non-IID for FedBully is a supplementary proposal to FedBully as it is capable of achieving similar results. Optimal values for local training epochs, sample ratio, and learning rate were determined in the experiments. The above methodology is implemented and fine-tuned, yielding optimal parameters to be used during implementation.

4 Experiments

4.1 Implementation Details

FedBully, a sentence-embedding based classifier to detect cyberbullying, incorporating the training procedure from federated averaging. The sentence embeddings allow sentences to be embedded in a context-aware manner, allowing any sequential classifiers to see cyberbullying easily.

Some of the sentence embedders that have been experimented with include Sentence BERT (SBERT) [34], Universal Sentence Encoders – DAN [35], and Universal Sentence Encoders - Transformers [35]. These embedders show notable improvements for contextual and intent-based classification of sentences, thereby increasing the performance of the text-based classifiers. The features extracted from these embedders are then

fed into different neural architectures, dense fully connected networks and convolutional neural networks-based classifiers.

Optimization of parameters in the model is done with respect to the binary crossentropy loss, i.e.,

$$CE = -y_1 log(f(s_1)) - (1 - y_1) log(1 - f(s_1))$$

Where, y_1 is the intent class label, and s_1 is the predicted probability of entry x being a positive case.

In the experiments, the training process is implemented using the federated averaging algorithm on a central server. The client devices were simulated using multiple centrally orchestrated systems, which each simulated client devices training in parallel, independent of each other. For experimentation, a cross-device setting was affected over a local-area network, with one central node and 20 to 35 distributed systems. Between different client instances, communication security was simulated using RSA encryption [41]. Each client machine was equipped with 8GB RAM.

FedBully is implemented on TensorFlow. On client updates, gradient-based training was optimized using the Adam optimizer [42], with an $lr = 0.001$. The batch size is fixed to 32 for dense networks in the reported experiments while using 16 for CNNs.

4.2 Dataset

The Cyberbullying-Datasets [43] is a public dataset of text data, carefully articulated for cyberbullying detection. This aforementioned assemblage is a aggregated from an ensemble to datasets from different sources task-oriented to the detection of cyberbullying without manual supervision. The dataset is a crowd source of data from various platforms, including “Kaggle”, “Twitter”, “Wikipedia Talk pages”, and “YouTube”. The data contains text with a binary label of whether it is a text which indicates bullying or not. The data consists of a variation of cyberbullying sentiments like hate speech, aggression, insults, and toxicity; however, this implementation aims to only detect cyberbullying [43].

This dataset consists of 448,874 samples, of which there were 57,651 (of which 30,536 were duplicates) positive cases and 391,223 negative cases. Therefore, to stimulate balance and uniformity, under-sampling is employed to balance data distribution. Thereby finally utilizing 27,115 positive samples and 27,155 negative samples, making a total of 54,230 samples. 43,000 samples comprised the training set and the remaining 11,230 for validation.

5 Result Analysis and Discussion

5.1 Baseline

For each sentence embedder, a fully connected layer with sigmoid activation was employed to translate the feature vector into sentence classification results. These networks were then trained and validated with K-fold-cross-validation (K = 5). The model converged after 25 epochs; Figure 1 illustrates the first sample's training progression during cross-validation. The model achieved 84.84% accuracy and 92.65% AUC. Other performance metrics can be checked in Table 1. This illustration serves as a baseline for all the following experiments.

5.2 FedBully – IID and Non-IID Data

For the experimentation on IID, multiple clients, K, were taken, each containing an equal number of text samples, with an approximately 1:1 category ratio. This methodology allowed model evaluation over the effect of various hyperparameters. On the other hand, for non-IID data, the dataset is first sorted according to its class and then divided into K number of clients in

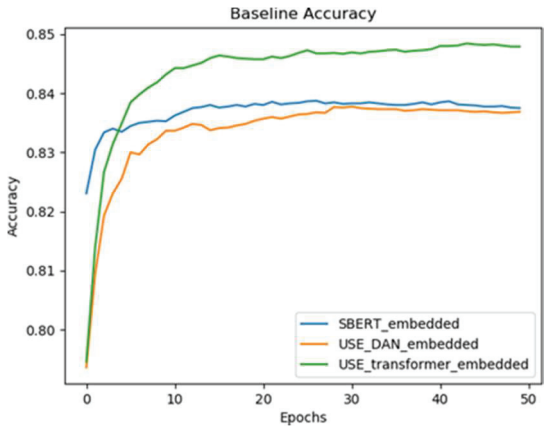

Figure 1 Training progress of Baseline networks.

Table 1 Baseline results

Model Name	Acc	AUC	F1-Score
USE-DAN-Dense	83.77	91.67	83.74
USE-Transformer-Dense	84.84	92.65	84.69
SBERT-Dense	83.87	91.93	83.53

sequential order; however, to represent client-level imbalance, each client receives x number of samples, which is uniformly chosen. Some key parameters to be noted are the K, m, E and η which are the total number of clients, the sampling ratio, the number of local epochs, the learning rate respectively. Without federated learning, the baseline model performance is reported in Table 1.

Looking at the experimental results for fixed local epochs $E = 5$ and learning rate $\eta = 0.001$, a general performance gain can be noted with the decrease in client training size and increase in the number of clients, for all possible networks, with a mean drop of 1.5% accuracy across different networks. At the same time, the effect of m becomes observable as it is gradually increased from 0.5 to 1.0 ratio of the original number of clients, noting a mean climb of 3.13% in accuracy for non-IID and 0.11% in AUC across different networks. The above results can be viewed in detail for all the architectures across different network types in Table 2 for the USE-Transformers network.

