Industrial Federated Topic Modeling

DI JIANG, AI Group, WeBank Co., Ltd., China
YONGXIN TONG, State Key Laboratory of Software Development Environment, Beihang University, China
YUANFENG SONG†, AI Group, WeBank Co., Ltd., China
XUEYANG WU‡, Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong
WEIWEI ZHAO, JINHUA PENG, RONGZHONG LIAN, QIAN XU, and QIANG YANG†, AI Group, WeBank Co., Ltd., China

Probabilistic topic modeling has been applied in a variety of industrial applications. Training a high-quality model usually requires a massive amount of data to provide comprehensive co-occurrence information for the model to learn. However, industrial data such as medical or financial records are often proprietary or sensitive, which precludes uploading to data centers. Hence, training topic models in industrial scenarios using conventional approaches faces a dilemma: A party (i.e., a company or institute) has to either tolerate data scarcity or sacrifice data privacy. In this article, we propose a framework named *Industrial Federated Topic Modeling* (iFTM), in which multiple parties collaboratively train a high-quality topic model by simultaneously alleviating data scarcity and maintaining immunity to privacy adversaries. iFTM is inspired by federated learning, supports two representative topic models (i.e., Latent Dirichlet Allocation and SentenceLDA) in industrial applications, and consists of novel techniques such as private Metropolis-Hastings, topic-wise normalization, and heterogeneous model integration. We conduct quantitative evaluations to verify the effectiveness of iFTM and deploy iFTM in two real-life applications to demonstrate its utility. Experimental results verify iFTM's superiority over conventional topic modeling.

CCS Concepts: • **Information systems → Data mining**; • **Computing methodologies → Topic modeling**;

Additional Key Words and Phrases: Topic models, federated learning, differential privacy

This is an extended and revised version of a conference paper published at CIKM 2019 [21].
†Also with the Hong Kong University of Science and Technology.
‡Work done when he worked as an intern at AI Group, Webank Co., Ltd.
This work is partially supported by the National Key Research and Development Program of China under Grant No. 2018AAA0101100, National Science Foundation of China (NSFC) under Grant Nos. 61822201 and U1811463, and State Key Laboratory of Software Development Environment (Beihang University) Open Program (SKLSDE-2020ZX-07).
Authors' addresses: D. Jiang, Y. Song, W. Zhao, J. Peng, R. Lian, Q. Xu, and Q. Yang, AI Group, WeBank Co., Ltd., China; emails: dijiang@webank.com, songyf@ese.ust.hk, {davezhao, kinvapeng, ronlian, qianxu}@webank.com, qyang@cse.ust.hk; Y. Tong (corresponding author), State Key Laboratory of Software Development Environment, Beihang University, China; email: yxtong@buaa.edu.cn; X. Wu, Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong; email: xwuba@cse.ust.hk.

2157-6904/2021/01-ART2 $15.00
https://doi.org/10.1145/3418283

ACM Reference format:
Di Jiang, Yongxin Tong, Yuanfeng Song, Xueyang Wu, Weiwei Zhao, Jinhua Peng, Rongzhong Lian, Qian Xu, and Qiang Yang. 2021. Industrial Federated Topic Modeling. *ACM Trans. Intell. Syst. Technol.* 12, 1, Article 2 (January 2021), 22 pages.
https://doi.org/10.1145/3418283

1 INTRODUCTION

Probabilistic topic modeling has been successfully used in many industrial applications, from military analysis [36] to web search log mining [6, 19, 20] to bioinformatics [1, 23, 31]. As training a high-quality topic model for a specific application typically requires comprehensive data to provide sufficient co-occurrence information, relying on data collected from a single party faces the challenge of data scarcity. Meanwhile, since these data are typically proprietary and sensitive, regulations such as the newly enforced European Union General Data Protection Regulation (GDPR) [5, 37, 40] may preclude uploading them to data centers and being utilized in a centralized approach. These two critical problems pose new challenges to conventional topic modeling, which we refer to as the state-of-the-art distributed architectures [30, 48, 49] for training topic models on computer clusters within a data center.

To solve the above problems, a new probabilistic topic modeling paradigm simultaneously alleviates data scarcity and ensures that data privacy is urgently needed in the industry. However, the vast discrepancy between the scenario of conventional topic modeling and that studied in this article results in three challenging research issues. First, how to protect the privacy of training data of each party from adversaries. Privacy is typically neglected in conventional topic modeling, and anyone who can access computing nodes or monitor network communication can quickly get a glimpse of the data of each party. New data regulations increasingly forbid such practice. Second, how to reduce the communication cost between computing nodes. Conventional topic modeling such as those deployed upon MapReduce [49] or ParameterServer [48] usually has a demanding requirement of communication efficiency that is only satisfied by a data-center-grade network. However, in the present problem, different parties may be located in different data centers and connected by low bandwidth. Hence, it is infeasible to allow computing nodes to communicate with each as before frequently. Third, how to handle the variety of data and models across different parties. Conventional topic modeling relies upon the assumption that different computing nodes store independent and identically distributed (i.i.d.) data and train the same topic model. However, this requirement can hardly be met in the present problem where each party usually has highly unbalanced data and trains heterogeneous topic models (i.e., topic models with different regularity).

Inspired by the concept of federated computation, which refers to a distributed architecture that a master coordinates a fleet of parties to compute aggregated statistics of private data [17, 29], we propose a framework named *Industrial Federated Topic Modeling* (iFTM) that solves the aforementioned problems in a principled approach. As shown in Figure 1, iFTM is composed of two computational components: party computation and master computation. Party computation provides a flexible mechanism for balancing model utility and data privacy. It seamlessly integrates differential privacy with Markov Chain Monte Carlo (MCMC) [14] for both private and efficient parameter inference. The local model of each party is encrypted by a topic-wise normalization mechanism and transmitted to the master without leakage of critical information of the training datasets. Master computation is responsible for integrating the transmitted local models into a global one and formalizing necessary information for meta-learning in the next iteration. Notably, master computation circumvents the rigid requirements such as frequent network communication and training the same topic model on every party, to achieve significantly lower communication cost and handle data that are not i.i.d.. In this article, we discuss the technical details of

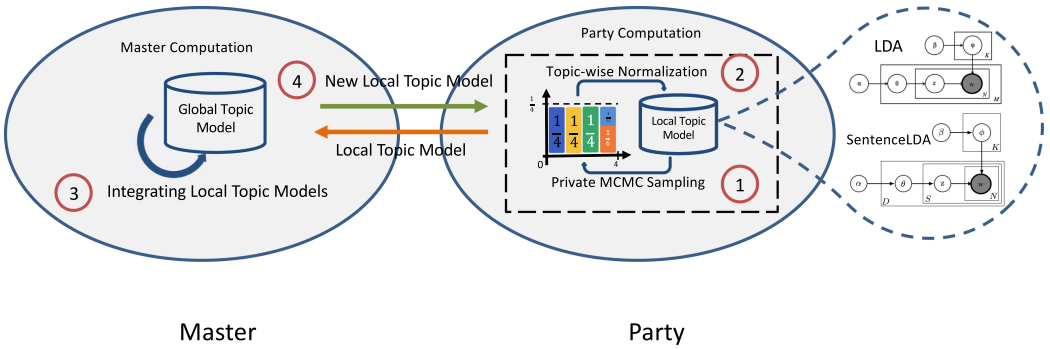


Fig. 1. The Industrial Federated Topic Modeling Framework: Party Computation (① Private MCMC Sampling and ② Topic-wise Normalization) and Master Computation (③ Integrating Local Topic Models and ④ Composing New Local Topic Models).

iFTM through two representative topic models widely used in industry: Latent Dirichlet Allocation (LDA) [4] and SentenceLDA [22], which paves the way for training a broader scope of topics models in the federated scenario. We evaluate iFTM in terms of quantitative metrics such as likelihood and communication cost as well as two real-life applications. The contributions of this article are summarized as follows:

- To the best of our knowledge, iFTM is the first framework that is specifically designed for large-scale distributed topic modeling with a guarantee of privacy protection.
- iFTM pioneers new topic modeling paradigms such as allowing heterogeneous topic models to be trained on data that are not i.i.d. across different parties.
- iFTM delicately avoids the demanding network communication that plagues conventional topic modeling, making it the first topic modeling framework applicable in federated scenarios.
- Quantitative evaluations and real-life applications demonstrate the necessity and effectiveness of iFTM.

