
PRIVACY-PRESERVING PHISHING EMAIL DETECTION BASED ON FEDERATED LEARNING AND LSTM

A PREPRINT

Yuwei Sun^{1,2,*}, Ng Chong³, and Hideya Ochiai¹

¹Graduate School of Information Science and Technology, The University of Tokyo

²Center for Advanced Intelligence Project, RIKEN

³Campus Computing Centre, United Nations University

*Corresponding author: Yuwei Sun, ywsun@g.ecc.u-tokyo.ac.jp

ABSTRACT

Phishing emails that appear legitimate lure people into clicking on the attached malicious links or documents. Increasingly more sophisticated phishing campaigns in recent years necessitate a more adaptive detection system other than traditional signature-based methods. In this regard, natural language processing (NLP) with deep neural networks (DNNs) is adopted for knowledge acquisition from a large number of emails. However, such sensitive daily communications containing personal information are difficult to collect on a server for centralized learning in real life due to escalating privacy concerns. To this end, we propose a decentralized phishing email detection method called the Federated Phish Bowl (FPB) leveraging federated learning and long short-term memory (LSTM). FPB allows common knowledge representation and sharing among different clients through the aggregation of trained models to safeguard the email security and privacy. A recent phishing email dataset was collected from an intergovernmental organization to train the model. Moreover, we evaluated the model performance based on various assumptions regarding the total client number and the level of data heterogeneity. The comprehensive experimental results suggest that FPB is robust to a continually increasing client number and various data heterogeneity levels, retaining a detection accuracy of 0.83 and protecting the privacy of sensitive email communications.

1 INTRODUCTION

The sharp uptick of phishing emails across the globe has exacerbated continued risks to personal data privacy and security in recent years. A recent report shows that phishing attacks soar 220% during the COVID-19 peak (Warburton, 2020). A phishing email usually adopts legitimate-look contexts to deceive users and steal sensitive information such as credit card numbers, bank accounts, and passwords. Phishing emails are prone to targeting a broad range of fields with highly crafted text data based on social engineering and users' personal online experiences. Despite existing detection mechanisms, phishing emails continue to slip past organizations' defenses. Notably, classical approaches such as signature-based detection can not work proactively due to ever-changing texts of phishing messages embedded with various types of hooks.

On the other hand, recent advancements in deep learning (DL) have made text mining of large numbers of phishing emails of various types possible. Knowledge acquisition and sentiment analysis based on natural language processing (NLP) offer a practical solution to the learning representation of text data with respect to the tone, grammatical coherence, emotion, and so forth, which significantly reduces human efforts on feature engineering. For this reason, it is considered that phishing emails' common features can be extracted and learned by a deep neural network (DNN), such as the long short-term memory (LSTM) model for phishing detection.

Despite this, training a DNN model usually needs vast data from both the phishing and legitimate categories. However, emails containing private data are difficult to collect on a server for centralized learning. The data being collected and processed by a model necessitates more consideration about privacy. Especially, with the promotion of legal restrictions

like GDPR (EU, 2016), it becomes more and more difficult and even unpractical to adopt centralized learning to operate on sensitive text data of emails. Moreover, such privacy concerns will refrain people from sharing their data, and a lack of training data can cause a trained model to fail during application. When the system encounters new samples unseen before, it will result in false positives, referring to legitimate emails incorrectly classified as phishing.

Our Contributions To this end, it seems more reasonable to leverage a decentralized learning architecture instead, such as the federated learning that allows a client to take full advantage of a DNN model while safeguarding the privacy of its emails. Furthermore, a decentralized learning architecture can improve the system’s adaptability to new samples by trained model sharing among clients with diverse data. In this research, we propose the Federated Phish Bowl (FPB), a federated learning-based phishing email detection method to tackle privacy-preserving model training on distributed email data based on the global word embedding and bidirectional-LSTM models. Moreover, we evaluated our method using the quarantined high-confidence phishing emails from Microsoft 365 in the organization. Through comprehensive experiments with various assumptions, the result suggests that the proposed method can improve a classifier’s performance in phishing email detection with privacy concerned and retain the performance when applying different client numbers and heterogeneity levels of client data.

Paper Outline This paper is organized as follows. Section 2 presents the related work of phishing email detection. Section 3 demonstrates the technical underpinnings of the proposed federated phishing email detection of FPB. Section 4 gives out detailed experiment settings and presents the evaluation results. Section 5 concludes the paper by discussing the societal implications, the limitations, and the future directions of this work.