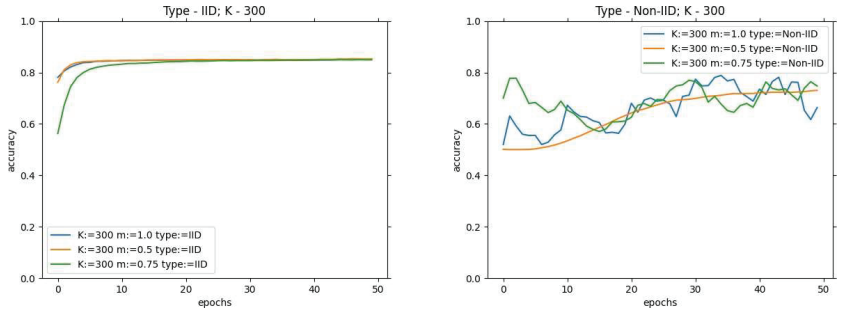

Figure 2 USE-Transformers learning progress (Accuracy) for variations on K and m, on fixed K = 300.

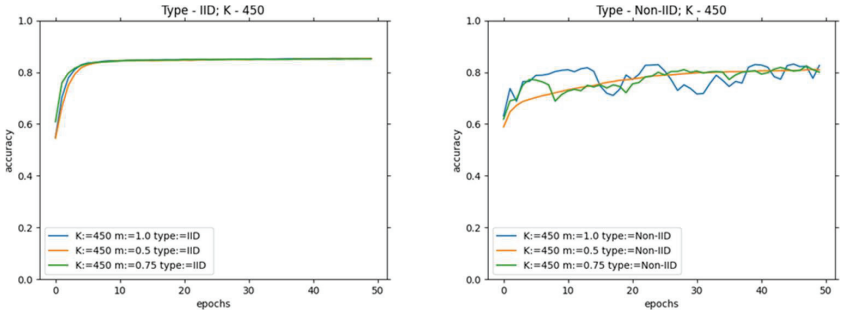

Figure 3 USE-Transformers learning progress (Accuracy) for variations on K and m, on fixed $K = 450$.

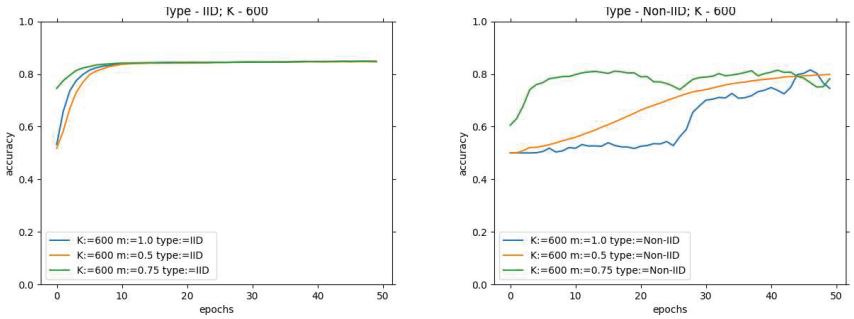

Figure 4 USE-Transformers learning progress (Accuracy) for variations on K and m, on fixed $K = 600$.

Table 2 USE-transformers analysis for fixed $\eta = 0.001$ and $E = 5$

K	m	Acc: IID	Acc: non-IID
300	0.5	85.11	73.20
300	0.75	84.95	77.78
300	1.0	85.34	78.84
450	0.5	85.34	81.13
450	0.75	85.16	82.47
450	1.0	85.36	83.16
600	0.5	84.78	79.83
600	0.75	84.90	81.44
600	1.0	84.86	81.55

Table 3 USE-Transformers Analysis for fixed $K = 450$ and $m = 1.0$ with a mean taken across learning rate (η)

E	Acc: IID	Acc: non-IID
1	79.48	73.31
5	81.27	78.02
10	83.08	80.41

In consideration of the above results, it can be realized that the ideal number of clients, K lies between [400, 500], and the ideal m lies between [0.9, 1.0]. Based on this, the effect of hyperparameters, E and η, with fixed $K = 450$ and $m = 1.0$, should be analyzed. Each of the architectures has been evaluated over $E = 1, 5, 10$ local epochs and $\eta = 0.1, 0.01, 0.001$ and their performance is outlined in Tables 3 and 4 for USE-Transformers. From these experiments, it becomes clear that the optimal values of E and η are 10 and 0.001, respectively.

Table 4 USE-Transformers Analysis for fixed $K = 450$ and $m = 1.0$ with a mean taken across local epochs (E)

η	Acc: IID	Acc: non-IID
0.1	81.36	76.43
0.01	81.94	77.58
0.001	81.47	78.93

Table 5 USE-Transformers Analysis for AUC metric variable K and m

K	m	AUC: IID	AUC: non-IID
600	1.0	88.11	85.72
600	0.75	88.29	85.36
600	0.5	87.88	85.54
450	1.0	90.37	87.91
450	0.75	90.46	87.25
450	0.5	90.18	87.54
300	1.0	92.06	89.28
300	0.75	88.63	88.77
300	0.5	91.73	89.4

Table 6 USE-Transformers Analysis for F1-Score metric variable K and m

K	m	F1-Score: IID	F1-Score: non-IID
600	1.0	84.44	67.40
600	0.75	84.33	73.95
600	0.5	84.19	76.76
450	1.0	85.39	81.55
450	0.75	85.05	77
450	0.5	85.29	78.44
300	1.0	85.07	64.42
300	0.75	84.44	67.71
300	0.5	85.21	50.02

5.3 Evaluation Metrics

The methodology presented is also analyzed based on other standard metrics. The paper includes an ablation study comparing the F1-score, sensitivity, specificity, precision, false-alarm, and AUC metrics. The metrics are presented in Tables 5, 6, 7, 8, 9, and 10.