The rest of the article is organized as follows: We review the related work in Section 2. Then, we discuss the technical details of iFTM in Section 3. We present the experimental results in Section 4 and finally conclude the article in Section 5.

2 RELATED WORK

The present work is related to a broad range of literature. We review the most related works in topic modeling, federated learning, and differential privacy, respectively.

2.1 Topic Modeling

Topic modeling has been intensively studied and widely used in industry for the past decade. LDA and SentenceLDA play important roles in the industry. In this field, recent advancement focuses on the design of new model structures and efficient inference methods under distributed environments [30, 46, 48, 49]. Typical extensions of LDA include Supervised LDA [28], TOT [41], Multifaceted Topic Model [39], and so on. However, in these conventional topic models, the issue of privacy is typically neglected. A recent work [34] proposes a technique to privatize the parameters of variational inference. However, this technique is based upon a single computing node and is not straightforward to apply in distributed computing. Although privacy and distributed computing are two critical factors determining the applicability of topic models in the industry, they have

been studied independently in existing work, and integrating them in a unified framework is still an open problem.

2.2 Federated Learning

While user-oriented topic modeling helps enhance the performance of many applications, user data is exposed to algorithms, which raises concerns about data privacy and security. This kind of dilemma happens in many real-world applications that need large-scale user data for training. In practice, data is fragmented and isolated distributed. The scarcity of data resources encourages researchers and engineers to explore approaches to collect and enlarge the dataset. Typically, there are two types of strategies: directly gathering data from multiple sources and collaboratively training a joint model across different institutions [45].

However, the increasing concern of data privacy and security has become a significant obstacle before applying these methods. For example, the European Union (EU) enforced the EU General Data Protection Regulation (GDPR) [38] in 2018 strictly restricted the personal data usage of institutions. The US, China, and other countries/regions also published their bills and laws for data protection, which means traditional methods of gathering data or collaborative learning may violate such regulations. The Cyber Security Law and the General Principles of the Civil Law, enacted in China, requires that data owners must not leak or tamper with the personal information that they collect. Companies or organizations need to ensure that third-parties conducting data transactions also follow legal data protection obligations.

To meet the legal requirements and satisfy users' expectations of privacy protection, researchers have proposed a novel machine learning paradigm, federated learning, which jointly conducts a learning algorithm across different parties (i.e., institutions, users, etc.) [29, 45]. Federated learning aims to give autonomy to each party in a collaborative union and leave the private information inside each party. To achieve this goal, researchers have to address two issues: (1) the non-IID data distribution on different parties and (2) privacy protection during collaboration.

For the first problem, Yang et al. [45] define three categories of data partitioned across different parties, i.e.,

(1) Horizontal federated learning (HFL): The samples in different parties belong to different identities with the feature from the same space. HFL is typically used in user-oriented applications, where the data are small and distributed on massive user clients, such as mobile applications. Hard et al. [17] and Leroy et al. [25] propose federated language modeling and keyword spotting for mobile phones. Their methods preserve privacy via keeping data locally.
(2) Vertical federated learning (VFL): The samples in different parties belong to the same identities with features from the same space, so these features can be virtually and vertically joint together. VFL is suitable for leveraging knowledge across different industries [8]. In this case, VFL helps complementation among different parties.
(3) Federated transfer learning (FTL): The sample is different in different parties, which combines transfer learning [32] with federated learning. FTL tries to dig the potential knowledge shared across parties even though there is no explicit alignment [27].

In terms of privacy and security protection, there are various technologies for different needs of protection level. Instead of only keeping data locally and sharing learned models, some researchers apply differential privacy [9] to reduce the probability of sensitive information leaks [3, 13, 16, 18, 33, 42]. Moreover, secure multi-party computation techniques, such as homomorphic encryption (HE) [35] and garbled circuit (GC) [47] are also used to encrypt the shared parameters or data.

2.3 Differential Privacy

Differential privacy (DP) is defined on the probability of leaking individual information by querying from a database. It quantitatively measures the risk of sniffing the output difference of algorithms between two inputs that differ by one sample. If the difference is large, one can detect individual privacy by constructing inputs. The formal definition of differential privacy is as follows: A randomized algorithm $\mathcal{M}(\mathbf{X})$ is considered as (ε, δ)-differentially private if

$$Pr(\mathcal{M}(\mathbf{X} \in \mathcal{S})) \leq e^{\varepsilon} Pr(\mathcal{M}(\mathbf{X}' \in \mathcal{S})) + \delta \tag{1}$$

for all $\mathcal{S} \subset \text{Range}(\mathcal{M})$ and for all *adjacent* datasets $\mathbf{X}, \mathbf{X}'$. The formula indicates that with a negligible probability, an attacker can sniff private information from the output of that randomized algorithm $\mathcal{M}(\mathbf{X})$. Adjacent datasets and sensitivity are two major elements in a DP setting, which capture the *hardness* of attacking this dataset (or model). The settings of DP are open, which allows researchers to customize privacy definition and protection degree according to the specific privacy requirement. There are many perturbation mechanisms to achieve a certain level of privacy under a given sensitivity, among which the Gaussian mechanism and Laplace Mechanism are mostly used. Laplace mechanism [10] provides ϵ-differential privacy, which adds Laplace noise to the revealed data (i.e., parameters or query outputs of a released model) with the amount of noise controlled by ϵ. Specifically, the L_1 sensitivity Δh for function h is defined as:

$$\Delta h = \max_{X, X'} ||h(X) - h(X^{'})||_1 \tag{2}$$

for all datasets $X, X^{'}$ differing in at most one element. The Laplace mechanism adds noise via:

$$\begin{aligned} \mathcal{M}_L(X, h, \epsilon) &= h(X) + (Y_1, Y_2, \ldots, Y_d), \\ Y_j &\sim Laplace(\Delta h/\epsilon), \forall j \in \{1, 2, \ldots, d\}, \end{aligned} \tag{3}$$

where d is the dimensionality of h. The $\mathcal{M}_L(X, h, \epsilon)$ mechanism is ϵ-differentially private.

A model always combines with multiple DP mechanisms. The composition of DP mechanisms will certainly accumulate the risk of leaking private information. The composition theorems [10] provides arithmetical methods to compute the consequence of composing multiple mechanisms. There are two major composition theories:

- *Sequential composition*: suppose $\mathcal{M}_j(\mathbf{X}), j = 1, \ldots, l$, are (ε_j)-differentially private, then the combination of these algorithms $\mathbf{X} \to (\mathcal{M}_1(\mathbf{X}), \ldots, \mathcal{M}_l(\mathbf{X}))$ is $(\sum_j \varepsilon_j)$-differentially private. In a special case, when all the $\mathcal{M}_j$ are homogeneous, the combination yields $(k\varepsilon, k\delta)$-differentially private.
- *Parallel composition*: Let $\mathbf{X_i}$ be arbitrary disjoint subsets of the input $\mathbf{X}$, and $\mathcal{M}_j(\mathbf{X}_j), j = 1, \ldots, l$, are (ε_j)-differentially private, then the combination of these algorithms $\mathbf{X} \to (\mathcal{M}_1(\mathbf{X_1}), \ldots, \mathcal{M}_l(\mathbf{X_l}))$ satisfies $(\max_j \varepsilon_j)$-differentially private.

Notably, when output is perturbed, any deterministic post-processing on it does not affect is differential privacy loss [10]. We leverage this property to speed up the sampling in our work.

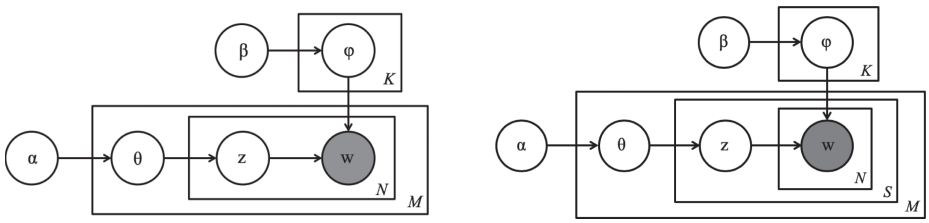
3 THE IFTM FRAMEWORK

We discuss iFTM based upon two topic models widely used in industry: LDA and SentenceLDA. The techniques discussed in this section pave the way for designing similar algorithms for other topic models. To facilitate the discussion thereafter, we list the notations that will be used in this article in Table 1. We first discuss the party computation in Section 3.1. Then, we discuss how the master computation works in Section 3.2 and finally present the workflow of iFTM in Section 3.3.