2 RELATED WORK

Many methods have been proposed to safeguard email users from phishing, including traditional inspection and ML-based methods. On the one hand, traditional methods usually involve analysis of email formats, the integrity of email senders, and other attached meta information. This detection method is usually based on blacklists, heuristics, and visual analytics, which are neither feasible nor adaptive to real-life ever-changing phishing emails. On the other hand, recent years’ advancement in ML offers promising solutions by large-scale knowledge acquisition from email lexical information (see Table 1). For instance, Sahingoz et al. (2019) demonstrated a real-time anti-phishing system and compared its performance with seven different ML methods. They suggested that the random forest had the best performance for phishing detection. Gutierrez et al. (2018) presented a random under-sampling boost (Reboots)-based method to build a retrainable system adaptive to newly observed samples. Moreover, deep neural networks (DNNs) (LeCun et al., 2015) such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been adopted to improve detection by efficient text mining, alleviating efforts in feature engineering of domain experts. For example, Nguyen et al. (2018) proposed a hierarchical attentive long short-term memory (LSTM)-based detection method that models the email bodies at the word level and the sentence level while leveraging a supervised attention mechanism. Furthermore, Smadi et al. (2018) presented reinforcement learning-based detection to reflect changes in newly explored behaviors, thus detecting zero-day phishing attacks. Fang et al. (2019) demonstrated the THEMIS where emails were modeled at the email header and body and the character and word level simultaneously. They verified its effectiveness by evaluating an unbalanced dataset with realistic ratios of phishing and legitimate emails.

Centralized phishing email detection has been studied for a long time. Though the methods above provided an effective solution to email security protection based on ML and DNNs, the centralized data processing might violate the security and privacy of daily communications by emails. For this reason, it necessitates a privacy-preserving architecture for the phishing email detection task. Unfortunately, there are still not many studies that leverage decentralized learning in phishing email detection. Thapa et al. (2021) presented federated learning (FL)-based detection with language models of BERT (Devlin et al., 2019) and THEMIS (Fang et al., 2019) respectively. They evaluated various settings with different email user numbers and training data distributions. The result showed that with the increase of clients, there was a significant decrease in model performance.

In this research, we present the federated phish bowl (FPB) that adopts a combined approach of the global word embedding and bidirectional-LSTM neural networks to alleviate the influence of increasing clients on the global model’s convergence. Moreover, we collected a real-world phishing email dataset from an intergovernmental organization, which includes a total of 594 samples of the most recently observed and quarantined phishing emails. In addition, we performed a comprehensive comparison of model performance among three different detection methods, namely the proposed FPB, centralized learning, and standalone learning without model sharing.

Table 1: ML Methodologies for Phishing Email Detection

WORK	MODEL TOPOLOGY	METHODOLOGY
Gutierrez et al. (2018)	Centralized	Random under-sampling boost
M. et al. (2018)	Centralized	Deep neural networks
Nguyen et al. (2018)	Centralized	LSTM with an attention mechanism
Smadi et al. (2018)	Centralized	Reinforcement learning
Sahingoz et al. (2019)	Centralized	Seven machine learning methods
Fang et al. (2019)	Centralized	Recurrent convolutional neural networks
Alhogail and Alsabih (2021)	Centralized	Graph convolutional networks
Thapa et al. (2021)	Decentralized	THEMIS and BERT

3 FEDERATED PHISHING EMAIL DETECTION

3.1 Data Privacy and Decentralized Deep Learning

In recent years, data privacy has become a major concern attracting attention from many walks of society, exacerbated by notorious issues such as the Cambridge Analytica scandal and FBI-Apple encryption dispute. The growing public awareness of data privacy and legal restrictions such as the General Data Protection Regulation (GDPR) have rendered the centralized processing of sensitive data in machine learning more and more difficult and even impractical, due to privacy interests in controlling how sensitive information is used.

To process and acquire knowledge from distributed data, there are two possible topologies of deep learning (DL) models, i.e., centralized deep learning (CDL) and decentralized deep learning (DDL). CDL leverages central high-performance computing (HPC) to perform large-scale training on data collected from diverse sources. However, the centralized architecture can result in critical data breaches due to single-point failures caused by adversarial attacks. In contrast, DDL is considered privacy-preserving thus facilitating distributed model training on edge devices like smartphones. Federated learning (FL) was proposed by Google to improve Google Keyboard’s performance in next word prediction (Konečný et al., 2016). FL is aimed to take full advantage of deep neural networks while safeguarding the privacy of local training data through model sharing and aggregation.

