Federated Topic Modeling

Di Jiang
AI Group, WeBank Co., Ltd
Shenzhen, China
dijiang@webank.com

Yuanfeng Song[*]
AI Group, WeBank Co., Ltd
Shenzhen, China
yfsong@webank.com

Yongxin Tong
BDBC, SKLSDE Lab and IRI
Beihang University
Beijing, China
yxtong@buaa.edu.cn

Xueyang Wu
The Hong Kong University of Science and Technology
Hong Kong SAR, China
xwuba@cse.ust.hk

Weiwei Zhao
AI Group, WeBank Co., Ltd
Shenzhen, China
davezhao@webank.com

Qian Xu
AI Group, WeBank Co., Ltd
Shenzhen, China
qianxu@webank.com

Qiang Yang[*]
AI Group, WeBank Co., Ltd
Shenzhen, China
qiangyang@webank.com

ABSTRACT

Topic modeling has been widely applied in a variety of industrial applications. Training a high-quality model usually requires massive amount of in-domain data, in order 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 paper, we propose a novel framework named *Federated Topic Modeling* (FTM), in which multiple parties collaboratively train a high-quality topic model by simultaneously alleviating data scarcity and maintaining immune to privacy adversaries. FTM is inspired by federated learning and consists of novel techniques such as private Metropolis Hastings, topic-wise normalization and heterogeneous model integration. We conduct a series of quantitative evaluations to verify the effectiveness of FTM and deploy FTM in an Automatic Speech Recognition (ASR) system to demonstrate its utility in real-life applications. Experimental results verify FTM's superiority over conventional topic modeling.

CCS CONCEPTS

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

KEYWORDS

Text Semantics; Topic Model; Bayesian Networks

ACM Reference Format:
Di Jiang, Yuanfeng Song, Yongxin Tong, Xueyang Wu, Weiwei Zhao, Qian Xu, and Qiang Yang. 2019. Federated Topic Modeling. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3357384.3357909

1 INTRODUCTION

Topic modeling has been successfully used in many industrial applications, from military analysis [41], to web search log mining [8, 23, 24], to medical informatics [1, 26, 36]. As training a high-quality topic model for a specific application typically requires comprehensive in-domain data to provide sufficient co-occurrence information, relying on data collected from a single party will be faced with 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) [7, 42, 43] may preclude uploading them to data centers and being utilized in a centralized approach. These two critical problems poses new challenges to conventional topic modeling, which we refer to as the state-of-the-art distributed architectures [35, 54, 55] for training topic models on computer clusters within a data center.

To solve the above problems, a new topic modeling paradigm simultaneously alleviating data scarcity and protecting data privacy is urgently needed in industry. However, the huge discrepancy between the scenario of conventional topic modeling and that studied in this paper 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 is able to access computing nodes or monitor network communication can easily get a glimpse of the data of each party. Such practice is increasingly forbidden by new data regulations. Second, how to reduce the communication cost

[*] Also with The Hong Kong University of Science and Technology.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM '19, November 3–7, 2019, Beijing, China

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3357909

RIGHTSLINK

between computing nodes. Conventional topic modeling such as those deployed upon MapReduce [55] or ParameterServer [54] usually has 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 frequently communicate with each as before. 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 trains the same topic model. However, this requirement can hardly be met in the present problem where each party usually stores 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 [20, 33], we propose a new framework named *Federated Topic Modeling* (FTM) that solved the aforementioned problems in a principled approach. As shown in Figure 1, FTM 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) [16] 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 data sets. 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, in order to achieve significantly lower communication cost and handle data that are not i.i.d.. We systematically evaluate FTM in terms of quantitative metrics such as likelihood and communication cost as well as real-life applications such as automatic speech recognition (ASR) [53]. The contributions of this paper are summarized as follows:

- To the best of our knowledge, FTM is the first framework that is specifically designed for large-scale distributed topic modeling with a guarantee of privacy protection.
- FTM pioneers in new topic modeling paradigms such as allowing heterogeneous topic models to be trained on data that are not i.i.d. across different parties.
- FTM 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 such as ASR demonstrate the necessity and effectiveness of FTM.