Before diving into the details of iFTM, we first highlight the difference between LDA and SentenceLDA, whose graphical models are illustrated in Figure 2. In the generative process of LDA,

Table 1. Notations for iFTM

Notation	Meaning
D	Size of documents
K	Number of topics
V	Size of vocabulary
Φ	Word distributions of topics
ϕ_k	Word distribution of topic k
Θ	Topic distributions of documents
θ_d	Topic distributions of document d
$\mathbf{w}$	Words vector of a document
w_{di}	ith word in document d
$\mathbf{z}$	Topic assignment vector of a document
z_{di}	Topic assignment of ith in document d
$\mathbf{z}_{-di}$	Topic assignment vector of document d except the ith word
k	Topic index
α	Dirichlet prior vector for θ
β	Dirichlet prior vector for φ
C_{dk}^{DK}	Number of words or sentences assigned to topic k in document d
C_{kw}^{KW}	Number of word w assigned to topic k
$C_{k\cdot}^{KW}$	Array with each element indicating the number of the corresponding word assigned to topic k
$\mathcal{M}$	Global topic model
$\mathcal{M}^*$	Updated global topic model
$\mathcal{M}_p$	Party p's local topic model



(a) LDA (b) SentenceLDA

Fig. 2. Graphical models of LDA and SentenceLDA.

each word is generated by its corresponding topic. However, in SentenceLDA, the words in the same sentence are constrained to be generated by the same topic. It is worth noting that the generative assumption of SentenceLDA is better fit for text written in natural language, since the semantic granularity of topics is typically coarser than that of the sentence.

3.1 Party Computation

Each party is the workhorse for training its local topic model. We now discuss several novel mechanisms of party computation to protect the privacy of data stored on each party. We start with

describing a private Gibbs sampling algorithm in Section 3.1.1 and Section 3.1.2 and then adapt it to a private Metropolis Hastings algorithm that enjoys much higher efficiency in Section 3.1.3 and 3.1.4. Finally, we discuss how to avoid revealing the original word distribution of the local model in Section 3.1.5.

3.1.1 Private Gibbs Sampling of LDA. Gibbs sampling is widely used for parameter inference of topic models [15, 22, 41]. Let the words $\mathbf{w}$ be the training data from a party and $\mathbf{z}$ be the latent topics assigned to $\mathbf{w}$. According to the generative assumption of LDA [4], the joint probability is as follows:

$$P(\mathbf{w}, \mathbf{z}, \Theta, \Phi|\alpha, \beta) = P(\Theta|\alpha)P(\Phi|\beta)P(\mathbf{z}|\Theta)P(\mathbf{w}|\mathbf{z}, \Phi), \tag{4}$$

where Θ are the topic distributions of documents, Φ are the word distributions of topics, α and β are hyperparameters that are usually fixed to constant values [15]. The Gibbs update of a topic z_{di} that corresponds to the ith word w in document d is defined as follows:

$$P(z_{di}|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta) \propto P(z_{di}|\theta_d)P(w_{di}|z_{di}, \Phi), \tag{5}$$

where $\mathbf{z}_{-di}$ are the latent topics except the one assigned for the ith word in d. In Equation (5), only the $P(w_{di}|z_{di}, \Phi)$ component needs to access the original data. Hence, we integrate out θ_d and the partially collapsed Gibbs update is as follows:

$$P(z_{di}|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta) \propto \frac{(C^{DK}_{dz_{di}} + \alpha)}{\sum_{k'}(C^{DK}_{dk'} + \alpha)} P(w_{di}|z_{di}, \Phi), \tag{6}$$

where $C^{DK}_{dz_{di}}$ is the number of words assigned to topic z_{di} in document d. Due to conjugacy of β and Φ, the update formula for ϕ_k is as follows:

$$P(\phi_k|\mathbf{w}, \mathbf{z}, \beta) \sim Dirichlet\left(C^{KW}_{k\cdot} + \beta\right), \tag{7}$$

where C^{KW}_{kw} is the number of w assigned to topic k and $C^{KW}_{k\cdot}$ is an array with one element indicating the number of the corresponding word assigned to topic k. We write $P(w|z_{di}, \Phi)$ in the exponential family form:

$$P(w_{di}|z_{di}, \Phi) = \phi_{z_{di}w_{di}} = \exp\left(\sum_{w'} n_{diw'} \log \phi_{z_{di}w'}\right), \tag{8}$$

where $n_{diw'} = \mathbb{I}[w' = w_{di}]$. Since the sampling algorithm interacts with the corpus only by the sufficient statistics for $\exp(\sum_{w'} n_{diw'} \log \phi_{z_{di}w'})$, we privatize the sufficient statistics (i.e., $n_{diw'}$) via the Laplace mechanism, resulting in privatized counts $\hat{n}_{diw'}$:

$$\hat{n}_{diw'} = n_{diw'} + Y. \tag{9}$$

We apply the "include/exclude" version of differential privacy, in which differing by a single entry refers to the inclusion or exclusion of that entry in the corpus. Since each counter $n_{diw'}$ is a sum of indicator vectors, it has $L1$ sensitivity of 1. We have:

$$Y \sim Laplace(1/\varepsilon). \tag{10}$$

The above formula means randomly drawing a sample from the Laplace distribution with the location parameter 0 and scale parameter $1/\varepsilon$. Since the scale parameter ε controls how much "noise" we add to the training data, in our experiments, we set its value of 8.0, 9.0, 10.0, and 11.0 to see the performance of our algorithm. Note that we only need to compute the privatized count $\hat{n}_{diw'}$ once, and it works as a proxy of the original coun in the following sampling algorithms. Hence, no original data is exposed to the sampling algorithm. According to References [43] and [11], it is easy to prove that such mechanism is ε-differentially private. After applying the Laplace mechanism, $\hat{n}_{di\cdot}$ is no longer sparse, and the complexity of Gibbs sampling via Equation (6) increases

from $O(K)$ per word to $O(KV)$ per word, where K is the number of topics and V is the size of the vocabulary. It is easy to see that the private Gibbs sampling algorithm is unrealistically inefficient for real-life applications where datasets are voluminous.

ALGORITHM 1: Private Metropolis Hastings for LDA

```
input: local training data
output: local topic model M_p
if it is the first global iteration then
    for each document d in local training data do
        for each word w_i in d do
            privatize n_{di·} according to Equation (9) and threshold by τ
        end
    end
    randomly assign a topic to each word in local corpus
else
    build a word-topic alias table based on the new local model from the master
    sample a topic for each word in local corpus according to the above word-topic alias table
    for each local iteration do
        build a doc-topic alias table according to Equation (17)
        build a word-topic alias table according to Equation (20)
        for each document d in local corpus do
            for each word w_{di} in d do
                propose a topic z_a with the doc-topic alias table;
                update z_i according to z_a and Equation (19);
                propose a topic z_b with the word-topic alias table;
                update z_i according to z_b and Equation (22);
            end
        end
        sample Φ according to Equation (7)
    end
compose local model M_p according to C^{KW}_{k·}
```

3.1.2 Private Gibbs Sampling of SentenceLDA. Similar to LDA, the joint probability for SentenceLDA [2] is as follows:

$$P(\mathbf{w}, \mathbf{z}, \Theta, \Phi|\alpha, \beta) = P(\Theta|\alpha)P(\Phi|\beta)P(\mathbf{z}|\Theta)P(\mathbf{w}|\mathbf{z}, \Phi), \tag{11}$$

where Θ are the topic distributions of documents, Φ are the word distributions of topics, α and β are hyperparameters that are usually fixed to constant values [15]. Different from LDA, the Gibbs update of a topic z_{ds} that corresponds to the sth **sentence** in document d is defined as follows:

$$P(z_{ds}|\mathbf{z}_{-ds}, \mathbf{w}, \alpha, \beta) \propto P(z_{ds}|\theta_d)P(\mathbf{w_{ds}}|z_{ds}, \Phi), \tag{12}$$

where $\mathbf{z}_{-ds}$ are the latent topics except the one assigned for the sth sentence in d. In Equation (12), only the $P(\mathbf{w_{ds}}|z_{ds}, \Phi)$ component needs to access the original data. Hence, we integrate out θ_d and the partially collapsed Gibbs update is as follows:

