Improving Federated Learning for Aspect-based Sentiment Analysis via Topic Memories

Han Qin♠*, Guimin Chen♢*, Yuanhe Tian♥*, Yan Song♠†
♠The Chinese University of Hong Kong (Shenzhen)
♢QTrade ♥University of Washington
♠hanqin@link.cuhk.edu.cn ♢chenguimin@foxmail.com
♥yhtian@uw.edu ♠songyan@cuhk.edu.cn

Abstract

Aspect-based sentiment analysis (ABSA) predicts the sentiment polarity towards a particular aspect term in a sentence, which is an important task in real-world applications. To perform ABSA, the trained model is required to have a good understanding of the contextual information, especially the particular patterns that suggest the sentiment polarity. However, these patterns typically vary in different sentences, especially when the sentences come from different sources (domains), which makes ABSA still very challenging. Although combining labeled data across different sources (domains) is a promising solution to address the challenge, in practical applications, these labeled data are usually stored at different locations and might be inaccessible to each other due to privacy or legal concerns (e.g., the data are owned by different companies). To address this issue and make the best use of all labeled data, we propose a novel ABSA model with federated learning (FL) adopted to overcome the data isolation limitations and incorporate topic memory (TM) proposed to take the cases of data from diverse sources (domains) into consideration. Particularly, TM aims to identify different isolated data sources due to data inaccessibility by providing useful categorical information for localized predictions. Experimental results on a simulated environment for FL with three nodes demonstrate the effectiveness of our approach, where TM-FL outperforms different baselines including some well-designed FL frameworks.[1]

1 Introduction

Aspect-based sentiment analysis (ABSA) is one of the most popular natural language processing (NLP) tasks aiming to predict the sentiment polarity (i.e., "*positive*", "*negative*", and "*neutral*") for an aspect term in sentences. Currently, methods based on deep learning have been widely utilized for ABSA and demonstrated excellent potentials (Chen et al., 2017; Zadeh et al., 2017; Zhang et al., 2018; Xue and Li, 2018; Zhao et al., 2018; Chaturvedi et al., 2018; Xu et al., 2019b). However, these methods still reach a bottleneck if there is no enough labeled training data. One feasible solution for it is to leverage extra labeled data from other sources or domains. However, in real applications, these data are always stored in different locations (nodes) and are inaccessible to each other owing to privacy or legal concerns.

To address the data isolation issue, federated learning (FL) (Shokri and Shmatikov, 2015; Konečný et al., 2016a,b) is proposed and has shown its great promises for many machine learning tasks, such as user-computer interaction (Aono et al., 2017), medical image analysis (Sheller et al., 2018), and financial data analysis (Yang et al., 2019a; He et al., 2020). In some cases, data in different nodes are encrypted and aggregated to the centralized model, and they are invisible to each other during the training stage (Hard et al., 2018). This property makes FL an essential technique for real applications with privacy and security requirements.

Recently, FL has been applied to many downstream natural language processing (NLP) applications (Zhu et al., 2020) such as mobile keyboard prediction (Hard et al., 2018), language model training (Chen et al., 2019), representation learning (Liu et al., 2019), spoken language understanding (Huang et al., 2020), medical relation extraction (Sui et al., 2020), medical named entity recognition (Ge et al., 2020), and news recommendation (Qi et al., 2020). However, conventional FL techniques are more suitable for nodes sharing homogeneous data, which is seldom the case for NLP tasks be-

*Equal contribution.

†Corresponding author.

[1] The code involved in this paper are released at https://github.com/cuhksz-nlp/ASA-TM.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3942–3954
November 7–11, 2021.

cause text data are usually heterogeneous in vocabularies and expression patterns. Particularly for ABSA, it is sensitive to the domain information, where one particular token may suggest completely different sentiment polarity in different datasets. Therefore, the restricted data access in traditional federated learning approaches could result in inferior performance for ABSA since they cannot update the model using all domain information. Unfortunately, limited attentions have been paid to address this issue. Most existing approaches with FL on NLP (e.g., for language modeling (Hard et al., 2018; Chen et al., 2019), named entity recognition (Ge et al., 2020), and text classification (Zhu et al., 2020)) mainly focus on optimizing the learning process and ignore domain diversities.

In this paper, we propose a neural model based on FL for ABSA in a distributed environment, namely TM-FL, with a topic memory to enhance FL by providing categorical (topic) information for localized predictions, which can address the difficulty of identifying text sources caused by data inaccessibility. Specifically, the topic model serves as a server-side component to read different inputs from each node and respond with categorical weights to help the backbone ABSA classifier. Compared with previous ABSA studies that leverage extra features, e.g., document information (Li et al., 2018a), commonsense knowledge (Ma et al., 2018), and word dependencies (Tang et al., 2020), our approach offers an alternative to improve ABSA by leveraging extra labeled data through the FL framework enhanced by TM. Experimental results on a simulated environment with isolated data from laptop, restaurant reviews, and social media (i.e., Tweets), demonstrate the effectiveness of our approach, where TM-FL outperforms different baselines including the ones with well designed FL framework.

2 Related Work

2.1 Federated Learning

Federated learning (FL) was first proposed by Google and then further developed by many studies over the past years (Shokri and Shmatikov, 2015; Konečnỳ et al., 2016a,b; McMahan et al., 2017). FL is to build machine-learning models based on datasets distributed across multiple devices while preventing data leakage. Generally, in federated learning, the data is locally stored in different nodes and never uploaded to the server or exchanged with each other node. Thus, the centralized model on the server-side cannot directly exploit the data to optimize its parameters. Instead, each node computes a local model update based on their data, and then the local updates in all nodes are aggregated by the centralized model to optimize parameters. Since such local model updates cannot be directly translated to the original data, the data privacy and security are significantly enhanced. However, there are some other approaches to apply FL, such as sending transformed or encrypted data which cannot be converted to the original data (Hard et al., 2018). FL has been applied to many areas (Yang et al., 2019b; Liu et al., 2020; Wang et al., 2020b; Zheng et al., 2020) and recently, many studies focus on optimizing the learning process (Konečnỳ et al., 2016b; Li et al., 2019; Zheng et al., 2020; Wang et al., 2020b).

Particularly, the FEDERATEDAVERAGING algorithm, proposed by McMahan et al. (2017), is to combine node updates and produce a new global model. At the beginning of each training round, the global model is sent to a subset of nodes. Each of the selected nodes then randomly samples a subset of its local dataset to train the model locally. In the training process, the nodes compute the average gradient on their local datasets with the current global model. The server collects the gradients and aggregates them to update the global model. This process repeats until the global model converges.