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

2 RELATED WORK

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

2.1 Topic Modeling

Topic modeling has been intensively studied and widely used for the last decade. Latent Dirichlet Allocation (LDA) [4] plays the most important role among all kinds of topic models. In the meantime, various extensions of LDA have been proposed in order to satisfy specific needs in different domains and applications. For example, Topic-over-time (TOT) [45] captures the changes of latent topic over times by jointly modeling the timestamps with word occurrence patterns. Supervised LDA [32] makes use of the labelled documents to guide the inference of the topics. Sentence LDA [2] incorporates the structure of the text during the model training and inference processes by restricting all the words in the same sentence sharing the same topic.

The recent advancement of this field focuses on training topic models such as LDA on clusters by distributed computing [35, 51, 54, 55], and LightLDA [54] maintains the state-of-the-art performance in terms of model quality. We consider these works as conventional topic modeling and the issue of privacy is typically neglected in them. A recent work [39] 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 a distributed computing. Although privacy and distributed computing are two critical factors determining the applicability of topic models in industry, they have been studied independently in existing work and how to integrate them in a unified framework is still an open problem.

2.2 Federated Learning

Many machine learning tasks have been achieving significant progress recently years with the development of machine learning techniques, such as deep neural networks, and the increasing amount of labeled data. However, high-quality labeled data are precious, which take huge human resources and time to collect and to annotate. Moreover, in many practical applications, the accessible data are very limited, making the learning difficult. To address the small data problem, some researchers focus on developing machine learning algorithms that can leverage the knowledge from a domain with rich data to help the learning in the target domain with limited data or even without labeled data, i.e., transfer learning [37]. The other proposed solution to address the small data problem is trying to join the fragmented small data from multi-party, i.e. collaborative machine learning. However, both of these solutions do not consider the data privacy and security, while personal information privacy and security have become a insurmountable concern before the wide applications of these techniques. Many countries enforce or plan to publish laws and regulations to protect the personal information privacy and security. For example, GDPR aims to protect the security and privacy of user, giving the right to user for ripping their personal information from companies.

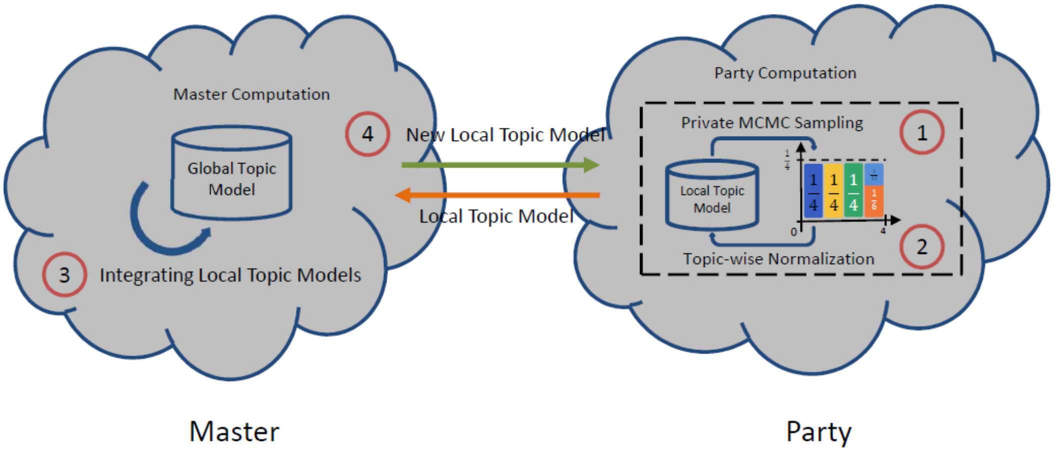


Figure 1: The Federated Topic Modeling Framework: Party Computation (① Private MCMC Sampling and ② Topic-wise Normalization) and Master Computation (③ Integrating Local Topic Models and ④ Composing New Local Topic Models)

To deal the the contradiction between the need of massive data for learning and the regulations to protect the data privacy and security, researchers propose privacy-preserving collaborative learning paradigm, federated learning [33, 50]. Recently, federated learning is combined with various machine learning algorithms such as neural networks [33], SVMs [6], Logistic regression [18]. According to how the data across parties are utilized in federated learning, algorithms can be categorized into three types [50]:

(1) **Horizontal federated learning**: data on different parties have the same feature space, and the samples on different parties are different. Samples are learned as virtually joint together across parties with protection of privacy and security.
(2) **Vertical federated learning**: data on different parties share the same sample ID space, and their feature spaces are different. The data are virtually joint in feature dimension through vertical federated learning, i.e., the feature of samples in each party are extended with features from other parties.
(3) **Federated transfer learning**: the sample ID spaces and features spaces of data in different parties are different. Federated transfer learning leverages the knowledge from a labeled party to help the party without labeled data.