$$P(z_{ds}|\mathbf{z}_{-ds}, \mathbf{w}, \alpha, \beta) \propto \frac{(C^{DK}_{dz_{ds}} + \alpha)}{\sum_{k'}(C^{DK}_{dk'} + \alpha)} P(\mathbf{w_{ds}}|z_{ds}, \Phi), \tag{13}$$

where $C^{DK}_{dz_{ds}}$ is the number of sentences assigned to topic z_{ds} in document d. Due to conjugacy of β and Φ, the update formula for ϕ_k is as follows:

$$P(\phi_k|\mathbf{w}, \mathbf{z}, \beta) \sim Dirichlet\left(C^{KW}_{k\cdot} + \beta\right), \tag{14}$$

where C^{KW}_{kw} is the number of word w assigned to topic k and $C^{KW}_{k\cdot}$ is an array with one element indicating the number of the corresponding word assigned to topic k. We write $P(\mathbf{w}|z_{ds}, \Phi)$ in the exponential family form:

$$P(\mathbf{w_{ds}}|z_{ds}, \Phi) \approx \prod_{w \in \mathbf{w_{ds}}} \phi_{z_{ds}w} = \prod_{w \in \mathbf{w_{ds}}} \exp\left(\sum_{w'} n_{dsw'} \log \phi_{z_{ds}w'}\right), \tag{15}$$

where $n_{dsw'} = \mathbb{I}[w' = w]$. Similar to the Private Gibbs Sampling algorithm for LDA, the sampling algorithm for SentenceLDA also interacts with the corpus only by the sufficient statistics for $\exp(\sum_{w'} n_{dsw'} \log \phi_{z_{ds}w'})$, and we privatize the sufficient statistics (i.e., $n_{dsw'}$) via the Laplace mechanism, resulting in privatized counts $\hat{n}_{dsw'}$:

$$\hat{n}_{dsw'} = n_{dsw'} + Y. \tag{16}$$

3.1.3 Private Metropolis-Hastings of LDA. To improve the efficiency of MCMC sampling and make it applicable to the massive dataset, we propose the private Metropolis-Hastings (MH) for LDA, and the algorithm is depicted in Algorithm 1.

Being the same as traditional MH, the private MH algorithm has two deliberately designed proposals for proposing a topic candidate for a word. The first proposal is the doc-topic proposal:

$$\Omega^z_d = \frac{(C^{DK}_{dz} + \alpha)}{\sum_{k'}(C^{DK}_{dk'} + \alpha)}, \tag{17}$$

where Ω^z_d can be straightforwardly interpreted as the "strength" of the relation between z and d.

For doc-topic proposal, the acceptance probability of topic transition from z to z' is:

$$min\left\{1, \frac{P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^z_d}{P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^{z'}_d}\right\}. \tag{18}$$

By replacing the component $P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)$ and $P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)$ with Equation (6), the above acceptance probability is updated as follows:

$$min\left\{1, \frac{(\hat{C}^{DK}_{dz'} + \alpha)\hat{P}(w|z', \Phi)(C^{DK}_{dz} + \alpha)}{(\hat{C}^{DK}_{dz} + \alpha)\hat{P}(w|z, \Phi)(C^{DK}_{dz'} + \alpha)}\right\}, \tag{19}$$

where the hat notation means that the statistics of w_{di} is removed from the corresponding value.

The second one is word-topic proposal, which is defined as:

$$\Omega^z_w = \frac{C^{KW}_{zw} + \beta}{\sum_{w'}(C^{KW}_{zw'} + \beta)}, \tag{20}$$

where Ω^z_w can be straightforwardly interpreted as the "strength" of relation between topic z and w.

For word-topic proposal, the acceptance probability of topic transition from z to z' is:

$$min\left\{1, \frac{P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^z_w}{P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^{z'}_w}\right\}. \tag{21}$$

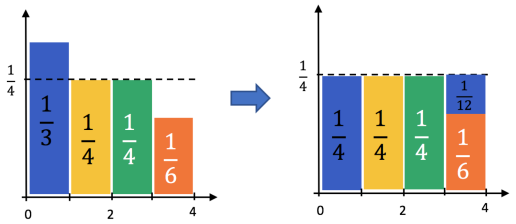


Fig. 3. A toy example of building alias table.

By replacing the component $P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)$ and $P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)$ with Equation (6), the above acceptance probability is updated as follows:

$$min\left\{1, \frac{(\hat{C}^{DK}_{dz'} + \alpha)\hat{P}(w|z', \Phi)(C^{KW}_{zw} + \beta)(\sum_{w'}(C^{KW}_{z'w'} + \beta))}{(\hat{C}^{DK}_{dz} + \alpha)\hat{P}(w|z, \Phi)(C^{KW}_{z'w} + \beta)(\sum_{w'}(C^{KW}_{zw'} + \beta))}\right\}, \tag{22}$$

where the hat notation means that the statistics of w_{di} is removed from the corresponding value.

The strategies for improving the sampling efficiency of private MH are twofold:

(1) Improving the sampling efficiency of the proposals of Equation (17) and Equation (20). To achieve this goal, we build a doc-topic alias table and a word-topic alias table for the two proposals, respectively, according to the alias method in Reference [48]. The key idea of the alias method is the construction of the alias table, which is illustrated by an example in Figure 3. During the construction process, the algorithm keeps moving "overfull" entries (entry one in the example) to the "underfull" entries (entry four in the example) in the table to make all the entries "exactly full." In the meantime, it guarantees each entry has at most two kinds of entry index. With alias method, the original non-uniform sampling process is transformed into a uniform one, and the time complexity of sampling a topic from a proposal is reduced from $O(K)$ per word to $O(1)$ per word. When sampling a new topic for a word, the doc-topic proposal and word-topic proposal are sequentially applied to achieve a high mixing rate.

(2) Reducing the computational cost of calculating the acceptance probabilities of Equation (19) and Equation (22). The bottleneck of calculating Equation (19) and Equation (22) lies in the component $P(w|z_{di}, \Phi)$. We utilize a threshold τ to sparsify the vector $\hat{n}_{di\cdot}$. As $\hat{n}_{diw_i}$ represents the count information, we clap $\hat{n}_{diw_i}$ to zero if $\hat{n}_{diw_i} \leq \tau$.

By collectively applying the above two strategies, the amortized time complexity of sampling a topic for a word by private MH can be reduced to $O(\frac{V}{2e^{\tau\varepsilon}})$. According to Reference [10], applying deterministic post-processing to a ε-differentially private mechanism is still ε-differentially private. Therefore, the above operation does not affect the privacy guarantee.

3.1.4 Private Metropolis Hastings of SentenceLDA. Similar to private MH for LDA depicted in Section 3.1.3, our propose private MH algorithm for SentenceLDA also has two proposals for proposing a topic candidate for a word. The first proposal is the doc-topic proposal:

$$\Omega^z_d = \frac{(C^{DK}_{dz} + \alpha)}{\sum_{k'}(C^{DK}_{dk'} + \alpha)}, \tag{23}$$

where Ω^z_d can be straightforwardly interpreted as the "strength" of the relation between z and d.

ALGORITHM 2: Private Metropolis Hastings for SentenceLDA

input: local training data
output: local topic model $\mathcal{M}_p$
if *it is the first global iteration* **then**
  for *each document d in local training data* **do**
    for *each word* w_i *in d* **do**
      privatize $n_{di\cdot}$ according to Equation (16) and threshold by τ
    end
  end
  randomly assign a topic to each word in local corpus
else
  build a word-topic alias table based on the new local model from the master
  sample a topic for each word in local corpus according to the above word-topic alias table
  for *each local iteration* **do**
    build a doc-topic alias table according to Equation (23)
    build a word-topic alias table according to Equation (25)
    for *each document d in local corpus* **do**
      propose a sentence-topic z_s with the doc-topic alias table;
      update z_i according to z_s and Equation (27);
      for *each word* w_{dsi} *in sentence s* **do**
        propose a topic z_{sw} with the word-topic alias table;
        update z_{si} according to z_{sw} and Equation (28);
      end
    end
    sample Φ according to Equation (14)
  end
compose local model $\mathcal{M}_p$ according to $C^{KW}_{k\cdot}$

For doc-topic proposal, the acceptance probability of topic transition from z to z' is:

$$min\left\{1, \frac{P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^z_d}{P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^{z'}_d}\right\}. \tag{24}$$

The second one is word-topic proposal, which is defined as:

$$\Omega^z_w = \frac{C^{KW}_{zw} + \beta}{\sum_{w'}(C^{KW}_{zw'} + \beta)}, \tag{25}$$

where Ω^z_w can be straightforwardly interpreted as the "strength" of relation between topic z and w.