2.2 Aspect-based Sentiment Analysis

Aspect-based sentiment analysis (ABSA) is a longstanding NLP task of detecting a sentiment polarity towards a given aspect term in a sentence. Many recent studies applied neural network approaches to ABSA (Chen et al., 2017; Ma et al., 2017; Fan et al., 2018; Gu et al., 2018; He et al., 2018b; Huang and Carley, 2018; Li et al., 2018b; Chen and Qian, 2019; Hu et al., 2019; Du et al., 2019; Sun et al., 2019; Zhang et al., 2019). Usually, external knowledge is incorporated to obtain better understandings of contextual information so as to enhance the model performance for natural language processing downstream tasks (including ABSA) (Li et al., 2018a; Ma et al., 2018; Chen et al., 2020b; Tang et al., 2020; Tian et al., 2020b; Chen et al., 2021; Tian et al., 2021b,c,d). However, most previous studies assume an ideal environment where all the data is accessible and visible to each other for the experiments, which is rarely the case in real appli-

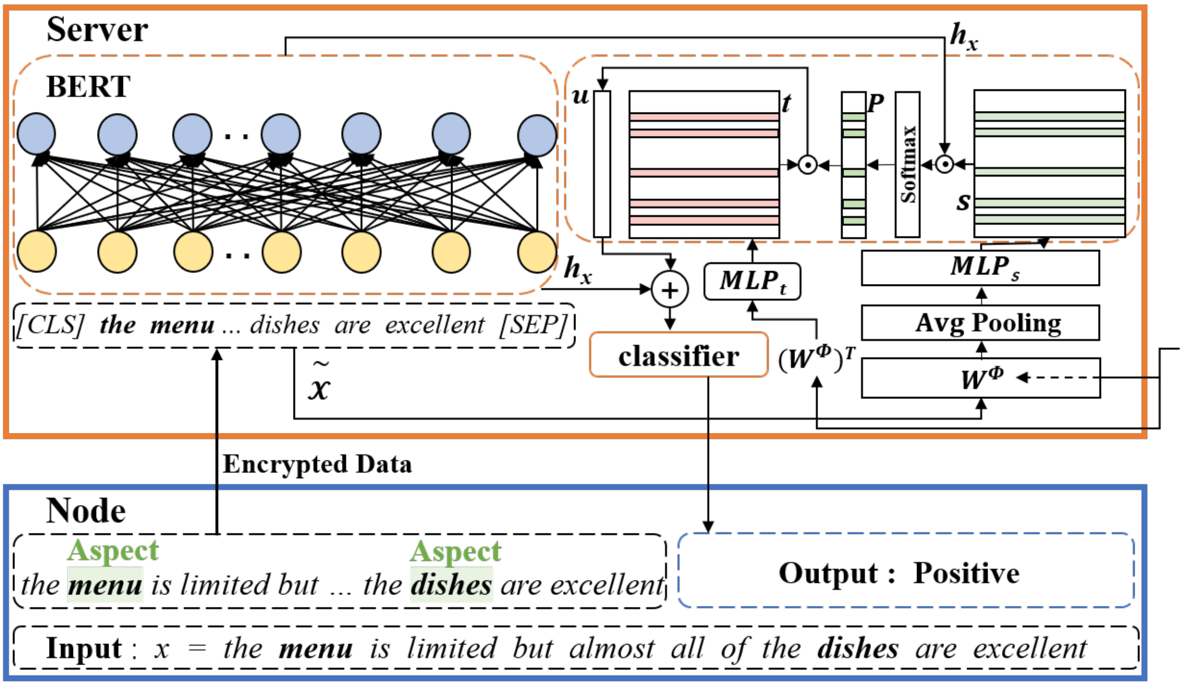

Server
BERT
h_x
u
t
P
Softmax
s
h_x
MLP_t
MLP_s
Avg Pooling
[CLS] the menu ... dishes are excellent [SEP]
classifier
$(W^{\Phi})^T$
W^{Φ}
$\tilde{x}$
Encrypted Data
Node
Aspect
Aspect
the menu is limited but ... the dishes are excellent
Output : Positive
Input : x = the menu is limited but almost all of the dishes are excellent

Figure 1: An overview of the architecture for the centralized model at server of TM-FL, where the aspect term(s) (e.g., *menu* and *dishes*) in the input sentence are highlighted in green.

cations. In this paper, we propose an alternative to handle the data isolation problem and improve ABSA under the constraints of FL by leveraging extra labeled data from different domains through a topic memory module.

3 The Proposed Method

We propose TM-FL for ABSA and the overall server-node architecture of our approach is illustrated in Figure 2. The centralized model is stored in the FL server and data from multiple sources (domains) are stored at different nodes (the i-th node is denoted by $\mathcal{N}_i$), respectively. Encrypted information (e.g., data, vectors, and loss) communicates between each node $\mathcal{N}_i$ and the FL server. In this way, the original data stays in the local node and is not accessible to the other nodes. To encode categorical information to facilitate localized prediction, we incorporate TM into the centralized model (Figure 1). Herein, FL encodes the topic information from the encrypted input and uses the encoded information to guide the centralized model to make a localized prediction. In the following texts, we introduce FL for ABSA and then the centralized model with TM.

3.1 Federated Learning

In federated learning, the data is stored in different local nodes and never exchanged with other nodes. Thus, the centralized model cannot directly access these data, but aggregate the encrypted information of data generated by each local node to complete an update in every training round. Specifically, there are different ways to apply federated learning, such as having the clients send the losses and gradients with respect to the local data back to the FL server (Konečnỳ et al., 2016a), or having the clients send encrypted information which cannot be deciphered back about the local data back to the FL server (Hard et al., 2018). In this paper, we follow the paradigm from Hard et al. (2018) to apply federated learning, where encrypted information, including hidden vectors and loss, are transferred between the server and clients. In addition, following (Chen et al., 2019), we adopt a modified version of FEDERATEDAVERAGING algorithm in which no models are sent to clients. In the training process of FL, the node $\mathcal{N}_i$ firstly encrypts the original input sentence $\mathcal{X}_i = x_1^{(i)}, x_2^{(i)} \cdots x_n^{(i)}$ with n words and the aspect term $\mathcal{A}_i = a_1^{(i)}, a_2^{(i)} \cdots a_m^{(i)}$ with m words ($\mathcal{A}_i$ is usually a sub-string of $\mathcal{X}_i$) into encrypted vectors $\widetilde{\mathcal{X}}_i$ and $\widetilde{\mathcal{A}}_i$ by

$$\widetilde{\mathcal{X}}_i, \widetilde{\mathcal{A}}_i = \text{Encrypt}(\mathcal{X}_i, \mathcal{A}_i) \tag{1}$$

Next, the encrypted $\widetilde{\mathcal{X}}_i$ and $\widetilde{\mathcal{A}}_i$ are sent to the server and fed into the centralized model. Then, the model processes the encrypted input and computes the score vectors $\mathbf{o}_i$ for all sentiment polarities by

$$\mathbf{o}_i = \text{TM-FL}\left(\widetilde{\mathcal{X}}_i, \widetilde{\mathcal{A}}_i\right) \tag{2}$$

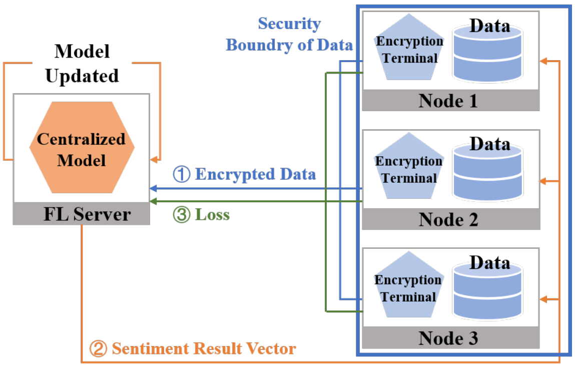


Figure 2: The server-node architecture of our approach.

