Modeling Without Sharing Privacy: Federated Neural Machine Translation

Jianzong Wang, Zhangcheng Huang, Lingwei Kong(✉), Denghao Li, and Jing Xiao

Ping An Technology (Shenzhen) Co., Ltd., Shenzhen, China
{wangjianzong347,huangzhangcheng624,konglingwei630,lidenghao842, xiaojing661}@pingan.com.cn

Abstract. Training neural machine translation models requires large amount of diverse training corpora. It poses a challenge for collecting sufficient data. In addition, labeling monolingual corpus demands professional knowledge in certain domain. Building collaboration between different institutes produces other problems such as legality of data exchange and commercial data leakage.

In this paper, we proposed a federated neural machine translation model *FedNMT* to train a robust machine translation system without sharing raw data from participants. By applying *FedNMT*, neural machine translation (NMT) systems can be ameliorated from the corpus held by different contributors without directly exposing them to one another. This approach preserves the user privacy by utilizing the federated learning framework, encryption techniques. In the federated learning paradigms, a global model is distributed to user clients, and a central server is built to aggregate the learning parameters and update the gradients. Experimental results show the effectiveness of our model in comparison with the data-centralized model.

Keywords: Machine translation · Federated learning · Privacy preserving

1 Introduction

The achievements of Neural Machine Translation [2,7,16] have drawn the attention of the professionals ever since its appearance. The training resource - a parallel corpus is considered as a key component for modeling faithfulness and fluency. However, researchers are rarely aware of parallel corpus privacy. In certain fields like health care, finance, and engineering, there are a few available data resources to be exploited. Exchanging training corpora obtained by individual institutions is prohibited due to data security policy and concerns.

There are some previous research focused on the privacy of NLP, such as federated learning in language modeling [5] and the privacy in NMT [8]. Unarguably, it is necessary to implement research on neural machine translation with respect to data privacy protection. To alleviate the problem of lack of data

W. Zhang et al. (Eds.): WISE 2021, LNCS 13080, pp. 216–223, 2021.
https://doi.org/10.1007/978-3-030-90888-1_18

source, previous research [9,17], proposed to exploit the monolingual corpora to import the model faithfulness and fluency. However, they are based on custom training corpora or general-domain publicly available corpora. Except for a few areas, data storage mechanisms are loosely regulated. In particular, transferring private data between hosts elevates the possibility of data breaches or the communication process to be hacked. Moreover, it also leads to inefficiency. Federated learning framework [4] has been developed to transmit encrypted intermediate model parameters between parties without sharing local data.

To enhance the performance of individual agents and build shared vocabularies without sharing training data, this paper proposes a federated neural machine translation model with exploited and jointly trained corpus held by different institutes under a data privacy-preserving scheme. Since each participants receive an identical copy of a *FedNMT* model, the training on particular model can continue locally using a new set of training corpus. We implement a series of experiments to demonstrate the feasibility of the privacy-preserving scheme. This paper has the following contributions:

- Present the first privacy preserving model in machine translation with high quality NMT model without sharing private data to our knowledge.
- We propose a privacy-preserving vocabulary generation method.
- Demonstrate competitive performance with data-centralised non-encrypted NMT methods.

2 Proposed Method

2.1 Problem Definition

In the setting of federated neural machine translation, the architecture is constructed by a server and N parties. Each party holds a private corpus $D_n = \{(x_i, y_i)\}_{i=1}^{K^{D_n}}$. When collaboratively training a deep neural network (DNN) model, the objective function of global model is formulated as

$$\min_{\Theta_G} \mathcal{L}(\underbrace{\sum_{n=1}^{N} D_n}_{locally fixed}; \Theta_G) = \min_{\Theta_n} \sum_{n=1}^{N} \frac{1}{N} \mathcal{L}(\underbrace{D_n}_{locally fixed}; \Theta_n) \tag{1}$$

where Θ_G is the weights of global model and Θ_n is the weights of private model. Global model weights Θ_G benefit from all the training corpus $\sum_{n=1}^{N} D_n$. The goal is to fit the model weights Θ_G effectively and achieve better model performance without sharing the private data. In this paper, we design a framework to train the NMT model and verify the effectiveness of our method.