For word-topic proposal, the acceptance probability of topic transition from z to z' is:

$$min\left\{1, \frac{P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^z_w}{P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^{z'}_w}\right\}. \tag{26}$$

By replacing the component $P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)$ and $P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)$ with Equation (13), the above two acceptance probabilities (Equation (24) and Equation (26)) are updated as follows:

$$min\left\{1, \frac{(\hat{C}^{DK}_{dz'} + \alpha)\hat{P}(w|z', \Phi)(C^{DK}_{dz} + \alpha)}{(\hat{C}^{DK}_{dz} + \alpha)\hat{P}(w|z, \Phi)(C^{DK}_{dz'} + \alpha)}\right\}, \tag{27}$$

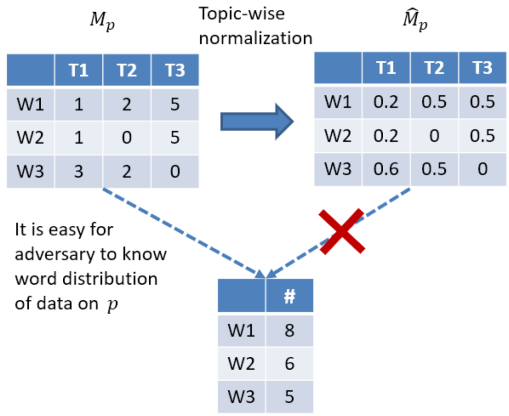


Fig. 4. A Toy example of topic-wise normalization. Hyperparameters are neglected in this example for simplicity.

where the hat notation means that the statistics of w_{di} is removed from the corresponding value, and

$$min\left\{1, \frac{(\hat{C}^{DK}_{dz'} + \alpha)\hat{P}(w|z', \Phi)(C^{KW}_{zw} + \beta)(\sum_{w'}(C^{KW}_{z'w'} + \beta))}{(\hat{C}^{DK}_{dz} + \alpha)\hat{P}(w|z, \Phi)(C^{KW}_{z'w} + \beta)(\sum_{w'}(C^{KW}_{zw'} + \beta))}\right\}, \tag{28}$$

where the hat notation means that the statistics of w_{di} are removed from the corresponding value.

3.1.5 Topic-wise Normalization. As for either LDA or SentenceLDA, the local model (i.e., the result of Algorithm 1 or Algorithm 2) of a party p can be represented as a word-topic matrix $\mathcal{M}_p$, in which each cell stores the frequency count of the corresponding word and topic. The information in $\mathcal{M}_p$ should be transmitted to the master through network communication. As shown in Figure 4, transmitting $\mathcal{M}_p$ exposes word distribution of the training data on p, since some important information may be recovered by $\mathcal{M}_p$ and deliberately designed language models. To solve this problem, we conduct topic-wise normalization and obtain the normalized word-topic matrix $\hat{\mathcal{M}}_p$, which is transmitted to the master. As $\mathcal{M}_p$ and $\hat{\mathcal{M}}_p$ result in exactly the same alias tables used in Algorithm 1, $\hat{\mathcal{M}}_p$ can be considered as the result of a lossless encryption mechanism, which protects the original word distribution of the training data on p.

3.2 Master Computation

Since the formats of Φ in LDA and SentenceLDA are the same, their corresponding master computations are the same as well. The duties of the master are twofold: integrating the local models from different parties and composing a new local model for each party. We first discuss how to integrate heterogeneous local topic models in Section 3.2.1. Then, we discuss the approach of composing a new local topic model for each party in Section 3.2.2. It is worth noting that master computation effectively handles topic models with different regularities and therefore is suitable for scenarios where data are not i.i.d. across parties.

3.2.1 Integrating Local Topic Models. Since the data of different parties are not necessarily i.i.d., the local models from different parties may contain different amounts of topics. The master is responsible for integrating these heterogeneous topic models.To compose a global model $\mathcal{M}^*$ based on the topics in local models, we rely on Weighted Jaccard Similarity to calculate the similarity between topics and merge the similar ones. The similarity between two topics z_i and z_j is defined

as:

$$\begin{aligned}\rho(z_i, z_j) =& \frac{\sum_{l=1}^{m} \min(p_{w_l}^{z_i}, p_{w_l}^{z_j})}{\sum_{l=1}^{m} \max(p_{w_l}^{z_i}, p_{w_l}^{z_j}) + \sum_{m+1}^{T} p_{w_l}^{z_i} + \sum_{l=m+1}^{T} p_{w_l}^{z_j}} \\ =& \frac{\sum_{l=1}^{m} \min(p_{w_l}^{z_i}, p_{w_l}^{z_j})}{\sum_{l=1}^{T} p_{w_l}^{z_i} + \sum_{l=1}^{T} p_{w_l}^{z_j} - \sum_{l=1}^{m} \min(p_{w_l}^{z_i}, p_{w_l}^{z_j})},\end{aligned} \tag{29}$$

where $P_{z_i} = (p_{w_1}^{z_i}, p_{w_2}^{z_i}, \ldots, p_{w_m}^{z_i}, p_{w_{m+1}}^{z_i}, \ldots, p_{w_L}^{z_i})$ and $P_{z_j} = (p_{w_1}^{z_j}, p_{w_2}^{z_j}, \ldots, p_{w_m}^{z_j}, p_{w_{m+1}}^{z_j}, \ldots, p_{w_L}^{z_j})$ are vectors representing the top-L words distribution of topic z_i and topic z_j. m $(0 \leq m \leq L)$ indicates the count of common words in their top-L words. Two topics are considered as redundant if the similarity $\rho(z_i, z_j)$ is beyond the threshold ξ. The threshold $\rho(z_i, z_j)$ is set empirically based on our experience in constructing high quality topic models.

Based on the above similarity metric, we detail the mechanism of integrating local models in Algorithm 3. The algorithm first concatenates two topic models (Line 2). Then it finds the redundant topic sets based on the Union-Find [12] algorithm (Line 2~11). For example, if (z_1, z_2) and (z_2, z_3) are considered as redundant based on Equation (29), $\{z_1, z_2, z_3\}$ will be taken as a disjoint topic set. For each topic set, we then merge the topics in the set to get the representative distribution (Lines 12~16) by adding each topic distribution sequentially and do the normalization. (In this case, the normalized distribution $\frac{w_1\vec{z_1}+w_2\vec{z_2}+w_3\vec{z_3}}{w_1+w_2+w_3}$ is chosen with $\vec{z_1}$, $\vec{z_2}$ and $\vec{z_3}$ removed from $\mathcal{M}^B$.) Since the data of different parties are highly unbalanced, we assign different weights w_i to the topics based on the data amount n_i of different parties. Finally, we can obtain the global topic model M^* (Line 18).

ALGORITHM 3: Integrating Local Topic Models

input: global topic model $\mathcal{M}$, local topic model $\mathcal{M}_p$.
output: updated global topic Model $\mathcal{M}^*$.

```
1  begin
2      concatenate M and M_p into M^B;
3      redundant topics R = {}
4      for each topic z_i in M^B do
5          for each topic z_j (j > i) in M^B do
6              calculate ρ(z_i, z_j) with Equation (29);
7              if ρ(z_i, z_j) ≥ ξ then
8                  Add (z_i, z_j) into R
9              end
10         end
11     end
12     for each set s in Union-Find(R) do
13         for each topic z_si (i > 1) in s do
14             add w_si z_si to z_s1, remove z_si from M^B;
15         end
16         normalize distribution z_s1;
17     end
18     M* = M^B
19 end
20 return M*;
```

3.2.2 Composing New Local Topic Models. Since the global topic model M^* is large and comprehensive, some topics in M^* are irrelevant to the data of certain parties. Hence, it is unnecessary to push all the information in M^* to each party. To effectively reduce the communication cost, we compose a new local model that is compact enough to be pushed to the corresponding party. To take full advantage of the global model to facilitate local training, we employ *meta-learning*[1] [26] to transfer meta-level knowledge (i.e., the topics of M^*) as high-quality initialization for next-iteration local training. Specifically, we scan each topic z_p in $\hat{M}_p$, choose the most similar topic z from the global topic model M^*, replace z_p with z in the new local topic model $M_p^{'}$, and push it to p. The algorithm of composing new local models is presented in Algorithm 4.