where each dimension of $\mathbf{o}_i$ corresponds to a particular sentiment polarities among *positive*, *negative*, and *neutral*. Afterward, $\mathbf{o}_i$ is passed back to $\mathcal{N}_i$ and decoded to the model prediction $\widehat{y}_i$ by

$$\widehat{y}^{(i)} = \text{softmax}\left(\mathbf{o}_i\right) \tag{3}$$

After that, we apply the negative log likelihood loss function to the sentiment polarity predictions and compute the loss for node N_i (i.e., $\mathcal{L}_\text{i}$) by

$$\mathcal{L}_\text{i} = -\log p(y^{(\text{i})^*}|\mathcal{X}_i, \mathcal{A}_i) \tag{4}$$

where $p(y^{(\text{i})^*}|\mathcal{X}_i, \mathcal{A}_i)$ denotes the predicted probability of the ground truth sentiment polarity $y^{(\text{i})^*}$ for a given aspect term $\mathcal{A}_i$ in $\mathcal{X}_i$. Finally, $\mathcal{L}_i$ is passed to the server and backpropagation is applied to update the parameters in the centralized model accordingly. The nodes will host no model but only encrypt the local data and send it to the server. In the following texts, we first describe how we construct a topic memory network and use it to capture domain-specific information. Then we explain how we apply our approach to ABSA.

3.2 Centralized Model with Topic Memory

Standard FL cannot utilize the categorical information from the isolated data and thus cannot achieve optimal results for localized prediction. This is a critical barrier for ABSA task, where the data from different sources always contains heterogeneous vocabularies and expressing patterns. In this work, we propose to leverage TM to explore the topic information in the data and use it to guide the centralized model for making localized prediction. As for the input sentence, previous studies concatenate aspect term(s) directly to the end of an input sentence with a special token[2] serving as the separator and feed the resulted sentence+aspect pair into an encoder (Song et al., 2019; Zeng et al., 2019; Phan and Ogunbona, 2020; Veyseh et al., 2020; Chen et al., 2020a). This straightforward method has been proved to be effective for ABSA. Following this paradigm, in the centralized model, we concatenate the encrypted $\widetilde{\mathcal{X}}_i$ and $\widetilde{\mathcal{A}}_i$ into a new sequence $\widetilde{\mathcal{X}}_i^E$ with a special token inserted between them, formalized by

$$\widetilde{\mathcal{X}}_i^E = \widetilde{\mathcal{X}}_i + [\text{SEP}] + \widetilde{\mathcal{A}}_i \tag{5}$$

Then we encode $\widetilde{\mathcal{X}}_i^E$ into vectorized representations $\mathbf{h}_{\mathcal{X}_i} \in \mathbb{R}^{d_h}$ (d_h is the vector dimension) by

$$\mathbf{h}_{\mathcal{X}_i} = \text{SE}(\widetilde{\mathcal{X}}_i^E) \tag{6}$$

where SE is the encoder for encoding the encrypted information. Based on $\widetilde{\mathcal{X}}_i$ and $\mathbf{h}_{\mathcal{X}_i}$, TM generates the topic vector (which is denoted as $\mathbf{u}_i \in \mathbb{R}^{d_h}$) through the following process.

Firstly, we use a matrix $\mathbf{W}^\phi \in \mathbb{R}^{d_v \times d_t}$ (d_v and d_t denote the vocabulary size and the topic size, respectively) to represent the topic model which is to obtain the categorical information, where the matrix $\mathbb{W}^\phi$ is from a pre-trained neural topic model[3]. Each row of $\mathbf{W}^\phi$ can be regarded as a word embedding for a particular word with each dimension of the embedding corresponding to the value for a specific topic. Similarly, each column of $\mathbf{W}^\phi$ can be regarded as a topic embedding for a particular topic. Next, we use $\mathbf{W}^\phi$ to map all words in $\widetilde{\mathcal{X}}_i$ to the corresponding word embeddings (the embedding for the j-th word in $\widetilde{\mathcal{X}}_i$ is denoted as $\mathbf{e}_{i,j}^x \in \mathbb{R}^{d_t}$), and map all topics to the corresponding topic embeddings (the embedding for the k-th topic is denoted as $\mathbf{e}_k^t \in \mathbb{R}^{d_v}$). Then, we apply average pooling to the word embeddings over $\widetilde{\mathcal{X}}_i$

$$\mathbf{e}_i^x = \text{AvgPooling}(\mathbf{e}_{i,1}^x \cdots \mathbf{e}_{i,j}^x \cdots \mathbf{e}_{i,l}^x) \tag{7}$$

where the k-th topic is represented by a one-dimensional vector ($\mathbf{e}_{i,k}^s \in \mathbb{R}^1$) in $\mathbf{e}_i^x \in \mathbb{R}^{d_t}$. We feed $\mathbf{e}_{i,k}^s$ into a multi-layer perceptron (MLP) to compute the source memories $\mathbf{s}_{i,k}$ by

$$\mathbf{s}_{i,k} = \text{MLP}_s(\mathbf{e}_{i,k}^s) \tag{8}$$

Afterward, we compute the attention weights $p_{i,k}$ for the k-th topic by

$$p_{i,k} = \frac{exp\left(\mathbf{h}_{\mathcal{X}_i} \cdot \mathbf{s}_{i,k}\right)}{\sum_{k=1}^{d_t} exp\left(\mathbf{h}_{\mathcal{X}_i} \cdot \mathbf{s}_{i,j}\right)} \tag{9}$$

Finally, $p_{i,k}$ is applied to target memories by

$$\mathbf{u}_i = \sum_{k=1}^{d_t} p_{i,k} \cdot \mathbf{t}_k \tag{10}$$

[2] If the encoder is BERT, the special token will be `[SEP]`.

[3] The details of the neural topic model are illustrated in Section 4.2

Dataset		Pos. #	Neu. #	Neg. #
LAP14	Train	994	464	870
	Test	341	169	128
REST14	Train	2,164	637	807
	Test	728	196	182
TWITTER	Train	1,561	3,127	1,560
	Test	173	346	173

Table 1: The number of aspect terms with "*positive*" (Pos.), "*neutral*" (Neu.), and "*negative*" (Neg.) sentiment polarities in the train/test sets of all three datasets.

where the target memory $\mathbf{t}_k \in \mathbb{R}^{d_h}$ is obtained by

$$\mathbf{t}_k = \text{MLP}_t(\mathbf{e}^t_k) \tag{11}$$

We perform element-wise addition on $\mathbf{h}_{\mathcal{X}_i}$ and $\mathbf{t}_i$, and pass the resulting vector to a fully connected layer to obtain $\mathbf{o}_i$, which can be formalized by

$$\mathbf{o}_i = \mathbf{W} \cdot (\mathbf{h}_{\mathcal{X}_i} + \mathbf{u}_i) + \mathbf{b} \tag{12}$$

where $\mathbf{W}$ and $\mathbf{b}$ are the trainable matrix and bias vector, respectively, in the fully connected layer.