2.2 Architecture of FedNMT

The proposed *FedNMT* framework is shown in Fig. 1, where S_A, T_A, S_B and T_B are the source and target sentences of Party A and Party B respectively. Each of

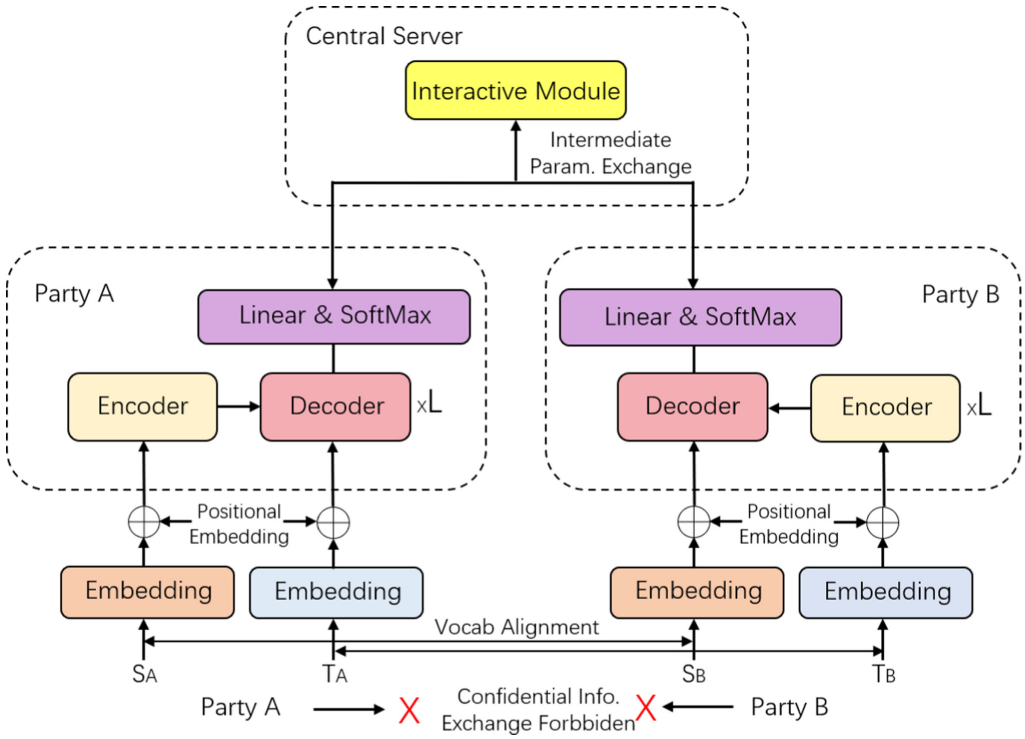

Fig. 1. Illustration of proposed FedNMT framework.

the party has similar layers where two of the embedding layers is for converting original data into the vectors, followed by N layers of encoders and decoders. Thus, each party uses self-attention mechanism to obtain the high-dimensional vectors that represents the semantic relationship between the source language and target language. The process of word embedding is referenced by a federated vocabulary which will be described in Sect. 2.3. The local model conforms with a transformer architecture, where the depth of encoder-decoder is L. At each step, source and target languages are embedded by each party, the semantic relationship between words is computed by the self-attention module which integrates into the encoder-decoder module. The server contains an interactive module which requires difference of model weights $\triangle\Theta_n = \Theta_n^{(N^{(local),t})} - \Theta_n^{(N^{(local),t-1})}$ and momentum variable β as inputs. We adopt the momentum restart mechanism [10] at each aggregation step. The output of interactive module is defined as

$$Z(\Theta_G^t, \beta^t) = Z(\Theta_G^{t-1}, \beta^{t-1}) + \triangle Z^{t-1} \tag{2}$$

$$\triangle Z^{t-1} = \frac{1}{\sum N_n^{local}} \sum N_n^{local} \cdot (\triangle\Theta_n, \triangle\Theta_\beta) \tag{3}$$

where N_n^{local} is the local training steps of Party n. After the parties receive updated $Z(\Theta_G^t, \beta^t)$, they re-initialize the momentum variables in local model and train with a new Θ_G^t.

2.3 Federated Vocabulary

The embedding process in Transformer is a combination of word embedding and positional embedding which requires a joint vocabulary to generate a vector representation of input language vocabulary pairs in the same space domain. This process refers to the vocabulary alignment process [5] in Fig. 1. In the step of building the federated vocabulary, we employed the method of Paillier homomorphic encryption [13] which allows secure computation over encrypted data. Given $\{[[E_1]], \ldots, [[E_i]]\}$ as the cipher-text and $\{k1, \ldots, ki\}$ as the scalar constants, homomorphic encryption supports the calculation of $(k1 \otimes [[E_1]]) \oplus \ldots \oplus (ki \otimes [[E_i]])$.

We proposes federated vocabulary alignment, which denotes parallel corpus owned by P_A and P_B, respectively. Firstly, party A and party B introduce local vocabularies V_A and V_B independently. Then, all local vocabularies are encrypted and denoted as $[[V_A]]$ and $[[V_B]]$. The federated vocabulary only provides order of words based on the total frequency of clients, and statistical word frequency is not transmitted from the server.