ALGORITHM 4: Composing New Local Topic Models

input: global topic model M^*, local topic model $\hat{M}^p$
output: new local topic model M'_p.

```
begin
    for each topic z_p in M̂_p do
        z = arg max_{z∈M*} ρ(z, z_p);
        if ρ(z, z_p) ≥ ξ then
            replace z_p with z into M'_p;
        end
        remove z from M*;
    end
end
return M'_p;
```

3.3 iFTM Workflow

The workflow of iFTM is presented in Algorithm 5. For each global iteration, during the party computation stage, each party trains a local topic model and pushes the local topic model to the master. During the master computation stage, the master sequentially merges all local topic models, maintains a global topic model M^* according to Algorithm 3, and composes and pushes new local models for each party according to Algorithm 4. The whole process repeats for a predefined number of global iterations. We will show later that few global iterations are sufficient to obtain a good global topic model M^*. The low synchronization frequency improves iFTM's to low bandwidth and network failures, which are more common in wide area networks than in data centers.

4 EXPERIMENTS

In this section, we evaluate the performance of iFTM in terms of both quantitative metrics and applications. In Section 4.1, we describe the experimental setup. In Section 4.2, we demonstrate the effectiveness of iFTM in alleviating data scarcity. In Section 4.3, we demonstrate the utility of iFTM in terms of different parameter settings. In Section 4.4, we gauge the communication cost of iFTM. Finally, we show the necessity and the promising performance of iFTM through real-life applications in Section 4.5 and Section 4.6.

[1]Meta-learning, also named learning to learn, is previously utilized in a supervised learning scenario. Meta-learning normally includes learning at two levels: higher-level learning to gain meta-knowledge and lower-level learning for new tasks directed by meta-knowledge [26].

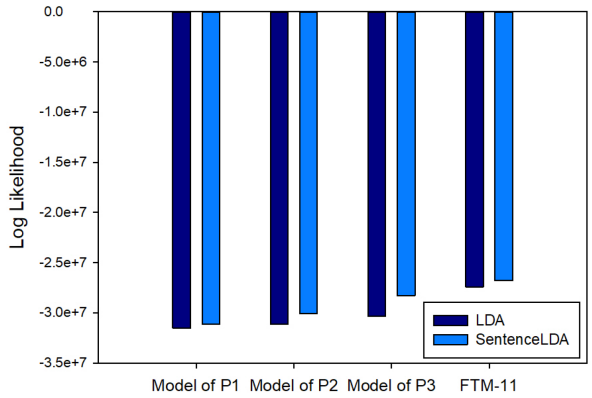


Fig. 5. Performance ofdata scarcity alleviation.

ALGORITHM 5: iFTM Workflow

```
for each global iteration do
    for each client p do
        train local topic model
        push local topic model M̂_p to master
    end
    integrate local topic models and obtain the global topic model M* according to Algorithm 3
    for each client p do
        compose new local topic models for p push new local topic model M'_p to p
    end
end
return the global topic model M*
```

4.1 Experimental Setup

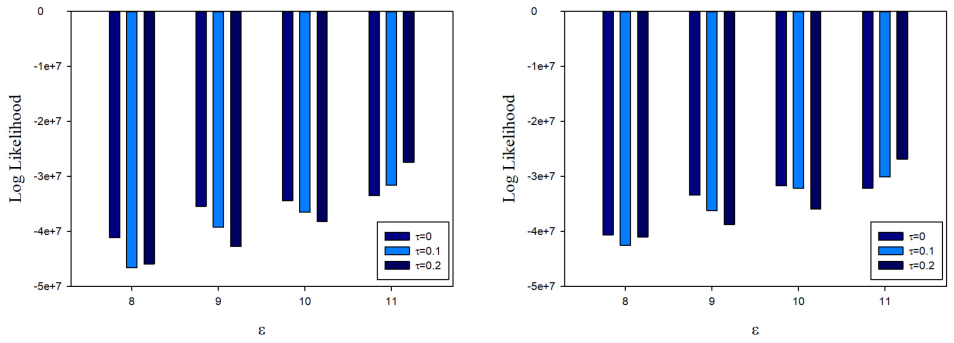
We assume that there are three parties denoted by P_1, P_2, and P_3, whose data are neither balanced nor i.i.d. Specifically, P_1, P_2, and P_3 store 29,723, 59,445, and 89,169 documents, respectively. A corpus containing other 29,700 documents is used as the testing data. LDA is trained through the LightLDA[2] toolkit and SentenceLDA is analogously trained. The number of topics has been tuned for each party to make them strong baselines.

4.2 Data Scarcity Alleviation

One major motivation of iFTM is to alleviate the data scarcity problem faced by each party. Hence, one important question is whether the model trained by iFTM is better than those trained by a single party relying on its data.

Figure 5 shows the comparison of the two topic models trained by iFTM and different parties in terms of the log-likelihood of testing data. We observe that harnessing more data usually results in better LDA models. By collectively utilizing data from all parties, iFTM achieves the highest likelihood. For example, iFTM-11 (i.e., iFTM with $\varepsilon = 11$ and $\tau = 0.2$) demonstrate the log-likelihood of -2.74×10^7 while the best LDA model trained by a single party is the one from P_3 and only achieves a log-likelihood of -3.03×10^7. Similar to LDA, more data usually results in better SentenceLDA

[2]https://github.com/Microsoft/LightLDA.



(a) LDA (b) SentenceLDA

Fig. 6. Performance of privacy protection.

models. By using the data from all parties, the SentenceLDA trained by iFTM achieves the highest likelihood. Similar results can be observed for both LDA and SentenceLDA with other settings of ε and τ. The result indicates that iFTM is effective in alleviating data scarcity and generates high-quality topic models that cannot be obtained based upon a single party's data.

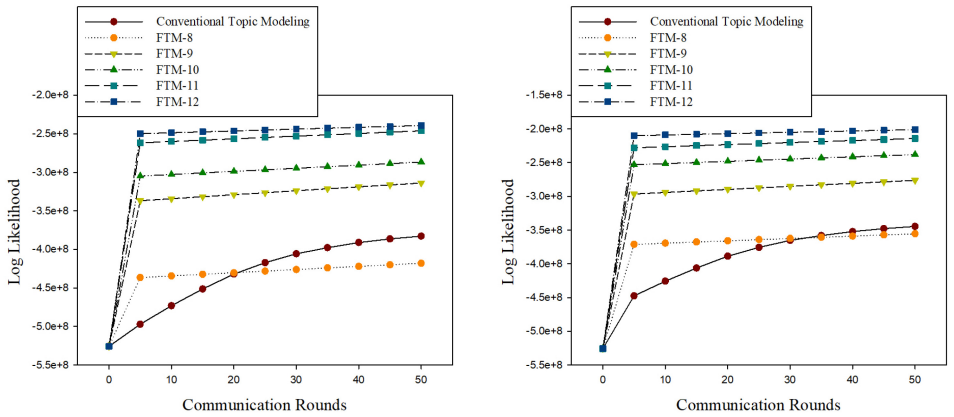
4.3 Privacy Protection

The performance of iFTM with different ε reflects the utility of iFTM after different levels of privacy protection.

Figure 6(a) shows the performance of LDA trained by iFTM with different ε (i.e., the scale parameter for Laplace distribution) and τ (i.e., the threshold for sparsifying vector $\hat{n}_{di\cdot}$). As ε increases, LDA trained by iFTM usually achieves a higher likelihood of the testing data. For example, $\varepsilon = 8$ and $\tau = 0.2$ achieve a log-likelihood of -4.59×10^7. When ε increases to 11, iFTM achieves a log-likelihood of -2.74×10^7. This observation is quite straightforward, since ε determines how much "noise" we add to the training data. In contrast, the effect of τ is more complicated, since it simultaneously affects the "noise" and the original data. When the "noise" is relatively moderate (e.g., $\varepsilon = 11$), a slightly higher τ (e.g., $\tau = 0.2$) will clap most of the noisy elements in $\hat{n}_{di\cdot}$ to zero and results in models with higher likelihood on testing data. Figure 6(b) shows the performance of SentenceLDA trained by iFTM. As ε increases, SentenceLDA trained by iFTM achieves a higher likelihood of the testing data. This observation is the same as LDA, since ε determines the level of "noise" we inject to training data. As for τ, when the "noise" is moderate (e.g., $\varepsilon = 11$), a slightly higher τ (e.g., $\tau = 0.2$) claps most of the noisy elements in $\hat{n}_{di\cdot}$ to zero and results in better SentenceLDA model. Empirically, for both LDA and SentenceLDA, $\tau = 0.2$ demonstrates a fairly good performance with moderate noise, and we utilize it by default in iFTM. As LDA and SentenceLDA differ by the prior distribution of word-topic sampling, the performance consistency of LDA and SentenceLDA indicates that the effect of our privacy protection is limited in sampling.