4 Experimental Settings

4.1 Datasets

To test the proposed approach, we follow the convention of recent FL-based NLP studies (Liu et al., 2019; Huang et al., 2020; Zhu et al., 2020; Sui et al., 2020; Tian et al., 2021a) to build a simulated environment where isolated data are stored in three nodes. Each node contains one of the three widely used English benchmark datasets (i.e., LAP14, REST14 (Pontiki et al., 2014), and TWITTER (Dong et al., 2014)) for ABSA, where each node contains all the data from the same domain. Particularly, LAP14 contains laptop computer reviews; REST14 consists of online reviews from restaurants; TWITTER includes tweets collected through Twitter API. For LAP14 and REST14, following previous studies (Tang et al., 2016b; Chen et al., 2017; He et al., 2018a), the aspect terms with "*conflict*" sentiment polarity[4] and the sentences without an aspect term are removed. For all datasets, we use their official train/test splits[5] and randomly pick 10% of the training set serving as the development set so as to find the best hyper-parameters, which are then applied to our

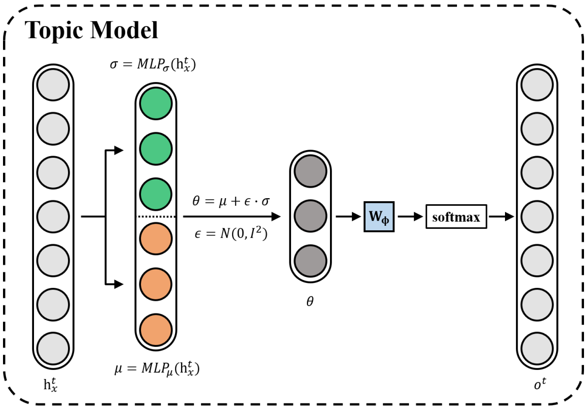

Topic Model
$\sigma = MLP_\sigma(\mathrm{h}_x^t)$
$\theta = \mu + \epsilon \cdot \sigma$
$\epsilon = N(0, I^2)$
$\mathbf{W}_\Phi$
softmax
θ
$\mu = MLP_\mu(\mathrm{h}_x^t)$
h_x^t
o^t

Figure 3: The overview of the neural topic model.

models when learning on the entire training set[6]. The statistics of the datasets (i.e., the numbers of aspect terms with "*positive*", "*negative*", and "*neutral*" sentiment polarities) of the three datasets is reported in Table 1.

To further improve the model performance by leveraging extra labeled data from different domains, we train a neural topic model and then obtain the topic-vocab matrix to initialize W^ϕ. We train our neural topic model on five online datasets: (1) Yelp dataset[7], (2) IMDb dataset[8], (3) Amazon dataset[9], (4) SemEval-2017 Task 4 (SemEval2017) dataset[10], and (5) MitchellAI-13-Opensentiment dataset (Mitchell et al., 2013). Particularly, Yelp contains online reviews of restaurants and hotels; IMDb contains reviews of movies; Amazon includes comments on goods; SemEval2017 and MitchellAI-13-Opensentiment contain tweets. We randomly sample 75K sentences from each domain (i.e., reviews of restaurants and hotels, reviews of movies, comments on goods, and tweets[11]) and put them together to form the combined training data with roughly 300K sentences[12] for the topic model.

4.2 Neural Topic Model

Inspired by Miao et al. (2017), we train a neural topic model based on variational auto-encoder

[4] "*Conflict*" is a sentiment polarity used to identify the aspect terms that have contradictory sentiment polarities in the same sentence in LAP14 and REST14.

[5] It is worth noting that LAP14, REST14, and TWITTER do not have their official development sets.

[6] We report the hyper-parameter settings in Appendix A.

[7] We obtained Yelp dataset from `https://www.yelp.com/dataset`

[8] We obtained IMDb dataset from `https://course.fast.ai/datasets#nlp`

[9] We obtained Amazon dataset from `https://course.fast.ai/datasets#nlp`

[10] We obtained SemEval-2017 Task 4 dataset from `https://alt.qcri.org/semeval2017/task4/`

[11] Since MitchellAI-13-Opensentiment only has 25K tweet sentences in total, we extract the other 50K tweet sentences from SemEval2017 and then merge them into a data collection consisting of 75K sentences of tweets.

[12] The duplicated sentences are removed.

(Kingma and Welling, 2013) to extract the latent topic distribution $\mathbf{z}$ with prior parameters μ and ϕ of the datasets, where the overall structure of the topic model is illustrated in Figure 3. Specifically, given an input sentence $\mathcal{X} = x_1 x_2 x_3 \cdots x_n$, we first obtain the one-hot representation $\mathcal{X}_{bow}$ of X and then pass it to a multi-layer perceptron (MLP) to get the hidden representation $\mathbf{h}^t_{\mathcal{X}}$ of the input sentence, formalized by

$$\mathcal{X}_{bow} = \text{one-hot}(\mathcal{X}) \tag{13}$$

and

$$\mathbf{h}^t_{\mathcal{X}} = \text{MLP}_{bow}(\mathcal{X}_{bow}) \tag{14}$$

where $\mathbf{h}^t_{\mathcal{X}} \in \mathbb{R}^{d_h}$. Next, the prior parameters μ and σ of the latent topic distribution $\mathbf{z}$ are estimated and defined as

$$\mu = \text{MLP}_{\mu}(\mathbf{h}^t_{\mathcal{X}}) \tag{15}$$

and

$$\sigma = \text{MLP}_{\sigma}(\mathbf{h}^t_{\mathcal{X}}) \tag{16}$$

where MLP_{μ} and MLP_{σ} refer to two different multi-layer perceptrons. Then, we randomly sample θ from $\mathbf{z}$ to be the latent topic representation of the input sentence $\mathcal{X}$. Afterward, we generate the output vector by

$$\mathbf{o}^{\text{t}} = \text{softmax}(\mathbf{W}_{\phi} \cdot \theta + \mathbf{b}_{\phi}) \tag{17}$$

where $\mathbf{W}^{\phi} \in \mathbb{R}^{d_v \times d_t}$ and $\mathbf{b}^{\phi} \in \mathbb{R}^{d_v}$ are trainable matrix and bias vector, respectively; $\mathbf{o}^{\text{t}} \in \mathbb{R}^{n \times d_v}$ refers to the predicted probability of words from all vocabularies of each position in the original input sentence. In practice, we train the topic model in an unsupervised manner and then extract the topic-vocab matrix to initialize W^{ϕ}. For sampling θ, we sample another random variable $\widehat{\epsilon} \in N(0,1)$ and then parameterize θ by $\theta = \mu + \widehat{\epsilon} \cdot \sigma$.