2.4 Secure Model Training

The scheme of Federated NMT model training is illustrated in Algorithm 1. For each training iteration N^{local}, the parties send the updated model parameters $\triangle\Theta^{local}$ to server which is responsible for aggregating the local parameters using *FedAVG* [12] and return back the updated model parameters.

The intermediate parameters exchange between central server and parties is confidential. At each round of federated aggregation, server only transmit the gains of parameters denoted as $\triangle\Theta^{local}$ between every two federated aggregation steps. To protect the privacy of the parties, the process of parameter updating is carried out under differential privacy [1,6], the transmission of model parameters at each federated step is accompanied with an additional noise $Lap(\frac{\sigma}{\epsilon})$, where σ is the sensitivity constant and ϵ is the privacy threshold. Thus, server only receives an encrypted value $\triangle\Theta^{local} + Lap(\frac{\sigma}{\epsilon})$ from the parties.

2.5 Domain Expert

In a real-world setting, the parties participating in a federated translation task may come from various domains. These conditions lead to a noise injection from the other parties. Moreover, the differential privacy updating policy may degrade the model performance. To alleviate this condition, we introduce the domain expert method inspired by [3,15].

Instead of using the global model to translate an input sentence directly, a private model is trained synchronously. The difference between a data-centralised training and our method is that the vocabulary is replaced with the federated vocabulary built in the federated training process in the federated learning scheme. Hence, for each party, the final output is a combination of joint training model and a private model with domain adaption. Let M_G be the global model trained by the federated learning and M_P be the private

Algorithm 1. Method of privacy-preserving model training

Require: Parallel Corpus D_A=$\{x_i,y_i\}^N$ and D_B=$\{x_j,y_i\}^N$ held by separate parties, Learning rate η, Proportion of model parameters to share ϖ, Noisy threshold ϵ.

procedure: Server initializes a global model Θ^{global}
Start the federated training with global step T
for t $\leftarrow$ 1 to T **do**
  Server distribute the updated parameters to parties
  for i $\leftarrow$ 1 to N^{local} **do**
    Training with a batch of corpus D_{local}
    Update $\Theta^{(i,t)}$
  end for
  Compute difference: $\triangle\Theta^{(local,t)} = \Theta^{(N^{local},t)}$ - $\Theta^{(N^{local},t-1)}$
  Add Laplacian Noise $Lap(\frac{\sigma}{\epsilon})$
  Privacy preserving transmission: $\triangle\Theta^{(local,t)} + Lap(\frac{\sigma}{\epsilon})$
  Privacy preserving federated aggregation
end for
end procedure

model optimized in a standard way for a specific domain. The final prediction $\hat{y} = \lambda(x)M_G(\Theta_G) + (1 - \lambda(x))M_P(\Theta_P)$, where Θ_G and Θ_P denote the model parameters of global and private model respectively. $\lambda(x)$ is a gating function following the Mixture of Experts(MoE) architecture. We set this gating function as $\lambda(x) = tanh(\theta \cdot x)$. The MoE architecture determines the influence ratio between the global model and private model in individual translation cases.

3 Experiment

3.1 Experiment Setups

The experiments are implemented on English-Chinese and English-German translation. The training datasets for task En-De are Europarl V9[1] and News Commentary v14[2] and newstests 2015 as the test set. The training dataset for task En-Zh is from CWMT [3], NEU2017 held by one of the participants and Casis2015 by the other wand newstests 2017 as test set. We randomly drew 2k samples from the training dataset as a validation set. Word segmentation is used for Chinese sentences with an open sourced tool called THULAC[4] [11]. After preprocessing, the resulting training corpus includes 2.07M and 3.5M language pairs for En-De and EN-Zh tasks respectively. In En-De task, Party A contains 1.75M sentence pairs and Party B contains 320K sentence pairs. In Zh-En task,

[1] http://www.statmt.org/europarl/v9/training/europarl-v9.de-en.tsv.gz.
[2] http://data.statmt.org/news-commentary/v14/news-commentary-v14.tsv.gz.
[3] http://mteval.cipsc.org.cn:81/agreement/wmt.
[4] https://github.com/thunlp/THULAC-Python.

Party A and Party B consists 1.8M and 1.7M sentence pairs. The language tool for evaluation is uncased 4-gram BLEU [14].