To this end, we propose the Federated Phish Bowl (FPB), a privacy-preserving and adaptive phishing email detection system. Emails containing personal information are difficult to collect on a server for centralized learning in real-life applications due to escalating privacy concerns. FPB leverages FL to allow common knowledge representation and sharing among clients about phishing, thus improving the system’s performance and adaptability.

3.2 Email Text Mining with Natural Language Processing

To analyze text sequence data like emails, natural language processing (NLP) acquires meaningful knowledge from the contexts of emails revealing hidden patterns in corpora. First, to preprocess an email, we extract information from both the header and body of the email. In particular, we extract the subject line from the header. For the body, we extract lexical information with the beautiful soup (Richardson, 2007) applied to parse any embedded HTML page. Then we concatenate the subject and the body text of an email as the input of a preprocessing pipeline (Fig. 1), which removes irrelevant information such as prepositions and assigns weights to important words in the input string. In detail, the pipeline performs the following tasks: removal of non-letter characters, conversion of characters into lowercase, tokenization, removal of stop words, lemmatization, removal of words with less than two characters, conversion of tokens back to strings, and mapping strings to feature vectors.

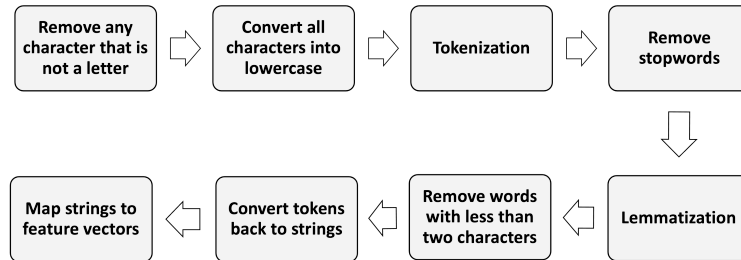


Figure 1: The pipeline for text mining of email data.

First, we remove characters that are not letters such as numbers and punctuation marks and further convert the remained characters to lowercase. Then, tokenization that splits a sentence into small chunks of words is used to extract tokens from the input strings. After that, we remove stop words such as “the”, “a”, and “is”, and convert words with various tenses and plurals into their base or dictionary forms based on lemmatization. Moreover, after the above steps, any word with less than two characters is removed because it is usually an incomplete part of a word without specific meaning. Finally, we combine these cleaned and standardized tokens back into a continual string resulting from the lexical analysis (Fig. 2).

Text Sequence	Lexical Analysis Result
CONFIRM YOUR DELIVERY DETAILS\r	confirm delivery detail
1st Reminder - Invitation to 2021 UI GreenMetr...	reminder invitation world univers
Purchase order537893\r	purchase order
Password Notification\r Notification For Your ...	password notification notification passcode

Figure 2: A sample result of the lexical analysis.

Furthermore, inputting the email data into a DL model for training necessitates representing these extracted strings with numerical feature vectors. To this end, an embedding layer is used to learn a mapping from the input strings to vector representations during the training. In other cases, the embedding layer can also be learned and transferred from elsewhere. Moreover, decentralized model training requires knowledge sharing of clients on skewed local training data. For this reason, instead of each client training an individual embedding layer for word representation, we adopt a global word embedding strategy by leveraging a pre-trained embedding model called GloVe (Pennington et al., 2014). The GloVe model is trained on a word-to-word co-occurrence matrix which tabulates how frequently words co-occur with one another in a given corpus. Notably, we employ the trained word embedding model based on six billion tokens from Wikipedia 2014 and Gigaword 5, with a 100-item feature vector output for each input word.

3.3 Phishing Email Detection Based on Bidirectional Long Short-Term Memory

Nowadays, recurrent neural networks (RNNs) have been applied in a broad field of speech recognition (Weninger et al., 2015), language translation (Wu et al., 2016), network intrusion detection (Zhou et al., 2021), and so forth. The characteristic of RNNs is automatically extracting hidden features from the input sequence data and classifying these features in a high-dimensional space.