The typical scenario of horizontal federated learning is the learning applications on mobile devices, such as language modeling [21], and keyword spotting [28]. Data in mobile devices are highly personal and thus sensitive, while in federated learning data are only operated locally, without transferring outside the party. To some extents, the privacy and security can be guaranteed by accessing the data locally. However, transferring parameters or gradients may also leak sensitive information [34, 44, 48]. To achieve higher privacy and security level, [5] improve the the horizontal federated learning algorithm by introducing the secure aggregation protocol. In addition, due to the probabilistic property of differential privacy [11], differentially privacy is widely used in federated learning for protecting the transaction of models or data [3, 15, 19, 22, 38, 46]. In vertical federated learning [10] and federated transfer learning [31], homomorphic encryption (HE) [40] are used, as HE does not inject noise to the transferred data. However, HE-based methods bring extra encryption computation costs and larger communication cost due to the ciphertext. Secure multi-party computation [52] methods are also used for collaborative learning, but the number of participants in this protocol are limited. In our work, we use horizontal federated learning techniques with differentially private protection on data privacy.

2.3 Differential Privacy

Differential privacy provides a quantitative definition on data and model that is compatible with many machine learning algorithms, which relieve the paradox of privacy-preserving learning that grapes helpful information from a population but learns nothing about any individual. The formal definition of differential privacy is as following: A randomized algorithm $\mathcal{M}(\mathbf{X})$ is considered as (ε, δ)-differentially private if

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

for all $\mathcal{S} \subset \text{Range}(\mathcal{M})$, and for all adjacent datasets $\mathbf{X}, \mathbf{X}'$. Here differential privacy leaves the definition of adjacent open to different settings. It means that one can infer privacy information from the output of that randomized algorithm $\mathcal{M}(\mathbf{X})$ with a negligible probability, where $\mathbf{X}$ can be regarded as any subset from a population. One can sniff individual information through enough many queries to elaborately constructed populations. Therefore, the difference of output of an algorithm with different $\mathbf{X}$ is critical to the individual privacy protection. In differential privacy, it is measured as *sensitivity*. Researchers develop many tools to guarantee a certain level of privacy under a given sensitivity, such as the Gaussian mechanism, and Laplace Mechanism. Laplace mechanism [12] is a common method for obtaining ϵ-differential privacy. The main idea is to add Laplace noise to the revealed data with the amount of noise controlled by ϵ. Specifically, the L_1 sensitivity Δh for function h is defined as:

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

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

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

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

Composition theorems [12] deal with the case when we combine several differentially private blocks together for more sophisticated algorithms and differential privacy satisfies sequential composition and parallel composition. *Sequential composition*: suppose $\mathcal{M}_j(\mathbf{X}), j = 1, \cdots, l$, are (ε_j)-differentially private, then the combination of these algorithms $\mathbf{X} \rightarrow (\mathcal{M}_1(\mathbf{X}), \cdots, \mathcal{M}_l(\mathbf{X}))$ 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 $\mathcal{M}_j(\mathbf{X}_j), j = 1, \cdots, l$, are (ε_j)-differentially private, then the combination of these algorithms $\mathbf{X} \rightarrow (\mathcal{M}_1(\mathbf{X_1}), \cdots, \mathcal{M}_l(\mathbf{X_l}))$ satisfies $(\max_j \varepsilon_j)$-differentially private. Besides composition of differentially private mechanisms, DP is also immune to any deterministic post-processing [13], which means we can take further operations over the privatized data, achieving the identical differential privacy budget.

3 FEDERATED TOPIC MODELING FRAMEWORK

The recent multitude of successful applications of topic modeling have almost relied on Latent Dirichlet Allocation (LDA) and variants of MCMC as the parameter inference algorithm [29, 54]. Thus, it is natural that we describe FTM by starting from LDA and MCMC. The techniques discussed in this section pave the way for designing similar algorithms for other topic models. In order to facilitate the discussion thereafter, we list the notations that will be used in this paper 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 whole workflow of FTM in Section 3.3.