Although AUC presents a metric to compare the performance of target classifiers, it does not validate key aspects such as sensitivity, precision, specificity, false alarm, and F1-Score. The paper demonstrates the above metrics in Tables 5, 6, 7, 8, 9, and 10. These key metrics are computed based

Table 7 USE-Transformers Analysis for False Alarm metric variable K and m

K	m	False-Alarm: IID	False-Alarm: non-IID
600	1.0	0.1289	0.0358
600	0.75	0.1207	0.0547
600	0.5	0.1234	0.0696
450	1.0	0.1551	0.1165
450	0.75	0.1409	0.0684
450	0.5	0.1443	0.0764
300	1.0	0.1457	0.0109
300	0.75	0.1207	0.036
300	0.5	0.1434	0.0271

Table 8 USE-Transformers Analysis for Precision metric variable K and m

K	m	Precision: IID	Precision: non-IID
600	1.0	86.48	93.63
600	0.75	87.12	91.88
600	0.5	86.87	90.54
450	1.0	84.73	86.84
450	0.75	85.7	90.72
450	0.5	85.5	90.09
300	1.0	85.34	96.88
300	0.75	87.15	93.65
300	0.5	85.55	94.74

Table 9 USE-Transformers Analysis for Sensitivity metric variable K and m

K	m	Precision: IID	Precision: non-IID
600	1.0	86.48	93.63
600	0.75	87.12	91.88
600	0.5	86.87	90.54
450	1.0	84.73	86.84
450	0.75	85.7	90.72
450	0.5	85.5	90.09
300	1.0	85.34	96.88
300	0.75	87.15	93.65
300	0.5	85.55	94.74

on standard mathematical and statistical equations presented in 4, 5, 6, 7, and 8.

The computed AUC value is presented in Table 5 displays important characteristic differences between IID and non-IID data. Even with high-performance results, non-uniformity and a noisy progression can be

Table 10 USE-Transformers Analysis for Specificity metric variable K and m

K	m	Specificity: IID	Specificity: non-IID
600	1.0	87.11	96.42
600	0.75	87.93	94.53
600	0.5	87.66	93.04
450	1.0	84.49	88.35
450	0.75	85.91	93.16
450	0.5	85.57	92.36
300	1.0	85.43	98.91
300	0.75	87.93	96.4
300	0.5	85.66	97.29

observed for non-IID data. This irregularity increases as there in an increment in number of clients, K. Sampling ratio, m, displays a similar progression-graph, supplementing the conclusion that an independent and identically distribution data performs vastly better than non-IID.

The performance of the system is measured with the following parameters. In this study, there are two outputs, whether a tweet is an act of cyberbullying or not. Based on this we employ traditional classification metrics computing standardized errors like true-positive, true-negative, false-positive, and false-negative. We expand on these metrics below.

The performance of the system is measured with the following and their extensions.

- **True Positive (TP):** When the tweet is labelled as an act of cyberbullying and the neural network also recognizes it as the same.
- **True Negative (TN):** When the tweet is not an act of cyberbullying, and the neural network also calculates it as the same.
- **False Positive (FP):** When a tweet was labelled as not a case of cyberbullying, but the neural network classifies it as one.
- **False Negative (FN):** When a tweet was labelled as cyberbullying, but the neural network predicts it as normal.

We incorporate the metrics, sensitivity, specificity, precision, F1-Score, and FalseAlarm. False alarm rate represents the proportion of false positives predicted by a classifier to the total number of negative labels.

We present these metric evaluations in the Tables 5, 6, 7, 8, 9, and 10. A confusion matrix is provided through the Figures 5, 6, 7 for the IID data setup. And a follow up is also displayed by the means of Figures 8, 9, and 10 for the non-IID data setup.

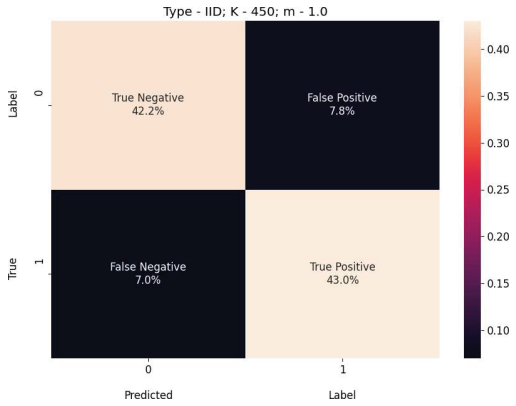

Figure 5 USE-Transformers confusion matrix for K = 450, m = 1.0, and IID-data.

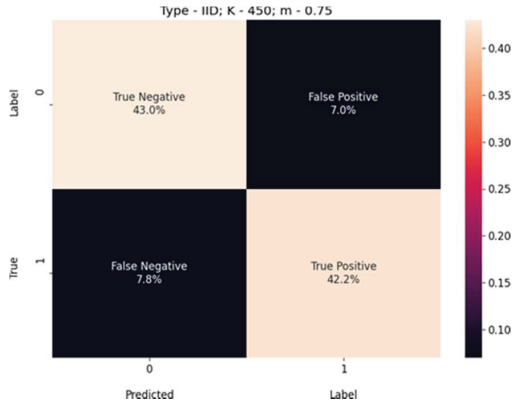

Figure 6 USE-Transformers confusion matrix for K = 450, m = 0.75, and IID-data.