To classify emails, we adopt an RNNs model called bidirectional long short-term memory (bidirectional-LSTM)(Schuster and Paliwal, 1997), which processes the input sequence from both the positive direction and the negative direction thus improving its learning performance. An output of the bidirectional-LSTM concatenates the results from the two directions. Moreover, an LSTM cell for forming the neural networks remembers long-term word associations leveraging three gates: the forget gate, the input gate, and the output gate. In particular, the forget gate determines which relevant information from the previous steps is retained. The input gate decides which relevant information is added from the current step, and the output gate produces the state sent to the next step.

We employ a five-layer neural networks model consisting of three bidirectional-LSTM layers and two fully connected layers for each client’s model training (Fig. 3). In detail, the input layer of the neural networks has a total of 200 time-steps, with each step having 100 embedded features. Then, it is followed by three bidirectional-LSTM layers with 100 memory units each. After that, a fully connected layer with 200 neurons is applied using as an activation function the ReLU. Finally, the output layer has a single neuron predicting the type of an input email using as an activation function the Sigmoid. In addition, legitimate emails are assigned a label of 0, and phishing emails are assigned a label of 1.

3.3.1 Federated Phishing Email Detection

Two models work simultaneously on the FPB server, where one is used for training, and the other is for real-time inferring. Then, for every several iterations of training, the retrained model will replace the current inference model. The two merits of this architecture are privacy-preserving data processing where original texts in emails are considered impossible to be reconstructed from shared model weights and a better model performance using model aggregation.

We assume an N-client scenario, where each client has the same number of well-balanced training data. In other words, data owned by each client are independent and identically distributed (IID). The discussion on model training with non-IID data is demonstrated in Section 4.2.2. Moreover, we adopt the FedAvg (Konečný et al., 2016) defined in (1), where a subset of clients are randomly selected to participate in each round’s model training and the weighted averaging of their local updates is used to update the current global model.

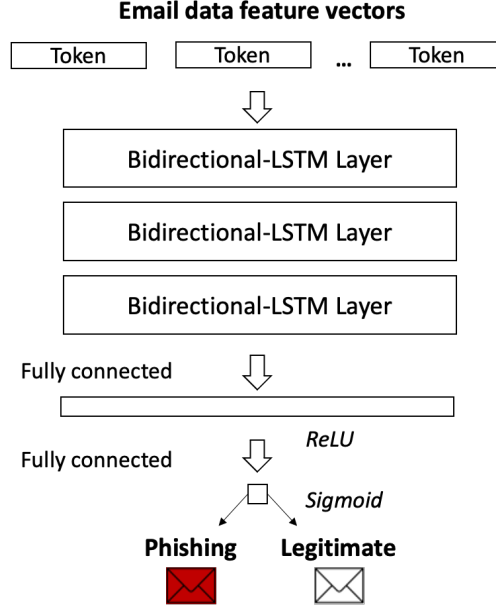


Figure 3: The architecture of a client's local model.

$$w_{t+1} = w_t + \sum_{i \in S} \frac{n_i}{n_S} (w_t^i - w_t) \quad (1)$$

Where w_{t+1} is the updated global model, w_t is the current global model, w_t^i is local update from client i , S is an union of the selected clients, and $\frac{n_i}{n_S}$ represents the ratio of client i 's local training data with respect to the training data of all selected clients.

Furthermore, to train an FPB model using the federated learning (FL), the parameter server (PS) initializes a global DL model based on the aforementioned bidirectional-LSTM neural networks and sends the global model with a global word embedding matrix to all clients at the first round of training (Fig. 4). Subsequently, every round, the PS randomly selects a subset of clients for local model training where a client updates a local DL model based on its training data. After training, a local model update is sent back to the PS to update the current global model using the weighted averaging aggregation. The updated global model will be used for the next round's training. As such, by sharing the global word embedding matrix and local model updates, the system achieves a better and better model for phishing email detection without revealing a client's sensitive email data.