4.4 Communication Cost

Figure 7(a) presents the communication costs of LDA trained by conventional topic modeling and iFTM with different ε. The baselines conventional topic modeling trained by LightLDA on a dataset consisting of the training data from P_1, P_2, and P_3. We observe that iFTM converges quickly within several rounds of communication, while conventional topic modeling demonstrates the much slower speed of convergence. iFTM with higher ε demonstrates superior performance in terms of model quality and communication efficiency. When ε is lower than 9, the final model



(a) LDA (b) SentenceLDA

Fig. 7. Likelihood versus communication.

of iFTM is slightly worse than conventional topic modeling. Conventional topic modeling usually needs more than 300 rounds of communications to achieve the likelihood that can be achieved by iFTM in less than 5 rounds. These results verify the superiority of iFTM in a low-bandwidth environment. Another interesting observation is that introducing moderate noise is beneficial for improving the model's quality under training. When ε is set to a value larger than 8, the models trained by iFTM achieve a higher likelihood than that trained by conventional topic modeling on original data. Figure 7(b) presents the communication costs of SentenceLDA trained by conventional topic modeling and iFTM with different ε. Similar to the case of LDA, the SentenceLDA model trained by iFTM converges quickly within several rounds of communication, and this convergence rate is much faster than its conventional counterparts. The SentenceLDA model trained by iFTM with higher ε demonstrates superior performance in terms of model quality and communication efficiency. These results again verify the superiority of iFTM in a low-bandwidth environment. To shed light on how iFTM works when the number of parties increases, we re-allocate the documents from P_1, P_2, and P_3 to four parties P_1', P_2', P_3', and P_4', which store 28,000, 57,000, 80,000 and 13,337 documents, respectively. The parameter settings of the four-party scenario is exactly the same as the three-party scenario. We find that with more parties, the framework needs about 15 rounds to generate a fairly good topic model, showing that the communication cost inevitably increases as more parties involved. However, such communication cost is still much lower than that caused by using ParameterServer in conventional topic modeling.

4.5 iFTM in Automatic Speech Recognition

Topic models are known for effectively improving the performance of Automatic Speech Recognition (ASR) systems by providing richer contextual information for the language model (LM) component in ASR [7, 44]. Specifically, topic models are utilized to calculate the probability of seeing a word given the context:

$$P_{TM}(w|C) = \sum_z P(w|z)P(z|C), \tag{30}$$

where z is the latent topic, $P(w|z)$ is word probability given the topic, and $P(z|C)$ is topic probability given the context C. Comparing with the traditional backoff n-gram language models, such topic-based approach is able to predict word probability based on much longer history and richer

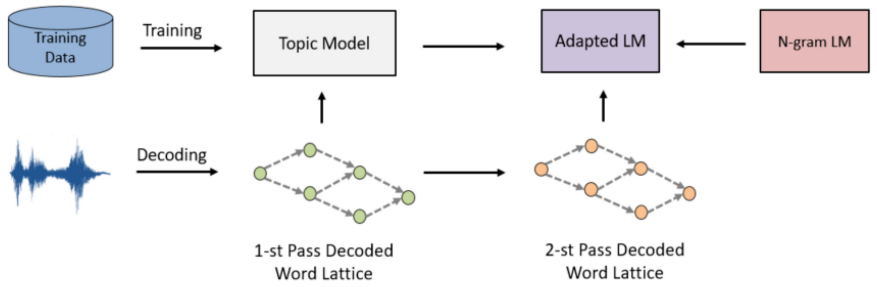


Fig. 8. The pipeline of applying topic model in ASR.

semantic information. In practice, we conduct a linear interpolation between the traditional backoff n-gram language model and that produced by Equation (30) to generate the adapted language model $P(w|C)$:

$$P(w|C) = \lambda P_{TM}(w|C) + (1 - \lambda)P_{LM}(w|C), \tag{31}$$

where $P_{LM}(w|C)$ is the probability given by the traditional backoff n-gram language model; λ is a tradeoff parameter and set empirically to 0.1 in our experiments. The pipeline of applying topic models in ASR is illustrated in Figure 8.

The premise of the above approach is to train a high-quality topic model. However, since the transcripts of audio recordings are private and highly sensitive, it is impossible to train a comprehensive topic model by conventional approach, and we resort to iFTM to solve this problem. In our experiment, three parties are involved. Party P_1 has the transcript corresponding to 100-hour audio recording, P_2 and P_3 have the transcripts of 50-hour audio recording, respectively. We train topic models for each party with the conventional topic modeling and train the iFTM model according to those discussed in Section 3.

As a testbed, a full-fledged ASR system is trained using the Kaldi toolkit.[3] We investigate whether introducing topic information into the language model component of the ASR system can improve its performance. The topic information is utilized in the same way as the Re-Decoding mechanism described in Reference [44]. The performances of the ASR system with different language model components are evaluated by the standard metric Word Error Rate (WER) [24]. The lower the WER, the better the performance of the ASR system. A dataset of 10-hour audio recordings is used for testing.

The experimental results are shown in Table 2. We observe that both LDA and SentenceLDA are effective in reducing WER, but the models trained on larger data are of higher quality. Even with the perturbation caused by privacy protection, iFTM still achieves the best performance in terms of reducing WER, since harnessing more data significantly increases the quality of the topic model. Besides, SentenceLDA consistently outperforms conventional LDA. This evaluation verifies our assumption that iFTM can solve the problems plaguing real-life applications and improve their performance to a level that can not be achieved before.

4.6 iFTM in Document Classification

Document classification is a critical task in natural language processing. The topic distribution can be considered as a semantic representation of the document. In this experiment, the 100-dimensional topic distribution of each document is utilized as extra features for the downstream classifier support vector machine (SVM). Each party has 5,000 documents for training and we

[3] http://kaldi-asr.org/.

Table 2. Introducing LDA and SentenceLDA into ASR

Models	WER
ASR without Topic Model	33.18%
ASR with LDA trained on P_1	31.40%
ASR with LDA trained on P_2	32.32%
ASR with LDA trained on P_3	33.04%
ASR with FTM-LDA	30.06%
ASR with SentenceLDA trained on P_1	30.98%
ASR with SentenceLDA trained on P_2	32.07%
ASR with SentenceLDA trained on P_3	32.99%
ASR with FTM-SentenceLDA	30.01%

Table 3. Introducing LDA and SentenceLDA into the Document Classifier

Models	Precision
Classification without Topic Model	75.67%
Classification with LDA trained on P_1	78.45%
Classification with LDA trained on P_2	79.34%
Classification with LDA trained on P_3	76.87%
Classification with FTM-LDA	80.56%
Classification with SentenceLDA trained on P_1	79.65%
Classification with SentenceLDA trained on P_2	79.12%
Classification with SentenceLDA trained on P_3	77.34%
Classification with FTM-SentenceLDA	81.03%

utilizes another 2,000 documents for testing. Each document is graded by human annotators with four categories. We compare different settings in terms of precision.

The experimental result is shown in Table 3. We find that the topic feature is effective for article quality evaluation and effective in boosting the downstream classifier's performance. Similar to the experimental results in ASR, more data usually results in better topic models and hence better classification performance. We further observe that SentenceLDA typically performs better than LDA, showing that the document's latent structure is critical for effective topic representation and demonstrates better classification performance. This application unequivocally verifies the value of iFTM in industrial scenarios.

5 CONCLUSION

In this article, we propose a novel framework named Federated Industrial Topic Modeling (iFTM) to solve two critical problems faced by industrial topic modeling: data scarcity and data privacy. By seamlessly combining techniques such as differential privacy, private MCMC sampling, and meta-learning, iFTM significantly alleviates the problem of data scarcity while providing a principled approach for protecting data privacy. With the federated architecture in iFTM, a master and a series of parties work collectively to train high-quality topic models with low communication cost. Our quantitative experiments show that iFTM is significantly promising, as high-quality topic models can be trained in federated settings. Empirical evaluation of iFTM on automatic speech recognition and document classification shows that it truly solves some real-life problems that have not been

successfully handled before. Future work involves implementing more topic models based upon iFTM.