Table 11 shows the difference in communicated data between protocol 0 and protocol 4 of secure aggregation, thereby supporting the claim for proposing protocol 4.

5.4 Simulated Non-IID for FedBully

This methodology allows the slightest effort to be put in by contributing entities. Since the central server only requires clients to send models trained on positive samples, they do not have to do manual annotation or labelling. Simultaneously, the central server's dataset imbalance can be avoided by generating an equal number of simulated clients trained in negative sample cases. This allows for lesser client-side effort and lowers computation and

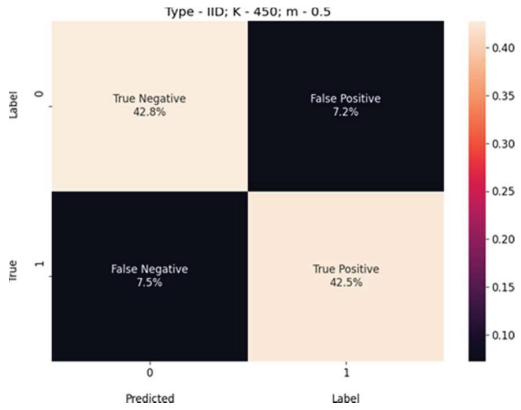

Figure 7 USE-Transformers confusion matrix for K = 450, m = 0.5, and IID-data.

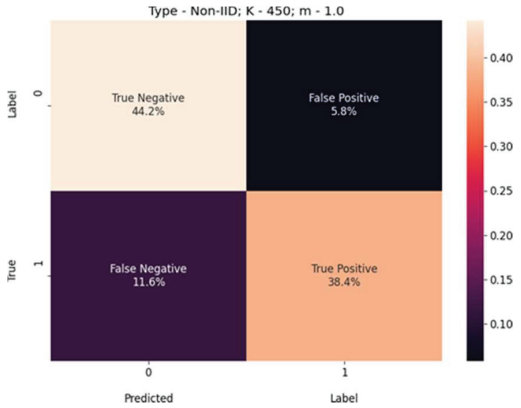

Figure 8 USE-Transformers confusion matrix for K = 450, m = 1.0, and non-IID-data.

communication costs as overall expenses are halved due to training on only a single set of labels. However, this methodology does have the drawback of reducing performance compared to the IID data setting. If simulated accurately, the training results and simulations will give results coherent to the non-IID setting, which was supported by the experiments.

5.5 Security Evaluation

The use of protocols 0 and 4 allows one to evaluate the security provided by secure aggregation. No clients can view any other client's subset information, allowing each client to be independently simulated. This means, whatever be

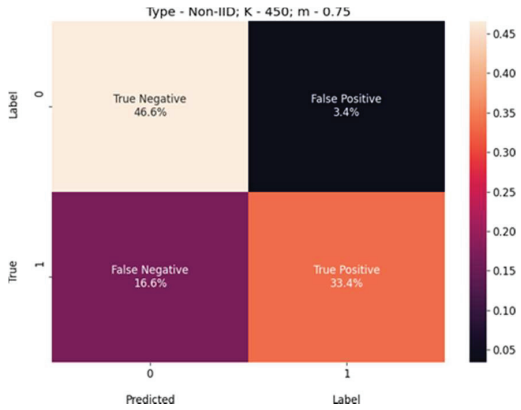

Figure 9 USE-Transformers confusion matrix for K = 450, m = 0.75, and non-IID-data.

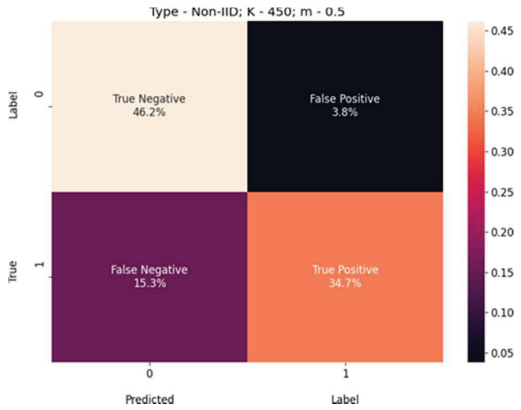

Figure 10 USE-Transformers confusion matrix for K = 450, m = 0.5, and non-IID-data.

Table 11 Communication-Complexity and Analysis between protocols

Model Name	Protocol-0	Protocol-1
USE-Transformer	38kB	13kB
USE-DAN	38kB	13kB
SBERT	43kB	17kB

the value of sampling ratio t, client-data is secure against each other. As for a malicious central server, the setting $t \geq n/2$ guarantees that the sum learned by the server contains the values of at least $t > n/2$ clients, and the protocol can deal with up to $n/2 - 1$ dropouts.

5.6 Privacy Analysis

The inclusion of federated learning, a privacy-enabled machine learning framework, allows the proposed methodology to quantify privacy leakage. The paper discusses two metrics used for further evaluations.

5.6.1 Number of times each sample is revisited

The experiments demonstrated above and scrutinized and analyzed to return the number of times each sample is revisited during the learning progression of a given neural network. This numeric value can then be used as a theoretical measure for possible privacy leakage. Each time a network learns over given data, it carries critical information, which may allow reconstruction of this input data [44, 69]. Keeping m = 1.0, the proposed methodology is evaluated against varying K values. We display the computed results in Table 12.

IID data demonstrates that the number of times each sample is revisited during training may be directly correlated to the number of clients active for communication in the learning step [68]. On the other hand, non-IID data does not follow any strict relation, and therefore its noisy learning curve does not allow one to draw any specific conclusions.