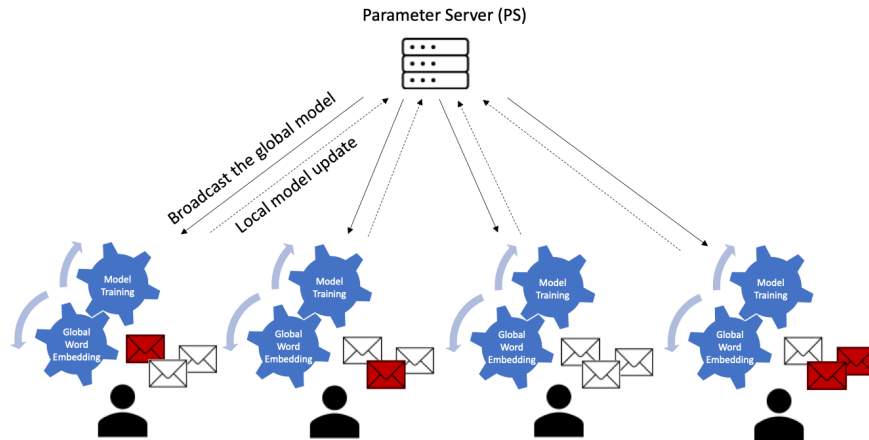


Figure 4: Phishing email detection based on global word embedding and federated learning.

4 EVALUATION

4.1 Dataset

We collected quarantined high-confidence phishing emails from Microsoft 365 (Microsoft, 2021), including a total of 678 phishing emails from the organization. Then, using the aforementioned lexical analysis and global word embedding, we extracted essential text information from these phishing emails and transformed them into feature vectors. Moreover, to regulate the input sequence, we set 200 as the maximum feature vector length, a feature vector with a length greater than 200 being truncated. Whereas, a vector with a length smaller than 200 was post-padded with zeros. In addition, any feature vector that had a length smaller than 10 was ignored. Finally, after removing all the under-length emails and duplicates, a total of 594 phishing email samples were retained for the dataset. To balance data in the dataset, we adopted legitimate email samples from the Enron dataset (Klimt and Yang, 2004), randomly selecting a total of 594 samples. Consequently, we obtained a dataset consisting of 1188 emails; each of which was represented with a 200-item feature vector. We further divided the dataset into a training set with 1069 samples and a validation set with 119 samples (a ratio of 0.9 : 0.1).

4.2 Numerical Results

4.2.1 Baseline Performances

Centralized Learning We performed centralized learning of the bidirectional-LSTM model on the training set as a baseline. Its performance was evaluated at the end of each round based on the validation set. The early stopping was employed to monitor the variance in validation loss with a patience value of 10, thus automatically stopping the training when there appeared no decrease of the validation loss over the last 10 epochs. Moreover, we applied as a learning function the Adam with a learning rate of 0.0001 and a batch size of 16. Finally, the model converged after 82 epochs' training, achieving an average validation accuracy of 0.86 over the last five epochs. Figure 5 illustrates the training progress of the centralized learning method based on the accuracy and loss.

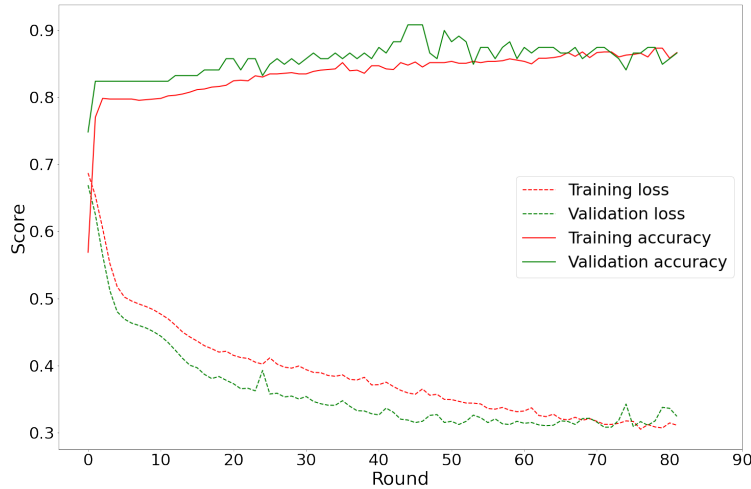


Figure 5: Model performance based on the centralized learning.

Standalone Learning It refers to a client training its local model purely on the local dataset, without sharing the trained model with others. In this case, we studied the standalone learning of each client in an FPB system, with the same hyperparameter and training setting as in the centralization learning. For each trained local model, we applied the validation set to evaluate its performance.