4.3 Implementation

In the experiments, we run the baselines without federated learning (i.e. BT-b and BT-l) on the single dataset (i.e. LAP14, REST14, or TWITTER) and the combined dataset consisting of all the three datasets, denoted by the union dataset. However, it is rarely practical to have the model trained on the union dataset in real applications (since the data are isolated in different nodes). Therefore, the experimental results on the union dataset reveal the possible upper-boundary of FL-based models and they are mainly used for reference. For FL baselines (i.e. FL) and our proposed approaches (i.e., TM-FL), we run them in the simulated environment where the LAP14, REST14, and TWITTER datasets are isolated to three nodes. Specifically, the first node holds LAP14; the second node holds REST14; the third node holds TWITTER.

		LAP14		**REST14**		**TWITTER**	
		Acc	F1	Acc	F1	Acc	F1
1	BT-b (single)	76.65	73.40	84.02	76.26	72.64	71.02
2	BT-b (union)	**80.72**	**76.87**	**85.54**	**78.68**	**76.16**	**74.80**
3	FL (BT-b)	79.31	75.11	84.46	76.95	74.57	73.32
4	TM-FL (BT-b)	80.56	76.78	84.55	77.58	74.76	73.54
5	BT-l (single)	78.84	74.73	85.27	77.80	73.31	72.38
6	BT-l (union)	**82.60**	**79.87**	**86.96**	**80.09**	**77.02**	**76.15**
7	FL (BT-l)	81.35	78.21	85.71	78.28	74.28	73.46
8	TM-FL (BT-l)	82.29	79.25	86.07	79.00	74.57	73.63

Table 2: Accuracy and Macro-F1 scores of models using BERT-base (BT-b) and BERT-large (BT-l) under different settings on three benchmark datasets.

For encoder, considering that high-quality text representations from pre-trained embeddings or language models are able to effectively to enhance the model performance (Mikolov et al., 2013; Song et al., 2018a,b; Song and Shi, 2018; Devlin et al., 2019; Diao et al., 2020; Song et al., 2021) and BERT-based models have achieved great success in many NLP tasks (Mao et al., 2019; Xu et al., 2019a; Song et al., 2020; Tang et al., 2020; Tian et al., 2020a,c, 2021b,c; Qin et al., 2021a,b), we use the BERT-base-uncased and BERT-large-uncased[13] (Devlin et al., 2019) to encode the encrypted input[14] (i.e., $\widetilde{\mathcal{X}}_i$ and $\widetilde{\mathcal{A}}_i$) from $\mathcal{N}_i$. For TM, we train our neural topic model using an unsupervised approach proposed by Miao et al. (2017) and then use the resulted topic-vocab matrix to initialize $\mathbf{W}^{\phi}$ in TM-FL. In the training process of TM-FL, both BERT and $\mathbf{W}^{\phi}$ are updated.[15] Moreover, it is noted that for baselines (i.e., BT) on the single dataset and the union dataset, we choose the models based on their F1 scores with respect to the dev set of each dataset separately. For FL and TM-FL, we choose the models according to their average F1 score of the three F1 scores over the dev sets of the three datasets. For the evaluation metrics, we follow previous studies (Tang et al., 2016a; Chen et al., 2017; He et al., 2018a; Sun et al., 2019; Zhang et al., 2019) to evaluate all models via accuracy and macro-averaged F1 scores over all sentiment polarities, i.e., *positive*, *neutral* and *negative*.

[13] We obtain the BERT models from `https://github.com/huggingface/pytorch-pretrained-BERT`.

[14] For the sake of simplicity, we do not perform actual encryption in the simulated environment.

[15] We report the hyper-parameter settings of different models with their size and running speed in Appendix B.

	LAP14		REST14		TWITTER	
	Acc	F1	Acc	F1	Acc	F1
*Mao et al. (2019)	75.84	72.49	82.49	72.10	72.35	69.45
*Xu et al. (2019b)	78.07	75.08	84.95	76.96	-	-
Sun et al. (2019)	77.19	72.99	82.30	74.02	74.66	73.66
Zhang et al. (2019)	75.55	71.05	81.22	72.94	72.69	70.59
Wang et al. (2020a)	78.21	74.07	**86.60**	**81.35**	76.15	74.88
Tang et al. (2020)	79.80	75.60	86.30	80.00	**77.90**	**75.40**
*FL (BERT-large)	81.35	78.21	85.71	78.28	74.28	73.46
*TM-FL(BERT-large)	**82.29**	**79.25**	86.07	79.00	74.57	73.63

Table 3: Comparison of model performance (accuracy and F1 scores) of our FL-based models (i.e., TM-FL and FL) with previous studies on the benchmark datasets (in new environments). Models with BERT-large are marked by "*".

5 Results and Analyses

5.1 Overall Results

To evaluate the TM-FL's performance, we compare it with 1) the baseline FL models without TM, i.e., FL (BT-b) and FL (BT-l); and 2) two BERT-only models without FL that all training instances are not isolated and they are accessible to each other. Table 2 illustrates the accuracy and F1 scores of our TM-FL models and all the aforementioned baselines on the test set of three benchmark datasets.[16]

There are several observations. First, in most cases, models under the FL framework (ID: 3, 4, 7, 8) outperform the models trained on the single datasets (ID: 1, 5) with different encoders. This confirms that FL works well to leverage extra isolated data with both BERT-base and BERT-large encoders. Second, FL baselines (ID: 3, 7) fail to outperform the models trained on the union of all datasets (ID: 2, 6) with different encoders on all datasets, which demonstrates that even though FL can leverage extra isolated data, it still fails to achieve the upper bound performance provided by models (ID 2, 6) that do not suffer from the data isolation problem. Third, our TM-FL models (ID: 4, 8) consistently outperform the FL baselines (ID: 3, 7) on all datasets. In addition, it is promising to observe that some results (e.g., ID: 4 on Lap14) from TM-FL are very close to the reference BERT-only models (ID: 2, 6) that provide potential upper boundaries for FL-based models, which demonstrates the effectiveness of the proposed TM module to leverage categorical information to facilitate localized prediction. Moreover, TM-FL shows higher improvements over FL on LAP14 and REST14 than that on TWITTER, which can be explained by that LAP14 and REST14 are product reviews focusing on a particular area whereas TWITTER contains social media texts that may share heterogeneity. Such difference, including the difference among domains and within the TWITTER domain, distracts the model on TWITTER.