ACKNOWLEDGMENT

We are grateful to anonymous reviewers for their constructive comments.

REFERENCES

[1] Corey Arnold and William Speier. 2012. A topic model of clinical reports. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 1031–1032.

[2] Georgios Balikas, Massih-Reza Amini, and Marianne Clausel. 2016. On a topic model for sentences. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 921–924.

[3] Johes Bater, Xi He, William Ehrich, Ashwin Machanavajjhala, and Jennie Rogers. 2018. Shrinkwrap: Differentially-private query processing in private data federations. *arXiv preprint arXiv:1810.01816* (2018).

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *J. Mach. Learn. Res.* 3, Jan. (2003), 993–1022.

[5] Peter Carey. 2018. *Data Protection: A Practical Guide to UK and EU Law*. Oxford University Press, Inc.

[6] Mark J. Carman, Fabio Crestani, Morgan Harvey, and Mark Baillie. 2010. Towards query log based personalization using topic models. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. ACM, 1849–1852.

[7] Kuan-Yu Chen, Hsuan-Sheng Chiu, and Berlin Chen. 2010. Latent topic modeling of word vicinity information for speech recognition. In *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP'10)*. IEEE, 5394–5397.

[8] Kewei Cheng, Tao Fan, Yilun Jin, Yang Liu, Tianjian Chen, and Qiang Yang. 2019. SecureBoost: A lossless federated learning framework. *CoRR* abs/1901.08755 (2019).

[9] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation*. 1–19.

[10] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Found. Trends® Theoret. Comput. Sci.* 9, 3–4 (2014), 211–407.

[11] James Foulds, Joseph Geumlek, Max Welling, and Kamalika Chaudhuri. 2016. On the theory and practice of privacy-preserving Bayesian data analysis. *arXiv preprint arXiv:1603.07294* (2016).

[12] Zvi Galil and Giuseppe F. Italiano. 1991. Data structures and algorithms for disjoint set union problems. *ACM Comput. Surv.* 23, 3 (1991), 319–344.

[13] Robin C. Geyer, Tassilo Klein, and Moin Nabi. 2017. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557* (2017).

[14] Walter R. Gilks, Sylvia Richardson, and David Spiegelhalter. 1995. *Markov Chain Monte Carlo in Practice*. Chapman and Hall/CRC.

[15] Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proc. Na. Acad. Sci.* 101, suppl 1 (2004), 5228–5235.

[16] Jihun Hamm, Yingjun Cao, and Mikhail Belkin. 2016. Learning privately from multiparty data. In *Proceedings of the International Conference on Machine Learning*. 555–563.

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

[18] Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith, and Brian Thorne. 2017. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677* (2017).

[19] Morgan Harvey, Fabio Crestani, and Mark J. Carman. 2013. Building user profiles from topic models for personalised search. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. ACM, 2309–2314.

[20] Di Jiang, Kenneth Wai-Ting Leung, Wilfred Ng, and Hao Li. 2013. Beyond click graph: Topic modeling for search engine query log analysis. In *Proceedings of the International Conference on Database Systems for Advanced Applications*. Springer, 209–223.

[21] Di Jiang, Yuanfeng Song, Yongxin Tong, Xueyang Wu, Weiwei Zhao, Qian Xu, and Qiang Yang. 2019. Federated topic modeling. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. ACM, 1071–1080.

[22] Yohan Jo and Alice H. Oh. 2011. Aspect and sentiment unification model for online review analysis. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*. ACM, 815–824.

[23] Amir Karami, Aryya Gangopadhyay, Bin Zhou, and Hadi Karrazi. 2015. FLATM: A fuzzy logic approach topic model for medical documents. In *Proceedings of the Conference of the North American Fuzzy Information Processing Society (NAFIPS'15) held jointly with the 5th World Conference on Soft Computing (WConSC'15)*. IEEE, 1–6.

[24] Dietrich Klakow and Jochen Peters. 2002. Testing the correlation of word error rate and perplexity. *Speech Commun.* 38, 1 (2002), 19–28.

[25] David Leroy, Alice Coucke, Thibaut Lavril, Thibault Gisselbrecht, and Joseph Dureau. 2018. Federated learning for keyword spotting. *arXiv preprint arXiv:1810.05512* (2018).

[26] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. 2017. Meta-SGD: Learning to learn quickly for few shot learning. *arXiv preprint arXiv:1707.09835* (2017).

[27] Yang Liu, Tianjian Chen, and Qiang Yang. 2018. Secure federated transfer learning. *CoRR* abs/1812.03337 (2018).

[28] Jon D. Mcauliffe and David M. Blei. 2008. Supervised topic models. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*. 121–128.

[29] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. 2016. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629* (2016).

[30] David Newman, Arthur Asuncion, Padhraic Smyth, and Max Welling. 2009. Distributed algorithms for topic models. *J. Mach. Learn.* 10, Aug. (2009), 1801–1828.

[31] David Newman, Sarvnaz Karimi, and Lawrence Cavedon. 2009. Using topic models to interpret MEDLINE's medical subject headings. In *Proceedings of the Australasian Joint Conference on Artificial Intelligence*. Springer, 270–279.

[32] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22, 10 (2010), 1345–1359.

[33] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. 2016. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755* (2016).

[34] Mijung Park, James Foulds, Kamalika Chaudhuri, and Max Welling. 2016. Private topic modeling. *arXiv preprint arXiv:1609.04120* (2016).

[35] Ronald L. Rivest, Len Adleman, Michael L. Dertouzos, et al. 1978. On data banks and privacy homomorphisms. *Found. Sec. Comput.* 4, 11 (1978), 169–180.

[36] Thomas Rusch, Paul Hofmarcher, Reinhold Hatzinger, Kurt Hornik, et al. 2013. Model trees with topic model preprocessing: An approach for data journalism illustrated with the Wikileaks Afghanistan war logs. *Ann. Appl. Statist.* 7, 2 (2013), 613–639.

[37] Jacob M. Victor. 2013. The EU general data protection regulation: Toward a property regime for protecting data privacy. *Yale LJ* 123 (2013), 513.

[38] Paul Voigt and Axel Von dem Bussche. 2017. *The EU General Data Protection Regulation (GDPR). A Practical Guide*, 1st (ed.). Springer International Publishing, Cham.

[39] Jan Vosecky, Di Jiang, Kenneth Wai-Ting Leung, Kai Xing, and Wilfred Ng. 2014. Integrating social and auxiliary semantics for multifaceted topic modeling in Twitter. *ACM Trans. Internet Technol.* 14, 4 (2014), 27.

[40] W. Gregory Voss. 2016. European Union data privacy law reform: General data protection regulation, privacy shield, and the right to delisting. *Bus. Law.* 72, 1 (2016), 221–233.

[41] Xuerui Wang and Andrew McCallum. 2006. Topics over time: A non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 424–433.

[42] Yang Wang, Quanquan Gu, and Donald Brown. 2018. Differentially private hypothesis transfer learning. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 811–826.

[43] Yu-Xiang Wang, Stephen E. Fienberg, and Alexander J. Smola. 2015. Privacy for free: Posterior sampling and stochastic gradient Monte Carlo. In *Proceedings of the International Conference on Machine Learning (ICML'15)*, Vol. 15. 2493–2502.

[44] Jonathan Wintrode and Sanjeev Khudanpur. 2014. Combining local and broad topic context to improve term detection. In *Proceedings of the IEEE Spoken Language Technology Workshop (SLT'14)*. IEEE, 442–447.

[45] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.* 10, 2 (2019), 12.

[46] Yuan Yang, Jianfei Chen, and Jun Zhu. 2016. Distributing the stochastic gradient sampler for large-scale LDA. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1975–1984.

[47] Andrew Chi-Chih Yao. 1982. Protocols for secure computations. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS'82)*, Vol. 82. 160–164.

[48] Jinhui Yuan, Fei Gao, Qirong Ho, Wei Dai, Jinliang Wei, Xun Zheng, Eric Po Xing, Tie-Yan Liu, and Wei-Ying Ma. 2015. LightLDA: Big topic models on modest computer clusters. In *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1351–1361.

[49] Ke Zhai, Jordan Boyd-Graber, Nima Asadi, and Mohamad L. Alkhouja. 2012. Mr. LDA: A flexible large scale topic modeling package using variational inference in MapReduce. In *Proceedings of the 21st International Conference on World Wide Web*. ACM, 879–888.

Received November 2019; revised May 2020; accepted July 2020