4.2.2 Performance of the Federated Phish Bowl

We studied a 10-client scenario with an IID setting. In this case, the training set was evenly divided into 10 subsets with the same number of phishing and legitimate emails. Then, for each round, a subset of three clients was randomly selected to conduct the local model training based on their own data. Notably, the local model training adopted the Adam with a learning rate of 0.0001, a batch size of 16, and an epoch of one. Moreover, we evaluated the updated

global model’s performance using the test set of the dataset at the end of each round. At last, we conducted the federated phishing email detection for 50 rounds and compared its performance with the baseline methods. Especially for the centralized learning and the federated learning, we performed 10 individual trials each. For the standalone learning, we evaluated the performance of a local model trained on the 10 clients’ local data respectively. Figure 6 illustrates the comprehensive comparison between the three methods against the average validation accuracy metric over the last five rounds. The result shows that FPB outperforms a client’s standalone learning. Though it appeared a slightly decreased performance due to the model aggregation, compared with the centralized method, it still achieved a competing performance while safeguarding the privacy of a client’s email data.

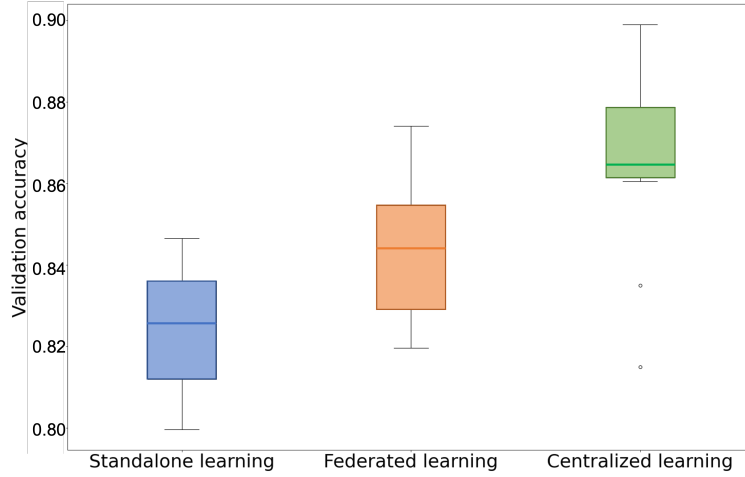


Figure 6: Comparison of validation accuracy between the centralized learning, federated learning, and standalone learning for phishing email detection.

Number of Participating Clients With a continually increasing client number, the performance of the system may greatly decrease due to a model aggregation with more local updates involved. For this reason, we studied the system’s performance when applying various client numbers of 10, 20, and 50 respectively. To ensure the same amount of data was used for each case and the only variable was the client number, we randomly selected 3, 6, and 15 clients accordingly for each round’s model training. Then, we performed model training with the IID setting for a total of 50 rounds and evaluated the global model’s validation accuracy as the metric. Figure 7 illustrates the global model’s performance at each round of the model training with 10, 20, and 50 clients respectively.

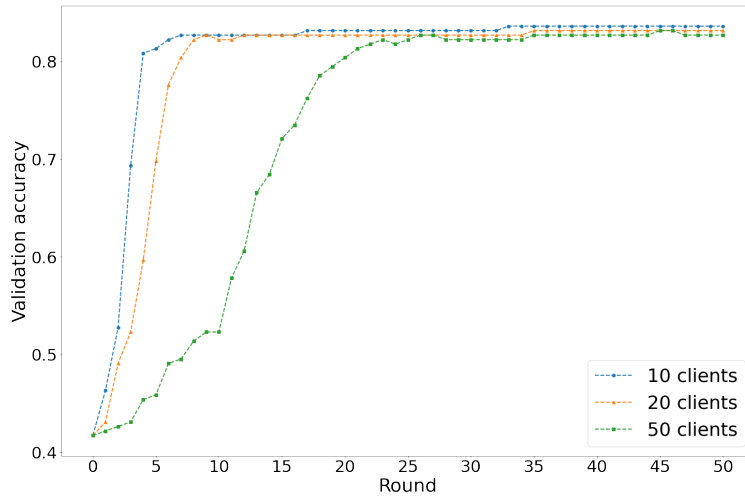


Figure 7: Test accuracy of the global model at each round with different client numbers.