Table 1: Notations for FTM

Notation	Meaning
D	the size of documents
K	the number of topics
V	the size of vocabulary
Φ	the word distributions of topics
ϕ_k	the word distribution of topic k
Θ	topic distributions of documents
θ_d	topic distributions of document d
$\mathbf{w}$	the words vector of a document
w_{di}	ith word in document d
$\mathbf{z}$	the topic assignment vector of a document
z_{di}	the topic assignment of ith in document d
$\mathbf{z}_{-di}$	the topic assignment vector of document d except the ith word
k	a topic index
α	Dirichlet prior vector for θ
β	Dirichlet prior vector for φ
C^{DK}_{dk}	the number of words assigned to topic k in document d
C^{KW}_{kw}	the number of word w assigned to topic k
$C^{KW}_{k\cdot}$	an 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

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

3.1.1 Private Gibbs Sampling. Gibbs sampling is widely used for parameter inference of topic models [17, 25, 45]. 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 [17]. 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 Eq. 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:

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

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:

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

where $n_{diw'} = \mathbb{I}[w' = w_{di}]$. Since the sampling algorithm interacts with the corpus only by the sufficient statistics for conditional probability $\exp(\sum_{w'} n_{diw'} \log \phi_{z_{di}w'})$, 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 $1/\varepsilon$. 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 [47] and [13], 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 Eq.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 in which data sets are voluminous.

ALGORITHM 1: Private Metropolis Hastings

```
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 Eq.9 and threshold by $\tau$
        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 Eq.11
        build a word-topic alias table according to Eq.14
        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 Eq.13;
                propose a topic $z_b$ with the word-topic alias table;
                update $z_i$ according to $z_b$ and Eq.16;
            end
        end
        sample $\Phi$ according to Eq.7
    end
compose local model $\mathcal{M}_p$ according to $C^{KW}_{k\cdot}$
```

3.1.2 Private Metropolis Hastings. To improve the efficiency of MCMC sampling and make it applicable on massive dataset, we resort to private Metropolis-Hastings (MH), which 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:

$$\Omega^z_d = \frac{(C^{DK}_{dz} + \alpha)}{\sum_{k'}(C^{DK}_{dk'} + \alpha)} \tag{11}$$

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:

$$min\{1, \frac{P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^z_d}{P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^{z'}_d}\} \tag{12}$$

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

$$min\{1, \frac{(\hat{C}^{DK}_{dz'} + \alpha)\hat{P}(w|z', \Phi)(C^{DK}_{dz} + \alpha)}{(\hat{C}^{DK}_{dz} + \alpha)\hat{P}(w|z, \Phi)(C^{DK}_{dz'} + \alpha)}\} \tag{13}$$

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:

$$\Omega^z_w = \frac{C^{KW}_{zw} + \beta}{\sum_{w'}(C^{KW}_{zw'} + \beta)} \tag{14}$$

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:

$$min\{1, \frac{P(z'|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^z_w}{P(z|\mathbf{z}_{-di}, \mathbf{w}, \alpha, \beta)\Omega^{z'}_w}\} \tag{15}$$

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

$$min\{1, \frac{(\hat{C}^{DK}_{dz'} + \alpha)\hat{P}(w|z', \Phi)(C^{KW}_{zw} + \beta)(\sum_{w'}(C^{KW}_{z'w'} + \beta))}{(\hat{C}^{DK}_{dz} + \alpha)\hat{P}(w|z, \Phi)(C^{KW}_{z'w} + \beta)(\sum_{w'}(C^{KW}_{zw'} + \beta))}\} \tag{16}$$

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

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

(1) Improving the sampling efficiency of the proposals of Eq.11 and Eq.14. 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 [54]. The key idea of the alias method is the construction of the alias table, which is illustrated by an example in Figure 2. 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". At the meantime, it guarantees each entry has at most two kinds of entry index. With alias method, the

RIGHTSLINK

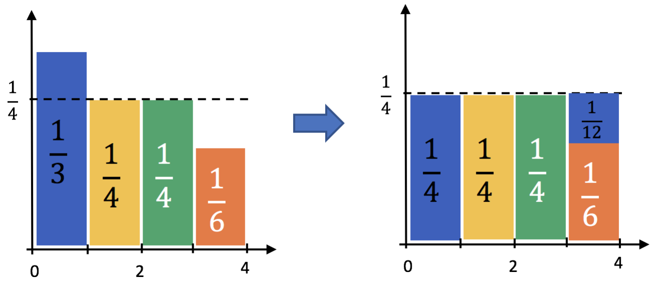
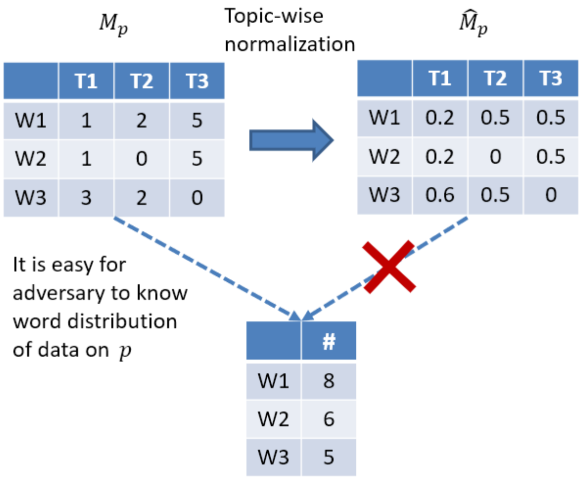