The experiment depth of encoder and decoder is 6. The amount of attention heads is 8 and the hidden embedding size is 512. The filter size for feed forward layer is 2048. A learning rate decay policy is also applied along with 4000 warm-up steps. In decoding, we set beam size to 6 and a length normalization weight of 1.5. The language tool for evaluation is uncased 4-gram BLEU [14]. *FedNMT* is the FL model without weighted average. *FedNMT+WA* denotes the model with weighted average at each federated aggregation step. We also conducted two experiments *FedNMT+DE* and *FedNMT+WA+DE* to augment the FL model with domain expert. In the privacy-preserving setup, we also train the *FedNMT+WA* model with low noise $\epsilon = 1$ and high noise $\epsilon = 2$. The baseline systems are based on Transformer[5]. All models are trained more than 10 epochs to ensure convergence and trained on two NVIDIA Tesla V100 GPUs.

Table 1. BLEU scores on English-German and English-Chinese translation.

System		En-De		En-Zh	
		Dev	Test	Dev	Test
Baseline system	$\text{Transformer}_{PartyA}$	19.44	19.81	14.28	14.13
	$\text{Transformer}_{PartyB}$	19.20	18.23	11.57	10.91
	+Data-centralised training	21.07	21.84	16.05	**16.53**
FedNMT system	FedNMT	20.72	20.91	16.08	16.02
	FedNMT+WA	20.79	21.12	16.52	16.21
FedNMT with Domain Expert	$\text{FedNMT+DE}_{PartyA}$	20.82	22.06	16.27	15.92
	$\text{FedNMT+DE}_{PartyB}$	20.38	21.35	16.54	16.43
	$\text{FedNMT+WA+DE}_{PartyA}$	**21.64**	**22.43**	**16.85**	16.39
	$\text{FedNMT+WA+DE}_{PartyB}$	20.99	21.45	16.50	16.56

3.2 Performance

The training loss is illustrated in Fig. 2a and Fig. 2b, which plot training loss by number of epochs for different experiment setups. The evaluation results of the experiment are shown in Table 1, the precision loss of FedNMT is small compared with the local baseline model. The comparison of BLEU score with different noise level is demonstrated in Table 2. The BLEU score of *FedNMT+WA* is better than the simple *FedNMT* system, while the model parameters of *FedNMT+WA* is adopted with weighted average scheme. From the result of FL system adopted with domain expert, it shows that the final outputs is partly determined by the private model that the individual party trained. The training loss with different noise constant ϵ is shown in Fig. 2c and Fig. 2d. As anticipated, there is a trade-off between model accuracy and privacy protection. Increment of ϵ indicates a higher level of user privacy protection, where the value determines the amount of noise added to the transmission.

[5] https://github.com/Kyubyong/transformer.

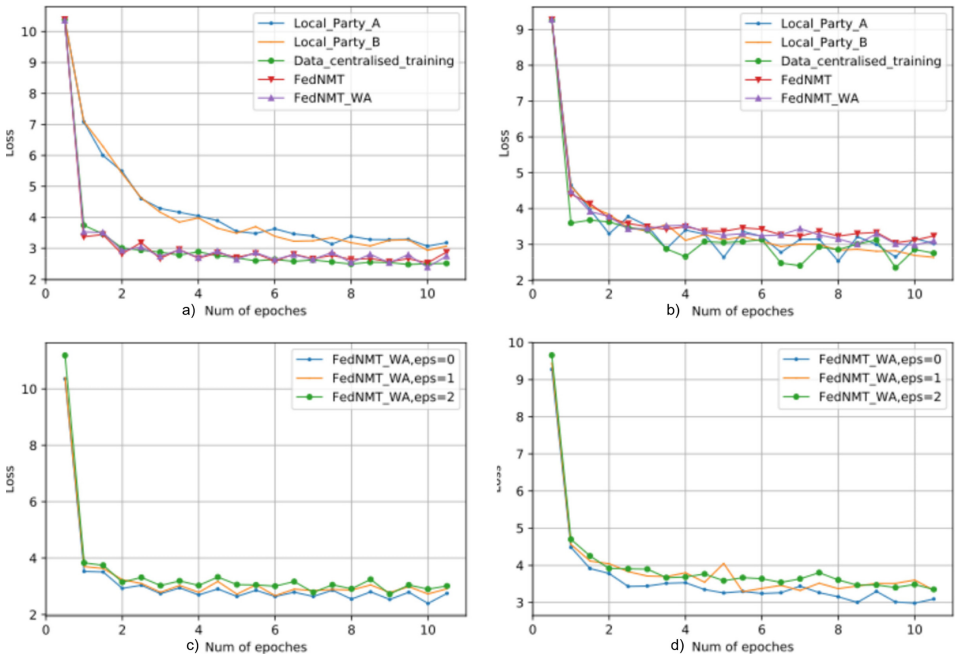

Fig. 2. The training loss of a) En-De, b) En-Zh task on test set: FL vs. non-FL and comparison of c) En-De, d) En-Zh training loss on test set with different noise level.