Data Heterogeneity In real-life applications, samples held by clients are typically skewed with different data sizes and distributions, i.e., non-IID. Such heterogeneous data distributions of clients usually result in a time-consuming convergence of the global model and divergence of the training algorithm (Sun et al., 2021). Mathematically, we use α to denote different levels of non-IID data in (2): $\alpha = 1.0$ indicating that data of each client only belong to one label, $\alpha = 0.6$ indicating that 80% of the data belong to one label and the remaining 20% data belong to the other label, $\alpha = 0.2$ indicating that 60% of the data belong to one label and the remaining 40% data belong to the other label, and $\alpha = 0$ indicating that data of each client is IID. In particular, we studied the heterogeneity cases of 0, 0.2, 0.6, and 1.0 in the three aforementioned scenarios with a client number of 10, 20, and 50 respectively. Furthermore, we assumed that half of the clients owned more phishing emails in their local training data whereas the other half owned more legitimate emails. Figure 8 illustrates the global model’s performance at each training round in the 10, 20, and 50 clients scenarios when applying various heterogeneity levels of client data. Consequently, the result suggests that FPB is robust to the continually increasing client number and various data heterogeneity levels, retaining a detection accuracy of 0.83 and protecting the privacy of sensitive email communications. Whereas, when applying the heterogeneity level of 1.0 in the 50-client scenario, the model failed to converge within the 50 rounds. Despite this, its performance appeared to be gradually increasing.

$$\alpha = |2P_i - 1| \quad (2)$$

Where P_i represents the possibility for an email sample of the client i belongs to the phishing and α takes the absolute value of the computed result.

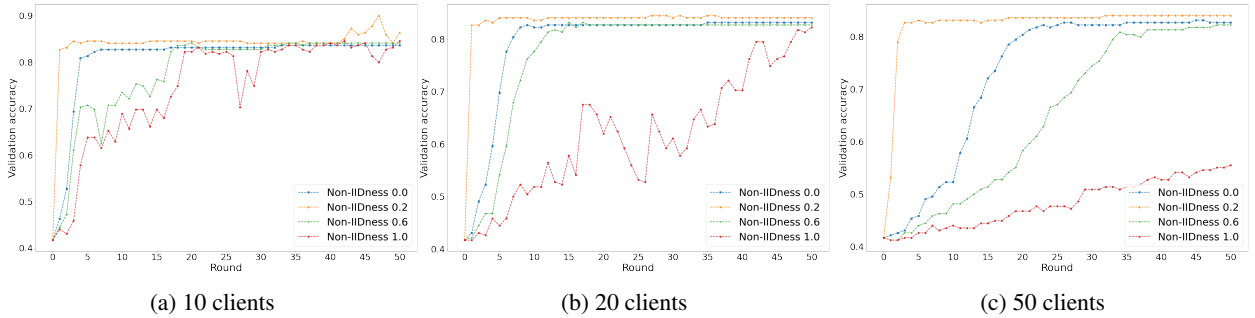


Figure 8: Performance evaluation for the different levels of non-IIDness.

5 CONCLUDING REMARKS

Societal Impact The adaptability of a phishing email detection system is largely restricted by accessible training data. Unfortunately, due to recent years’ escalating privacy concerns and heavily promoted data protection regulations, centralized data processing reveals underlying risks of data exposure and sensitive information leak. We proposed the Federated Phish Bowl (FPB) for tackling adaptive phishing email detection based on sensitive email data, leveraging natural language processing and bidirectional-LSTM. Based on the comprehensive evaluation, FPB shows a respectable match with the centralized approach while safeguarding the privacy of email communications, with great robustness to the increasing client number and various data heterogeneity levels. In addition, such a decentralized framework might result in increasing adversarial attacks at the edge, for instance, backdoor attacks (Bagdasaryan et al., 2020).

Limitations The model we considered so far was based on synchronous federated learning (FL), where the aggregation was performed after receiving all local updates. However, in some cases, conditions such as limited network bandwidth, data volume, and computing device constraints become the bottleneck of training. In contrast, the asynchronous FL might offer a better solution to the client scheduling problem, which allows a client to upload the local update at any stage of the training (Khan et al., 2020; Lu et al., 2020).

References

- David Warburton. Phishing attacks soar 220% during covid-19 peak as cybercriminal opportunism intensifies. <https://www.f5.com/labs/articles/threat-intelligence/2020-phishing-and-fraud-report>, 2020. Accessed: 2021-09-17.
- EU. General data protection regulation. <https://gdpr-info.eu>, 2016. Accessed: 2021-09-13.