5.6.2 Sampling ratio inclusive computation of number of times each sample is revisited

Although the number of times each sample is evaluated provides a quantifiable measure of privacy for federated learning models, it does not take into account dropouts during communication which is an important aspect of the proposed methodology. Therefore, an evaluation study is purposed for this specific task, where the observations taking into account the sampling ratio are included in Table 13.

Taking into account the sampling ratio of inclusion of a particular data-sample at the aggregation step, a uniform increase can still be observed with an increase in K [44, 70]. Another observation is the increase in the number of reiterations required to return best-performance results.

Table 12 Number of times each sample is evaluated, with K variable

K	IID	Non-IID
300	85	165
450	140	100
600	250	235

Table 13 Sampling ration inclusive computation of number of times each sample is revisited, with m variable (IID-Data)

m	300	450	600
1.0	85	140	250
0.75	125	117	133
0.5	127	153	146

Table 14 Comparision of Proposed Approach algorithms against existing methods

Existing Research	Accuracy	AUC	F1-Score
[45]	0.76	–	–
[46]	0.88	–	–
[47]	0.48	–	–
[48]	0.76	–	–
[49]	–	–	0.73
[50]	–	–	0.81
[51]	–	–	0.91
[52]	–	–	0.64
[53]	–	0.83	–
[54]	–	0.83	0.75
[55]	–	0.8	0.68
[56]	–	0.9	0.91
[57]	–	0.91	0.91
[58]	–	0.89	–
[59]	–	0.92	–
[60]	–	0.68	–
[61]	–	0.81	–
[62]	–	–	0.37
[63]	–	–	0.83
[64]	–	–	0.84
USE-DAN-Dense	0.84	0.92	0.84
SBERT-Dense	0.84	0.92	0.84
USE-Transformer-Dense	0.85	0.93	0.85

5.7 Comparison with Existing Methods

Table 14 compares the proposed approach algorithms to other significant research efforts in the same area, exhibiting the proposed approach's enhanced performance in terms of accuracy, AUC, and F1 Scores. Cyberbullying is a pervasive issue that has adverse effects on individuals and society as a whole. Prior methodologies in detecting cyberbullying have shown some success, but they are often limited by factors. We highlight this limitation and

clearly distinguish the improvement FedBully brings. The proposed methodology addresses these limitations and outperforms a significant number of previous literature in cyberbullying detection.

6 Conclusion

FedBully is a novel implementation combining security principles with machine learning to detect cyberbullying. It removes the requirement of a centralized training system and also allows for rapid and progressive learning over a distributed setting. Using secure aggregation will enable us to ensure client data privacy and promotes the protection of sensitive data. Unlike previous work, this paper addresses a cross-device federated setting for intent classification. At the same time, FedBully is lightweight yet efficient, allowing training on restricted resource platforms, such as mobile phones/low-resource devices, and others. FedBully employs sentence encoders, which give significant results for cyberbullying detection, however unlike other implementations FedBully, secures client-data privacy. Extensive experiments show that data independence has decisive significance in model performance; simultaneously, the model proves robust towards change in local epochs and m (sampling ratio). To address practical implementation and dataset imbalance, the paper proposes two methodologies, weighted aggregation for FedBully and simulated non-IID for FedBully, which aim to improve accuracy and reduction in manual pre-processing, respectively. Future development could work upon these two methodologies to efficiently solve the same.

Cyberbullying can heavily impact people suffering under it. A robust, quick, and efficient methodology that can be crowd-sourced would prove to be an efficient solution to detect the same automatically. Such automatic detection may allow faster and accurate reporting of cyberbullying cases, allowing administrative authorities to take effect immediately. Federated learning and secure aggregation offer an efficient solution to this social issue, and execution of such methodology can benefit its people undergoing it.

References

[1] J.W. Patchin, S. Hinduja, 'Bullying Beyond the Schoolyard: Preventing and Responding to Cyberbullying', Corwin: Thousand Oaks, CA, 2014. ISBN: 1483349934.

[2] K. RIGBY, 'Effects of peer victimization in schools and perceived social support on adolescent well-being', Journal of Adolescence, 2000, 23, 57–68. doi: https://doi.org/10.1006/jado.1999.0289.
[3] K. Rigby, 'What children tell us about bullying in schools', Children Australia, 1997, 22, 28–34. doi: 10.1017/S1035077200008178.
[4] C.F. Yen, T.L. Liu, P. Yang, H.F. Hu, 'Risk and Protective Factors of Suicidal Ideation and Attempt among Adolescents with Different Types of School Bullying Involvement', Archives of Suicide Research, 2015, 19, 435–452, [https://doi.org/10.1080/13811118.2015.1004347930]. PMID: 26566860, doi: 10.1080/13811118.2015.1004490.
[5] H. Rosa, N. Pereira, R. Ribeiro, P. Ferreira, J. Carvalho, S. Oliveira, L. Coheur, P. Paulino, A. Veiga Simão, 'Trancoso, I. Automatic cyberbullying detection: A systematic review', Computers in Human Behavior 2019, 93, 333–345. doi: https://doi.org/10.1016/j.chb.2018.12.021.
[6] L. Cheng, Y.N. Silva, D. Hall, H. Liu, 'Session-Based Cyberbullying Detection: Problems and Challenges', IEEE Internet Computing, 2021, 25, 66–72. doi: 10.1109/MIC.2020.3032930.
[7] M.A. Al-garadi, K.D. Varathan, S.D. Ravana, 'Cybercrime detection in online communications: The experimental case of cyberbullying detection in the Twitter network', Computers in Human Behavior, 2016, 63, 433–443. doi: https://doi.org/10.1016/j.chb.2016.05.051.
[8] V. Nahar, S. Al-Maskari, X. Li, C. Pang, 'Semi-supervised Learning for Cyberbullying Detection in Social Networks', In Proceedings of the Databases Theory and Applications; H. Wang, M.A. Sharaf, Eds.; Springer International Publishing: Cham, 2014; pp. 160–171.
[9] J. Konečný, H.B. McMahan, D. Ramage, P. Richtárik, 'Federated Optimization: Distributed Machine Learning for On-Device Intelligence', 2016, [arXiv:cs.LG/1610.02527]
[10] L.T. Phong, Y. Aono, T. Hayashi, L. Wang, S. Moriai, 'Privacy-Preserving Deep Learning via Additively Homomorphic Encryption'. Trans. Info. For. Sec. 2018, 13, 1333–1345.
[11] T. Li, A.K. Sahu, A. Talwalkar, V. Smith, 'Federated Learning: Challenges, Methods, and Future Directions'. IEEE Signal Processing Magazine 2020, 37, 50–60. doi: 10.1109/MSP.2020.2975749.
[12] B. Liu, B. Yan, Y. Zhou, Y. Yang, Y. Zhang, 'Experiments of Federated Learning for COVID-19 Chest X-ray Images', 2020, [arXiv:eess.IV/2007.05592].
[13] P. Kairouz, H.B. McMahan, B. Avent, A. Bellet, M. Bennis, A.N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et