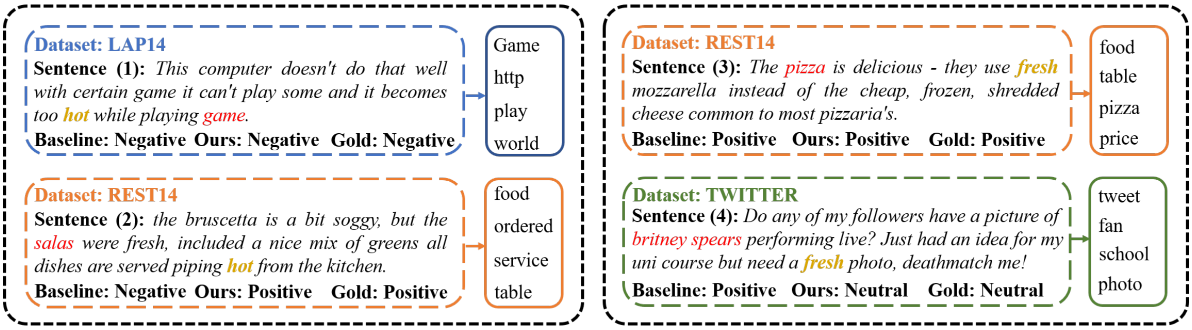
	Lap14		Rest14		TWITTER	
	Acc	F1	Acc	F1	Acc	F1
FL (BT-b)	79.31	75.11	84.46	76.95	74.57	73.32
TM-FL + R. (BT-b)	79.84	75.57	84.32	77.01	73.75	73.39
TM-FL + T. (BT-b)	**80.56**	**76.78**	**84.55**	**77.58**	**74.76**	**73.54**
FL (BT-l)	81.35	78.21	85.71	78.28	74.28	73.46
TM-FL + R. (BT-l)	81.98	78.52	85.52	78.75	74.11	73.49
TM-FL + T. (BT-l)	**82.29**	**79.25**	**86.07**	**79.00**	**74.57**	**73.63**

Table 4: Experimental results of FL baselines, our proposed TM-FL with random initialized $\mathbf{W}^{\phi}$ (R.) and pretrained $\mathbf{W}^{\phi}$ (T.) on the test sets, where BT-b and BT-l refer to BERT-base and BERT-large respectively.

5.2 Comparison with Previous Studies

Since our experimental settings are different from the settings of most previous studies on the three benchmark datasets, direct comparisons of our results with previous studies are not valid. Compared with those previous studies focusing on a single domain, FL can access extra data to help the model even though data from different datasets are not visible to each other. For previous studies working on multiple datasets at the same time and leveraging external knowledge, they do not conduct their experiments in an environment suffering from data isolation problems. To provide relatively fair comparisons with previous studies on the single dataset, we build another three simulated environments for FL and TM-FL where a single dataset, instead of the three datasets, is distributed through all the isolated nodes in each environment. Thus, it is ensured that for each dataset, external knowledge is not introduced into the model during the training process. Therefore, to a certain extent, it is relatively valid to compare our results with previous studies on every single dataset, where the comparisons are reported in Table 3. It is noted that although TM-FL suffers from data isolation under the simulation setting, it still outperforms some studies (Mao et al., 2019; Xu et al., 2019b) using BERT-large (marked by "*") and achieve state-of-the-art results on Lap14, which further confirms the effectiveness of our approach to leverage local isolated data. Besides, TM-FL fails to outperform Wang et al. (2020a) and Tang et al. (2020) on Rest14 and TWITTER, which could be explained that they leverage dependency

[16] For the same group of models, we report the F1 scores on the development sets in Appendix C, and the mean and standard deviation of their test set results in Appendix D.



Dataset: LAP14

Sentence (1): *This computer doesn't do that well with certain game it can't play some and it becomes too hot while playing game.*

Baseline: Negative Ours: Negative Gold: Negative

Dataset: REST14

Sentence (2): *the bruscetta is a bit soggy, but the salas were fresh, included a nice mix of greens all dishes are served piping hot from the kitchen.*

Baseline: Negative Ours: Positive Gold: Positive

Dataset: REST14

Sentence (3): *The pizza is delicious - they use fresh mozzarella instead of the cheap, frozen, shredded cheese common to most pizzaria's.*

Baseline: Positive Ours: Positive Gold: Positive

Dataset: TWITTER

Sentence (4): *Do any of my followers have a picture of britney spears performing live? Just had an idea for my uni course but need a fresh photo, deathmatch me!*

Baseline: Positive Ours: Neutral Gold: Neutral

(a) (b)

Figure 4: A case study on two groups of sentences , where the aspect terms extracted from different nodes (i.e., LAP14, REST14, and TWITTER) are highlighted in red colors , and the predictions of TM-FL (BERT-large) and the FL baseline, as well as the gold labels, are presented below the corresponding sentence. The word list on the right side shows the top four topics (ranked by their receiving wights) in TM.

information and use advanced architectures (e.g., GCN) to encode it.

5.3 The effect of Topic Model

To further explore the effect of the topic model, we test FL and our proposed TM-FL on the test sets with randomly initialized W^{ϕ} and pre-trained W^{ϕ} obtained from the topic model, and report the results in Table 4. First, it is observed that TM-FL with either pre-trained W^{ϕ} or randomly initialized W^{ϕ} outperforms FL, which is reasonable that TM is able to leverage the domain information from extra labeled data and hence help ABSA on localized sentiment polarity prediction. Moreover, TM-FL with pre-trained W^{ϕ} (T.) outperforms TM-FL with randomly initialized W^{ϕ} (R.), demonstrating the effectiveness of the topic model to leverage external topic knowledge with regard to specific domains from other datasets to help the centralized model on ABSA in the simulated environment.

5.4 Case Study

To examine whether our approach with TM is able to capture categorical information to facilitate localized prediction, we conduct a case study with two-sentence groups (i.e., the first group with sentence (1), sentence (2), and the second group with sentence (3), sentence (4)), where all sentences are obtained from different domains (i.e., the test sets of LAP14, REST14, and TWITTER datasets). Figure 4 illustrates such two-sentence groups (the aspect term is highlighted in red color in each sentence), where the predictions from the FL baseline (with BERT-large) and our TM-FL, as well as the gold labels, are also presented. Besides, the top four topic words (ranked based on the received weights in TM) for each individual sentence are presented on the right side. It is worth noting that in each group, both sentences share some same opinion words (i.e. opinion word "*hot*" and "*salas*" which are highlighted in yellow, respectively) which convey contradictory sentiment polarities. Specifically, in the first sentence group, the shared opinion word is "*hot*", which generally demonstrates negative sentiment polarity in laptop reviews while shows positive sentiment polarity in restaurant reviews. In LAP14, among the instances containing "*hot*", 75% of them are associated with the negative sentiment polarity, whereas in REST14, no more than 1/3 of such instances are associated with the negative sentiment polarity. Compared with FL baselines, our approach enhanced by TM successfully leverage the categorical information and hence is able to distinguish the cues from "*hot*" in a particular context, where results incorrect predictions for both instances, whereas FL fails to recognize that "*hot*" suggests a positive sentiment polarity in the sentence (2) from restaurant reviews and thus results in an incorrect prediction. Moreover, in the second sentence group, the word "*fresh*" serves as the shared opinion word with its sentiment polarity generally being generally positive in the domain of restaurant reviews and neutral in the domain of tweets. FL successfully models the opinion word "*fresh*" and predict the sentiment polarity for the aspect term "*pizza*" for sentence (3), while it fails to distinguish the domain difference between sentence (3) and sentence (4). Therefore, due to the cue from "*fresh*" in restaurant domain, FL incorrectly models the opinion word "*fresh*" in another domain and hence make incorrect sentiment polarity prediction with regard to the aspect term "*britney spears*". However, our approach is able to distinguish the domain information in the sentence (3) and sentence (4), resulting incorrect predictions for both instances.