Figure 2: A Toy Example of Building Alias Table


M_p
Topic-wise normalization
$\hat{M}_p$
It is easy for adversary to know word distribution of data on p

	T1	T2	T3
W1	1	2	5
W2	1	0	5
W3	3	2	0

	T1	T2	T3
W1	0.2	0.5	0.5
W2	0.2	0	0.5
W3	0.6	0.5	0

	#
W1	8
W2	6
W3	5

Figure 3: A Toy Example of Topic-wise Normalization. Hyperparameters are neglected in this example for simplicity.

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 high mixing rate.

(2) Reducing the computational cost of calculating the acceptance probabilities of Eq.13 and Eq.16. The bottle neck of calculating Eq.13 and Eq.16 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 [12], applying a deterministic post-processing to a ε-differentially private mechanism is still ε-differentially private. Therefore, the above operation does not affect the privacy guarantee.

3.1.3 Topic-wise Normalization. The local model (i.e., the result of Algorithm 1) 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 3, 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

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 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. In order 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:

$$\begin{aligned}\rho(z_i, z_j) &= \frac{\sum_{l=1}^{m} \min(p_{w_l}^{z_i}, p_{w_l}^{z_j})}{\sum_{l=1}^{m} \max(p_{w_l}^{z_i}, p_{w_l}^{z_j}) + \sum_{m+1}^{T} p_{w_l}^{z_i} + \sum_{l=m+1}^{T} p_{w_l}^{z_j}} \\ &= \frac{\sum_{l=1}^{m} \min(p_{w_l}^{z_i}, p_{w_l}^{z_j})}{\sum_{l=1}^{T} p_{w_l}^{z_i} + \sum_{l=1}^{T} p_{w_l}^{z_j} - \sum_{l=1}^{m} \min(p_{w_l}^{z_i}, p_{w_l}^{z_j})}\end{aligned} \quad (17)$$

where $P_{z_i} = (p_{w_1}^{z_i}, p_{w_2}^{z_i}, \cdots, p_{w_m}^{z_i}, p_{w_{m+1}}^{z_i}, \cdots, p_{w_L}^{z_i})$ and

$$P_{z_j} = (p_{w_1}^{z_j}, p_{w_2}^{z_j}, \cdots, p_{w_m}^{z_j}, p_{w_{m+1}}^{z_j}, \cdots, p_{w_L}^{z_j})$$

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.

Based on above similarity metric, we detail the mechanism of integrating local models in Algorithm 2. The algorithm first concatenates two topic models (Line 2). Then it finds the redundant topic sets based on the Union-Find [14] algorithm (Line 2 ∼ 11). For example, if (z_1, z_2) and (z_2, z_3) are considered as redundant based on Eq. 17, $\{z_1, z_2, z_3\}$ 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 (Line 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 weight 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 2: Integrating Local Topic Models

```
input  : global topic model $\mathcal{M}$, local topic model $\mathcal{M}_p$.
output : updated global topic Model $\mathcal{M}^*$.
begin
    concatenate $\mathcal{M}$ and $\mathcal{M}_p$ into $\mathcal{M}^B$;
    redundant topics $\mathcal{R} = \{\}$
    for each topic $z_i$ in $\mathcal{M}^B$ do
        for each topic $z_j$ $(j > i)$ in $\mathcal{M}^B$ do
            calculate $\rho(z_i, z_j)$ with Eq. (17);
            if $\rho(z_i, z_j) \geq \xi$ then
                Add $(z_i, z_j)$ into $\mathcal{R}$
            end
        end
    end
    for each set s in Union-Find($\mathcal{R}$) do
        for each topic $z_{si}$ $(i > 1)$ in s do
            add $w_{si}\vec{z_{si}}$ to $\vec{z_{s1}}$, remove $\vec{z_{si}}$ from $\mathcal{M}^B$;
        end
        normalize distribution $\vec{z_{si}}$;
    end
    $\mathcal{M}^* = \mathcal{M}^B$
end
return $\mathcal{M}^*$;
```