al. 'Advances and Open Problems in Federated Learning', 2021, [arXiv:cs.LG/1912.04977].

[14] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, D. Ramage, 'Federated Learning for Mobile Keyboard Prediction', 2019, [arXiv:cs.CL/1811.03604].

[15] S. Ramaswamy, R. Mathews, K. Rao, F. Beaufays, 'Federated Learning for Emoji Prediction in a Mobile Keyboard', 2019, [arXiv:cs.CL/1906.04329].

[16] M. Chen, R. Mathews, T. Ouyang, F. Beaufays, 'Federated Learning Of Out-Of-Vocabulary Words', 2019, [arXiv:cs.CL/1903.10635].

[17] H. Miyajima, N. Shigei, H. Miyajima, N. Shiratori, 'Federated Learning with Divided Data for BP. Proceedings of the International MultiConference of Engineers and Computer Scientists 2021', 2021, pp. 94–99.

[18] L.U. Khan, W. Saad, Z. Han, E. Hossain, C.S. Hong, 'Federated Learning for Internet of Things: Recent Advances, Taxonomy, and Open Challenges'. IEEE Communications Surveys Tutorials 2021, 23, 1759–1799. doi: 10.1109/COMST.2021.3090430.

[19] K. Pillutla, S.M. Kakade, Z. Harchaoui, 'Robust Aggregation for Federated Learning', 2019, [arXiv:stat.ML/1912.13445].

[20] O. Gencoglu, 'Cyberbullying Detection With Fairness Constraints' IEEE Internet Computing 2021, 25, 20–29. doi: 10.1109/MIC.2020.3032440561.

[21] V. Balakrishnan, S. Khan, H.R. Arabnia, 'Improving cyberbullying detection using Twitter users' psychological features and machine learning. Computers Security 2020, 90, 101710. doi: https://doi.org/10.1016/j.cose.2019.101710.

[22] A. Muneer, S.M. Fati, 'A Comparative Analysis of Machine Learning Techniques for Cyberbullying Detection on Twitter', Future Internet 2020, 12. doi: 10.3390/fi12110187.

[23] S. Nadali, M.A.A. Murad, N.M. Sharef, A. Mustapha, S. Shojaee, 'A review of cyberbullying detection: An overview'. In Proceedings of the 2013 13th International Conference on Intelligent Systems Design and Applications, 2013, pp. 325–330. doi: 10.1109/ISDA.2013.6920758.

[24] H. Miyajima, H. Miyajima, N. Shiratori, 'Fast and Secure Edge-computing Algorithms for Classification Problems', IAENG International Journal of Computer Science, 2019.

[25] N. Rezvani, A. Beheshti, A. Tabebordbar, 'Linking Textual and Contextual Features for Intelligent Cyberbullying Detection in Social Media', In Proceedings of the Proceedings of the 18th International Conference

on Advances in Mobile Computing Multimedia; Association for Computing Machinery: New York, NY, USA, 2020; MoMM '20, pp. 3–10. doi: 10.1145/3428690.3429171.

[26] M. Dadvar, K. Eckert, 'Cyberbullying Detection in Social Networks Using Deep Learning Based Models', In Proceedings of the Big Data Analytics and Knowledge Discovery; M. Song, I.Y. Song, G. Kotsis, A.M. Tjoa, I. Khalil, Eds.; Springer International Publishing: Cham, 2020; pp. 245–255.

[27] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, B. Xu, 'Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification', In Proceedings of the Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers); Association for Computational Linguistics: Berlin, Germany, 2016; pp. 207–212. 423 doi: 10.18653/v1/P16-2034.

[28] P. Wanda, J.H. Jie, 'DeepSentiment : Finding Malicious Sentiment in Online Social Network based on Dynamic Deep Learning', IAENG International Journal of Computer Science, 2019, pp. 4–12.

[29] V. Indurthi, B. Syed, M. Shrivastava, N. Chakravartula, M. Gupta, V. Varma, 'FERMI at SemEval-2019 Task 5: Using Sentence embeddings to Identify Hate Speech Against Immigrants and Women in Twitter', In Proceedings of the Proceedings of the 13th International Workshop on Semantic Evaluation; Association for Computational Linguistics: Minneapolis, Minnesota, USA, 2019, pp. 70–74. doi: 10.18653/v1/S19-2009.