6 Conclusion

In this paper, we present TM-FL, a domain-aware topic memory network under the federated learning framework to enhance ABSA under the restriction of data isolation issues. Specifically, our approach offers an alternative to enhance ABSA by leveraging extra labeled data through the FL framework improved by TM. Experimental results on three widely used English benchmark datasets demonstrate the effectiveness of our method, which outperforms all the baseline models trained under the federated learning framework and competes for state-of-the-art performance on all datasets.

Acknowledgements

This work is supported by NSFC under the project "The Essential Algorithms and Technologies for Standardized Analytics of Clinical Texts" (12026610) and Shenzhen Science and Technology Program under the project "Fundamental Algorithms of Natural Language Understanding for Chinese Medical Text Processing". It is also supported by Shenzhen Institute of Artificial Intelligence and Robotics for Society under the project "Automatic Knowledge Enhanced Natural Language Understanding and Its Applications" (AC01202101001).

References

Yoshinori Aono, Takuya Hayashi, Lihua Wang, Shiho Moriai, et al. 2017. Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Transactions on Information Forensics and Security*, 13(5):1333–1345.

Iti Chaturvedi, Edoardo Ragusa, Paolo Gastaldo, Rodolfo Zunino, and Erik Cambria. 2018. Bayesian network based extreme learning machine for subjectivity detection. *Journal of The Franklin Institute*, 355(4):1780–1797.

Chenhua Chen, Zhiyang Teng, and Yue Zhang. 2020a. Inducing Target-Specific Latent Structures for Aspect Sentiment Classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5596–5607.

Guimin Chen, Yuanhe Tian, and Yan Song. 2020b. Joint Aspect Extraction and Sentiment Analysis with Directional Graph Convolutional Networks. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 272–279.

Guimin Chen, Yuanhe Tian, Yan Song, and Xiang Wan. 2021. Relation Extraction with Type-aware Map Memories of Word Dependencies. *Findings of the Association for Computational Linguistics: ACLIJC-NLP*.

Mingqing Chen, Ananda Theertha Suresh, Rajiv Mathews, Adeline Wong, Cyril Allauzen, Françoise Beaufays, and Michael Riley. 2019. Federated learning of n-gram language models. *arXiv preprint arXiv:1910.03432*.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 452–461.

Zhuang Chen and Tieyun Qian. 2019. Transfer capsule network for aspect level sentiment classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 547–556.

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

Shizhe Diao, Jiaxin Bai, Yan Song, Tong Zhang, and Yonggang Wang. 2020. ZEN: Pre-training Chinese Text Encoder Enhanced by N-gram Representations. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4729–4740.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 49–54.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Tong Xu, and Ming Liu. 2019. Capsule network with interactive attention for aspect-level sentiment classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5489–5498.

Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442.

Suyu Ge, Fangzhao Wu, Chuhan Wu, Tao Qi, Yongfeng Huang, and Xing Xie. 2020. Fedner: Medical named entity recognition with federated learning. *arXiv preprint arXiv:2003.09288*.

Shuqin Gu, Lipeng Zhang, Yuexian Hou, and Yin Song. 2018. A position-aware bidirectional attention network for aspect-level sentiment analysis. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 774–784.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*.

Anxun He, Jianzong Wang, Zhangcheng Huang, and Jing Xiao. 2020. FedSmart: An Auto Updating Federated Learning Optimization Mechanism. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, pages 716–724. Springer.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018a. Effective Attention Modeling for Aspect-level Sentiment Classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1121–1131.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018b. Exploiting document knowledge for aspect-level sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–585.

Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. 2019. Can: Constrained attention networks for multi-aspect sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4593–4602.

Binxuan Huang and Kathleen Carley. 2018. Parameterized convolutional neural networks for aspect level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1091–1096.

Zhiqi Huang, Fenglin Liu, and Yuexian Zou. 2020. Federated learning for spoken language understanding. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3467–3478.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Jakub Konečnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. 2016a. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*.

Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. 2016b. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.

Junjie Li, Haitong Yang, and Chengqing Zong. 2018a. Document-level multi-aspect sentiment classification by jointly modeling users, aspects, and overall ratings. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 925–936.

Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. 2019. Fair resource allocation in federated learning. *arXiv preprint arXiv:1905.10497*.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018b. Transformation networks for target-oriented sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 946–956.

Dianbo Liu, Dmitriy Dligach, and Timothy Miller. 2019. Two-stage federated phenotyping and patient representation learning. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 283–291.

Yang Liu, Anbu Huang, Yun Luo, He Huang, Youzhi Liu, Yuanyuan Chen, Lican Feng, Tianjian Chen, Han Yu, and Qiang Yang. 2020. Fedvision: An online visual object detection platform powered by federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13172–13179.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4068–4074.

Yukun Ma, Haiyun Peng, and Erik Cambria. 2018. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Qianren Mao, Jianxin Li, Senzhang Wang, Yuanning Zhang, Hao Peng, Min He, and Lihong Wang. 2019. Aspect-based sentiment classification with attentive neural turing machines. In *IJCAI*, pages 5139–5145.

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

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *International Conference on Machine Learning*, pages 2410–2419. PMLR.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1643–1654.

Minh Hieu Phan and Philip O. Ogunbona. 2020. Modelling Context and Syntactical Features for Aspect-based Sentiment Analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3211–3220.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect Based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35.

Tao Qi, Fangzhao Wu, Chuhan Wu, Yongfeng Huang, and Xing Xie. 2020. Privacy-preserving news recommendation model learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1423–1432.

Han Qin, Guimin Chen, Yuanhe Tian, and Yan Song. 2021a. Improving Arabic Diacritization with Regularized Decoding and Adversarial Training. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Han Qin, Yuanhe Tian, and Yan Song. 2021b. Relation Extraction with Word Graphs from N-grams. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Micah J Sheller, G Anthony Reina, Brandon Edwards, Jason Martin, and Spyridon Bakas. 2018. Multi-institutional Deep Learning Modeling without Sharing Patient Data: A Feasibility Study on Brain Tumor Segmentation. In *International MICCAI Brainlesion Workshop*, pages 92–104.

Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1310–1321.

Yan Song and Shuming Shi. 2018. Complementary Learning of Word Embeddings. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4368–4374.

Yan Song, Shuming Shi, and Jing Li. 2018a. Joint Learning Embeddings for Chinese Words and Their Components via Ladder Structured Networks. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4375–4381.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018b. Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180.

Yan Song, Yuanhe Tian, Nan Wang, and Fei Xia. 2020. Summarizing Medical Conversations via Identifying Important Utterances. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 717–729.

Yan Song, Tong Zhang, Yonggang Wang, and Kai-Fu Lee. 2021. ZEN 2.0: Continue Training and Adaption for N-gram Enhanced Text Encoders. *arXiv preprint arXiv:2105.01279*.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314*.

Dianbo Sui, Yubo Chen, Jun Zhao, Yantao Jia, Yuantao Xie, and Weijian Sun. 2020. FedED: Federated learning via ensemble distillation for medical relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2118–2128.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019. Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5679–5688.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective LSTMs for Target-Dependent Sentiment Classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect Level Sentiment Classification with Deep Memory Network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency Graph Enhanced Dual-transformer Structure for Aspect-based Sentiment Classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6578–6588.