Table 2. FedNMT+WA (F.WA) Model performance by varing the noise level.

System	En-De		En-Zh	
	Dev	Test	Dev	Test
F.WA, $\epsilon = 0$	20.79	21.12	16.52	16.21
F.WA, $\epsilon = 1$	20.51	20.93	16.47	16.05
F.WA, $\epsilon = 2$	20.34	20.48	16.55	15.79

For English-German system, the setup of *FedNMT* with weighted average and domain expert outperforms the baseline system, the BLEU score is higher than the data-centralised model by 0.59 BLEU points in En-De test set. For English-Chinese system, the transformer model with data-centralised training results in the best BLEU score. The BLEU of FL systems is slightly smaller than the local training model, and it still has an improvement versus the model with training corpus from Party A or Party B. Experiment results show that our model is competitive with data-centralised model. As expected, each party benefit from federated training. There is an noticeable increasing of BLEU score compared with the model with only the local training corpus. There is a trade-off between user privacy protection and translation system performance.

4 Conclusion

In this paper, we presented a neural machine translation model under the framework of federated learning, enumerate and examine the efficiency and accuracy

of this model. The experiment validates the benefit from of *FedNMT* without data leakage. Experiment results show that the precision loss of our model is relatively small compared to the local model which holds all the training corpus and the effectiveness of *FedNMT*. Despite the successes of applying federated learning into deep learning, we expect more future research to be conducted on natural language processing with encryption strategy to secure user privacy.

Acknowledgement. This paper is supported by National Key Research and Development Program of China under grant No. 2018YFB0204403.

References

1. Abadi, M., Chu, A., Goodfellow, I.J., et al.: Deep learning with differential privacy, pp. 308–318. ACM (2016)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2015)
3. Ben-David, S., Blitzer, J., Crammer, K., et al.: A theory of learning from different domains. Mach. Learn. **79**(1-2), 151–175 (2010)
4. Bonawitz, K., Eichner, H., Grieskamp, W., et al.: Towards federated learning at scale: system design. CoRR abs/1902.01046 (2019)
5. Chen, M., Suresh, A.T., Mathews, R., et al.: Federated learning of N-gram language models, pp. 121–130 (2019)
6. Cheng, H.-P., et al.: Towards decentralized deep learning with differential privacy. In: Da Silva, D., Wang, Q., Zhang, L.-J. (eds.) CLOUD 2019. LNCS, vol. 11513, pp. 130–145. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-23502-4_10
7. Gehring, J., Auli, M., Grangier, D., et al.: Convolutional sequence to sequence learning. Proc. Mach. Learn. Res. **70**, 1243–1252 (2017)
8. Hisamoto, S., Post, M., Duh, K.: Membership inference attacks on sequence-to-sequence models: is my data in your machine translation system? Trans. Assoc. Comput. Linguistics **8**, 49–63 (2020)
9. Lample, G., Ott, M., Conneau, A., et al.: Phrase-based & neural unsupervised machine translation. In: EMNLP (2018)
10. Li, W., et al.: Privacy-preserving federated brain tumour segmentation. In: Suk, H.-I., Liu, M., Yan, P., Lian, C. (eds.) MLMI 2019. LNCS, vol. 11861, pp. 133–141. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32692-0_16
11. Li, Z., Sun, M.: Punctuation as implicit annotations for Chinese word segmentation. Comput. Linguistics **35**(4), 505–512 (2009)
12. McMahan, B., Moore, E., Ramage, D., et al.: Communication-efficient learning of deep networks from decentralized data. Proc. Mach. Learn. Res. **54**, 1273–1282 (2017)
13. Moriai, S.: Privacy-preserving deep learning via additively homomorphic encryption. In: 26th IEEE Symposium on Computer Arithmetic, ARITH 2019, Kyoto, Japan, 10–12 June 2019, p. 198. IEEE (2019)
14. Papineni, K., Roukos, S., Ward, T., Zhu, W.: BLEU: a method for automatic evaluation of machine translation, pp. 311–318. ACL (2002)
15. Peterson, D., Kanani, P., Marathe, V.J.: Private federated learning with domain adaptation. CoRR abs/1912.06733 (2019)
16. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need (2017)
17. Wang, S., Liu, Y., Wang, C., et al.: Improving back-translation with uncertainty-based confidence estimation, pp. 791–802 (2019)