[30] J. Yadav, D. Kumar, D. Chauhan, 'Cyberbullying Detection using Pre-Trained BERT Model', In Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 1096–1100. doi: 10.1109/ICESC48915.2020.9155700.433

[31] M. Ptaszynski, F. Masui, T. Nitta, S. Hatakeyama, Y. Kimura, R. Rzepka, K. Araki, Sustainable cyberbullying detection with category-maximized relevance of harmful phrases and double-filtered automatic optimization. International Journal of Child-Computer Interaction 2016, 8, 15–30. doi: https://doi.org/10.1016/j.ijcci.2016.07.002.

[32] M. Dadvar, D. Trieschnigg, R. Ordelman, F. de Jong, 'Improving Cyberbullying Detection with User Context', In Proceedings of the Advances in Information Retrieval, P. Serdyukov, P. Braslavski, S.O. Kuznetsov, J. Kamps, S. Rüger, E. Agichtein, I. Segalovich, E. Yilmaz, Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2013; pp. 693–696.

[33] X. Zhu, J. Wang, Z. Hong, J. Xiao, 'Empirical Studies of Institutional Federated Learning For Natural Language Processing', In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2020; Association for Computational Linguistics: Online, 2020; pp. 625–634. doi: 10.18653/v1/2020.findings-emnlp.55.

[34] N. Reimers, I. Gurevych, 'Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks', 2019, [arXiv:cs.CL/1908.10084].

[35] D. Cer, Y. Yang, S. yi Kong, N. Hua, N. Limtiaco, R.S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, et al. 'Universal Sentence Encoder', 2018, [arXiv:cs.CL/1803.11175].

[36] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, A. Bordes, 'Supervised Learning of Universal Sentence Representations from Natural Language Inference Data', 2018, [arXiv:cs.CL/1705.02364].

[37] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H.B. McMahan, S. Patel, D. Ramage, A. Segal, K. Seth, Practical Secure Aggregation for Federated Learning on User-Held Data, 2016, [arXiv:cs.CR/1611.04482].

[38] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, H.B. McMahan, et al. 'Towards Federated Learning at Scale: System Design', 2019, [arXiv:cs.LG/1902.01046].

[39] H. Miyajima, N. Shigei, H. Miyajima, N. Shiratori, 'Securely Distributed Computation with Divided Data for Particle Swarm Optimization', Proceedings of the International MultiConference of Engineers and Computer Scientists 2021, 2021, pp. 1–6.

[40] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, V. Chandra, 'Federated Learning with Non-IID Data', 2018, [arXiv:cs.LG/1806.00582].

[41] P. Parmar, S.B. Padhar, S.N. Patel, N.I. Bhatt, R. Jhaveri, 'Survey of Various Homomorphic Encryption algorithms and Schemes', International Journal of Computer Applications 2014, 91, 26–32.

[42] D.P. Kingma, J. Ba, 'Adam: A Method for Stochastic Optimization', 2017, [arXiv:cs.LG/1412.6980].

[43] F. Elsafoury, 'Cyberbullying datasets', 2020. doi:10.17632/jf4pzyvnpj.1.

[44] J. Geiping, H. Bauermeister, H. Dröge, M. Moeller, 'Inverting Gradients – How easy is it to break privacy in federated learning?', 2020, [arXiv:cs.CV/2003.14053].

[45] Dadvar, M., Trieschnigg, D., de Jong, F. Experts and Machines Against Bullies: A Hybrid Approach to Detect Cyberbullies. 2014, Vol. 8436. doi: 10.1007/978-3-319-06483-3_25.

[46] . Potha, N., Maragoudakis, M. Cyberbullying Detection using Time Series Modeling. In Proceedings of the 2014 IEEE International Conference on Data Mining Workshop, 2014, pp. 373–382. doi: 10.1109/ICDMW.2014.170.

[47] Del Bosque, L.P., Garza, S.E. Aggressive Text Detection for Cyberbullying. In Proceedings of the Human-Inspired Computing and Its Applications; Gelbukh, A., Espinoza, F.C., Galicia-Haro, S.N., Eds., Springer International Publishing: Cham, 2014; pp. 221–232.

[48] Ibn Rafiq, R., Hosseinmardi, H., Han, R., Lv, Q., Mishra, S., Mattson, S.A. Careful what you share in six seconds: Detecting cyberbullying instances in Vine. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2015, pp. 617–622. doi: 10.1145/2808797.2809381.

[49] Waseem, Z., Hovy, D. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In Proceedings of the Proceedings of the NAACL Student Research Workshop; Association for Computational Linguistics: San Diego, California, 2016; pp. 88–93. doi: 10.18653/v1/N16-2013.

[50] Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y. Abusive Language Detection in Online User Content. In Proceedings of the Proceedings of the 25th International Conference on World Wide Web; International World Wide Web Conferences Steering Committee: Republic and Canton of Geneva, CHE, 2016; WWW '16, p. 145–153. doi: 10.1145/2872427.2883062.

[51] Waseem, Z. Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter. In Proceedings of the Proceedings of the First Workshop on NLP and Computational Social Science; Association for Computational Linguistics: Austin, Texas, 2016; pp. 138–142. doi: 10.18653/v1/W16-5618.

[52] Singh, V.K., Huang, Q., Atrey, P.K. Cyberbullying detection using probabilistic socio-textual information fusion. In Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2016, pp. 884–887. doi: 10.1109/ASONAM.2016.7752342.

[53] Raisi, E., Huang, B. Cyberbullying Detection with Weakly Supervised Machine Learning. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2017, pp. 409–416.