Yuanhe Tian, Guimin Chen, Han Qin, and Yan Song. 2021a. Federated Chinese Word Segmentation with Global Character Associations. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4306–4313, Online.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021b. Aspect-based Sentiment Analysis with Type-aware Graph Convolutional Networks and Layer Ensemble. In *Proceedings of the 2021 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2910–2922.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021c. Enhancing Aspect-level Sentiment Analysis with Word Dependencies. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3726–3739.

Yuanhe Tian, Guimin Chen, Yan Song, and Xiang Wan. 2021d. Dependency-driven Relation Extraction with Attentive Graph Convolutional Networks. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Yuanhe Tian, Wang Shen, Yan Song, Fei Xia, Min He, and Kenli Li. 2020a. Improving Biomedical Named Entity Recognition with Syntactic Information. *BMC Bioinformatics*, 21:1471–2105.

Yuanhe Tian, Yan Song, and Fei Xia. 2020b. Joint Chinese Word Segmentation and Part-of-speech Tagging via Multi-channel Attention of Character N-grams. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2073–2084.

Yuanhe Tian, Yan Song, Fei Xia, and Tong Zhang. 2020c. Improving Constituency Parsing with Span Attention. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1691–1703.

Amir Pouran Ben Veyseh, Nasim Nouri, Franck Dernoncourt, Quan Hung Tran, Dejing Dou, and Thien Huu Nguyen. 2020. Improving Aspect-based Sentiment Analysis with Gated Graph Convolutional Networks and Syntax-based Regulation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4543–4548.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020a. Relational Graph Attention Network for Aspect-based Sentiment Analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238.

Yansheng Wang, Yongxin Tong, and Dingyuan Shi. 2020b. Federated latent dirichlet allocation: A local differential privacy based framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6283–6290.

Hu Xu, Bing Liu, Lei Shu, and S Yu Philip. 2019a. Bert Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2324–2335.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019b. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2324–2335.

Wei Xue and Tao Li. 2018. Aspect Based Sentiment Analysis with Gated Convolutional Networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523.

Mengwei Yang, Linqi Song, Jie Xu, Congduan Li, and Guozhen Tan. 2019a. The Tradeoff Between Privacy and Accuracy in Anomaly Detection Using Federated XGBoost. *arXiv preprint arXiv:1907.07157*.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019b. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor Fusion Network for Multimodal Sentiment Analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1114.

Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, and Xuli Han. 2019. Lcf: A local context focus mechanism for aspect-based sentiment classification. *Applied Sciences*, 9(16):3389.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4560–4570.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.

Jianqiang Zhao, Xiaolin Gui, and Xuejun Zhang. 2018. Deep convolution neural networks for twitter sentiment analysis. *IEEE Access*, 6:23253–23260.

Wenbo Zheng, Lan Yan, Chao Gou, and Fei-Yue Wang. 2020. Federated meta-learning for fraudulent credit card detection. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*.

Xinghua Zhu, Jianzong Wang, Zhenhou Hong, and Jing Xiao. 2020. Empirical Studies of Institutional Federated Learning for Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 625–634.

Appendix A. Hyper-parameter Settings

Table 5 reports the hyper-parameters tested in training our models. We test all combinations of them for each model and use the one achieving the highest accuracy score in our final experiments.

Hyper-parameters	Values
Learning Rate	$5e-6, \mathbf{1e-5}, 2e-5, 3e-5$
Warmup Rate	$0.06, \mathbf{0.1}$
Dropout Rate	$\mathbf{0.1}$
Batch Size	$8, \mathbf{16}, 32$

Table 5: The hyper-parameters tested in tuning our models, where the best ones used in our final experiments are highlighted in boldface.

Appendix B. Model Size and Performance

Table 6 reports the number of trainable parameters and the inference speed (sentences per second) of the baseline (i.e., BERT (single), BERT (union), and FL with BERT-base and BERT-large) and our models (i.e., TM-FL with BERT-base and BERT-large) on all of the three datasets. All models are performed on an NVIDIA Tesla V100 GPU.

Appendix C. Experimental Results on the Development Set

Table 7 reports the F1 scores of different models on the development sets of LAP14 and REST14.[17]

Appendix D. Mean and Deviation of the Results

In the experiments, we test models with different configurations. For each model, we train it with the best hyper-parameter setting using five different random seeds. We report the mean (μ) and standard deviation (σ) of the F1 scores on the test sets of LAP14, REST14 and TWITTER in Table 8.

Models	**Para.**	**LAP14**	**REST14**	**TWITTER**
		Speed	Speed	Speed
BT-b (single)	109M	63.1	63.1	63.1
BT-b (union)	109M	63.1	63.1	63.1
FL (BT-b)	109M	63.1	63.1	63.1
TM-FL (BT-b)	143M	57.3	57.3	57.3
BT-l (single)	335M	29.2	29.2	29.2
BT-l (union)	335M	29.2	29.2	29.2
FL (BT-l)	335M	29.2	29.2	29.2
TM-FL (BT-l)	380M	23.9	23.9	23.9

Table 6: Numbers of trainable parameters (Para.) in different models and the inference speed (sentences per second) of these models on the test sets of both datasets. "BT-b" and "BT-l" refer to encoder BERT-base and BERT-large respectively.

Models	**LAP14**	**REST14**
BT-b (single)	73.64	75.85
BT-b (union)	**76.75**	**78.25**
FL (BT-b)	74.92	77.19
TM-FL (BT-b)	76.65	**78.25**
BT-l (single)	75.57	78.67
BT-l (union)	**80.09**	**80.22**
FL (BT-l)	78.84	78.72
TM-FL (BT-l)	77.99	79.78

Table 7: F1 scores of our TM-FL models and the baselines (i.e., single domain model, union domain model and standard FL) under different settings with BERT-base and BERT-large on the development set of LAP14, REST14. "BT-b" and "BT-l" refer to encoder BERT-base and BERT-large respectively.

Models	**LAP14**		**REST14**		**TWITTER**	
	μ	σ	μ	σ	μ	σ
BT-b (single)	73.15	0.18	76.02	0.17	70.70	0.24
BT-b (union)	76.21	0.44	**78.30**	0.34	**74.40**	0.33
FL (BT-b)	74.90	0.15	76.51	0.37	72.42	0.36
TM-FL (BT-b)	**76.45**	0.31	76.99	0.46	72.69	0.40
BT-l (single)	73.98	0.47	77.47	0.29	71.98	0.36
BT-l (union)	**79.60**	0.14	**79.50**	0.43	**75.90**	0.21
FL (BT-l)	77.77	0.42	78.04	0.12	72.95	0.39
TM-FL (BT-l)	79.00	0.21	78.76	0.15	73.44	0.23

Table 8: The mean (μ) and standard deviation (σ) of F1 scores of our TM-FL model and baselines on the test set of LAP14, REST14 and TWITTER for aspect-based sentiment analysis. "BT-b" and "BT-l" refer to encoder BERT-base and BERT-large respectively.

[17] TWITTER does not have an official dev set.