- Ozgur Koray Sahingoz, Ebubekir Buber, Onder Demir, and Banu Diri. Machine learning based phishing detection from urls. *Expert Systems with Applications*, 117:345–357, 2019. ISSN 0957-4174.
- Christopher N. Gutierrez, Taegyu Kim, Raffaele Della Corte, Jeffrey Avery, Dan Goldwasser, Marcello Cinque, and Saurabh Bagchi. Learning from the ones that got away: Detecting new forms of phishing attacks. *IEEE Transactions on Dependable and Secure Computing*, 15(6):988–1001, 2018.
- Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. ISSN 1476-4687.
- Minh Nguyen, Toan Nguyen, and Thien Huu Nguyen. A deep learning model with hierarchical lstms and supervised attention for anti-phishing, 2018.
- Sami Smadi, Nauman Aslam, and Li Zhang. Detection of online phishing email using dynamic evolving neural network based on reinforcement learning. *Decision Support Systems*, 107:88–102, 2018. ISSN 0167-9236.
- Yong Fang, Cheng Zhang, Cheng Huang, Liang Liu, and Yue Yang. Phishing email detection using improved rcnn model with multilevel vectors and attention mechanism. *IEEE Access*, 7:56329–56340, 2019.
- Chandra Thapa, Jun Wen Tang, Alsharif Abuadba, Yansong Gao, Seyit Camtepe, Surya Nepal, Mahathir Almashor, and Yifeng Zheng. Evaluation of federated learning in phishing email detection, 2021.
- J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.
- Hiransha M., Nidhin Unnithan, Vinayakumar Ravi, and Soman K., P. Deep learning based phishing e-mail detection cen-deepsam. *Proceedings of the 1st Anti-Phishing Shared Task Pilot at 4th ACM IWSPA co-located with 8th ACM Conference on Data and Application Security and Privacy*, 2018.
- Areej Alhogail and Afrah Alsabih. Applying machine learning and natural language processing to detect phishing email. *Computers and Security*, 110:102414, 2021. ISSN 0167-4048.
- Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtarik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. In *NIPS Workshop on Private Multi-Party Machine Learning*, 2016.
- Leonard Richardson. Beautiful soup documentation. *April*, 2007.
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.
- Felix Weninger, Hakan Erdogan, Shinji Watanabe, Emmanuel Vincent, Jonathan Le Roux, John R. Hershey, and Björn W. Schuller. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR. In *Latent Variable Analysis and Signal Separation - 12th International Conference, LVA/ICA 2015, Liberec, Czech Republic, August 25-28, 2015, Proceedings*, volume 9237 of *Lecture Notes in Computer Science*, pages 91–99. Springer, 2015.
- Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google’s neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.
- Xiaokang Zhou, Yiyong Hu, Wei Liang, Jianhua Ma, and Qun Jin. Variational lstm enhanced anomaly detection for industrial big data. *IEEE Transactions on Industrial Informatics*, 17(5):3469–3477, 2021. doi:10.1109/TII.2020.3022432.
- Mike Schuster and Kuldip K. Paliwal. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.*, 45(11): 2673–2681, 1997.
- Microsoft. Anti-phishing policies in microsoft 365. <https://docs.microsoft.com/en-us/microsoft-365/security/office-365-security/set-up-anti-phishing-policies?view=o365-worldwide>, 2021. Accessed: 2021-09-17.
- Bryan Klimt and Yiming Yang. The enron corpus: A new dataset for email classification research. In *Proceedings of the 15th European Conference on Machine Learning, ECML’04*, page 217–226, Berlin, Heidelberg, 2004. Springer-Verlag. ISBN 3540231056.
- Yuwei Sun, Hideya Ochiai, and Hiroshi Esaki. Decentralized deep learning for mobile edge computing: A survey on communication efficiency and trustworthiness, 2021.

- Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 2938–2948. PMLR, 2020. URL <http://proceedings.mlr.press/v108/bagdasaryan20a.html>.
- Latif U. Khan, Shashi Raj Pandey, Nguyen H. Tran, Walid Saad, Zhu Han, Minh N. H. Nguyen, and Choong Seon Hong. Federated learning for edge networks: Resource optimization and incentive mechanism. *IEEE Communications Magazine*, 58(10):88–93, 2020.
- Yunlong Lu, Xiaohong Huang, Yueyue Dai, Sabita Maharjan, and Yan Zhang. Differentially private asynchronous federated learning for mobile edge computing in urban informatics. *IEEE Trans. Ind. Informatics*, 16(3):2134–2143, 2020.