[54] Wulczyn, E., Thain, N., Dixon, L. Ex Machina. Proceedings of the 26th International Conference on World Wide Web 2017. doi: 10.1145/3038912.3052591.

[55] Dani, H., Li, J., Liu, H., Sentiment Informed Cyberbullying Detection in Social Media; 2017; pp. 52–67. doi: 10.1007/978-3-319-71249-9_4.

[56] Chatzakou, D., Kourtellis, N., Blackburn, J., De Cristofaro, E., Stringhini, G., Vakali, A. Mean Birds. Proceedings of the 2017 ACM on Web Science Conference 2017. doi: 10.1145/3091478.3091487.

[57] Agrawal, S., Awekar, A. Deep Learning for Detecting Cyberbullying Across Multiple Social Media Platforms. Advances in Information Retrieval 2018, pp. 141–153. doi: 10.1007/978-3-319-76941-7_11.

[58] Huang, Q., Inkpen, D., Zhang, J., Van Bruwaene, D. Cyberbullying Intervention Based on Convolutional Neural Networks. In Proceedings of the Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018); Association for Computational Linguistics: Santa Fe, New Mexico, USA, 2018; pp. 42–51.

[59] Zhang, Z., Robinson, D., Tepper, J. Detecting hate speech on Twitter using a convolution-GRU based deep neural network. 2018.

[60] Rafiq, R.I., Hosseinmardi, H., Han, R., Lv, Q., Mishra, S. Scalable and Timely Detection of Cyberbullying in Online Social Networks. In Proceedings of the Proceedings of the 33rd Annual ACM Symposium on Applied Computing; Association for Computing Machinery: New York, NY, USA, 2018; SAC '18, pp. 1738–1747. doi: 10.1145/3167132.3167317.

[61] Rosa, H., Carvalho, J.P., Calado, P., Martins, B., Ribeiro, R., Coheur, L. Using Fuzzy Fingerprints for Cyberbullying Detection in Social Networks. In Proceedings of the 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2018, pp. 1–7. doi: 10.1109/FUZZ-IEEE.2018.8491557.

[62] Kao, H.T., Yan, S., Huang, D., Bartley, N., Hosseinmardi, H., Ferrara, E. Understanding Cyberbullying on Instagram and Ask.Fm via Social Role Detection. In Proceedings of the Companion Proceedings of The 2019 World Wide Web Conference; Association for Computing Machinery: New York, NY, USA, 2019; WWW '19, pp. 183–188. doi: 10.1145/3308560.3316505.

[63] Kumar, A., Nayak, S., Chandra, N., Empirical Analysis of Supervised Machine Learning Techniques for Cyberbullying Detection: Proceedings of ICICC 2018, Volume 2; 2019; pp. 223–230. doi: 10.1007/978-981-13-2354-6_24.

[64] Rosa, H., Matos, D., Ribeiro, R., Coheur, L., Carvalho, J.P. A "Deeper" Look at Detecting Cyberbullying in Social Networks. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1–8. doi: 10.1109/IJCNN.2018.8489211.

[65] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li and H. Vincent Poor, "Federated Learning for Internet of Things: A Comprehensive Survey," in IEEE Communications Surveys & Tutorials, vol. 23, no. 3, pp. 1622–1658, third quarter 2021, doi: 10.1109/COMST.2021.3075439.

[66] O. A. Wahab, A. Mourad, H. Otrok and T. Taleb, "Federated Machine Learning: Survey, Multi-Level Classification, Desirable Criteria and Future Directions in Communication and Networking Systems," in IEEE Communications Surveys & Tutorials, vol. 23, no. 2, pp. 1342–1397, Secondquarter 2021, doi: 10.1109/COMST.2021.3058573.

[67] K. Pillutla, S. M. Kakade and Z. Harchaoui, "Robust Aggregation for Federated Learning," in IEEE Transactions on Signal Processing, vol. 70, pp. 1142–1154, 2022, doi: 10.1109/TSP.2022.3153135.

[68] Y. Liu, X. Zhu, J. Wang and J. Xiao, "A Quantitative Metric for Privacy Leakage in Federated Learning," ICASSP 2021 – 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 2021, pp. 3065–3069, doi: 10.1109/ICASSP39728.2021.9413539.

[69] Xuefei Yin, Yanming Zhu, and Jiankun Hu. 2021. A Comprehensive Survey of Privacy-preserving Federated Learning: A Taxonomy, Review, and Future Directions. ACM Comput. Surv. 54, 6, Article 131 (July 2022), 36 pages. https://doi.org/10.1145/3460427.

[70] Q. Li et al., "A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection," in IEEE Transactions on Knowledge and Data Engineering, doi: 10.1109/TKDE.2021.3124599.

Biographies

Nisha P. Shetty has acquired her bachelor's and master's degree from Visvesvaraya Technological University. She is currently pursuing her doctorate at Manipal Institute of Technology, Manipal. She is working in the area of social network security.

Balachandra Muniyal's research area includes Network Security, Algorithms, and Operating systems. He has more than 30 publications in national and international conferences/journals. Currently, he is working as a Professor in the Dept. of Information & Communication Technology, Manipal Institute of Technology, Manipal. He has around 25 years of teaching experience in various Institutes.

Aman Priyanshu is a final-year undergraduate at the Manipal Institute of Technology. His research interests include Privacy Preserving Machine Learning, Explainable AI, Fairness, and AI for Social Good.

Vedant Rishi Das is currently pursuing his bachelor's degree in Computer Science and Engineering branch in Manipal Institute of Technology, Manipal. His areas of interests are Natural Language Processing and Machine Learning.