3.2.2 *Composing New Local Topic Models.*

Since the global topic model M^* is large and comprehensive, some topics in M^* is irrelevant to the data of certain parties. Hence, it is unnecessary to push all the information in M^* to each individual party. In order to effectively reduce the communication cost, we compose a new local model that is compact enough to be pushed to the corresponding party. In order to take full advantages of the global model to facilitate local training, we employ *meta-learning* [1] [30] 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 3.

3.3 FTM Workflow

The workflow of FTM is presented in Algorithm 4. For each global iteration, during the party computation stage, each party trains a local topic model according to Algorithm 1 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 2, composes and pushes new local models for each party according to Algorithm 3. The whole process repeats for a predefined number of global iterations.

ALGORITHM 3: Composing New Local Topic Models

```
input  : global topic model $\mathcal{M}^*$, local topic model $\hat{M}^p$
output : new local topic model $M_p^{'}$.
begin
    for each topic $z_p$ in $\hat{M}_p$ do
        $z = \arg\max_{z \in M^*} \rho(z, z_p)$;
        if $\rho(z, z_p) \geq \xi$ then
            replace $z_p$ with $z$ into $M_p^{'}$;
        end
        remove $z$ from $M^*$;
    end
end
return $M_p^{'}$;
```

We will show later that few global iterations are sufficient to obtain a good global topic model M^*. The low synchronization frequency improves FTM's robustness to low bandwidth and network failures, which are more common in wide area networks than in data centers.

ALGORITHM 4: FTM Workflow

```
for each global iteration do
    for each client p do
        train local topic model according to Algorithm 1
        push local topic model $\hat{M}_p$ to master
    end
    integrate local topic models and obtain the global topic
      model $M^*$ according to Algorithm 2
    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 EXPERIMENTS

In this section, we evaluate the performance of FTM 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 FTM in alleviating data scarcity. In Section 4.3, we demonstrate the utility of FTM in terms of different parameter settings. In Section 4.4, we gauge the communication cost of FTM. Finally, we show the necessity and the promising performance of FTM through a real-life application of ASR in Section 4.5.

4.1 Experimental Setup

We assume that there are three parities 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 stores 29723, 59445 and 89169 documents respectively. A corpus containing another 29700 documents is used as the testing data. These baselines are trained through the LightLDA[2] toolkit and the

[1] Meta-learning, also named learning to learn, is previously utilized in 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[30].

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

RIGHTSLINK

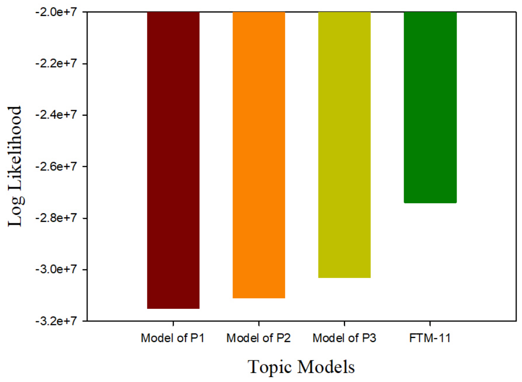


Figure 4: Performance of Data Scarcity Alleviation

number of topics has been tuned for each party to make them strong baselines.

4.2 Data Scarcity Alleviation

One major motivation of FTM is to alleviate the data scarcity problem faced by each individual party. Hence, one important question is whether the model trained by FTM is better than those trained by a single party relying on its own data. Figure 4 shows the comparison of FTM and different parties’ topic models in terms of the log likelihood of testing data. We observe that harnessing more data usually results in better topic models. By collectively utilizing data from all parties, FTM achieves the highest likelihood. For example, FTM-11 (i.e., FTM with $\varepsilon = 11$ and $\tau = 0.2$) demonstrate the log likelihood of -2.74×10^7 while the best topic model trained by a single party is the one from P_3 and only achieves a log likelihood of -3.03×10^7. Similar results can be observed with many other settings of ε and τ. We skip them due to space limitation. The result indicates that FTM is effective to alleviate data scarcity and generate high-quality topic models that cannot be obtained based upon a single party’s data.

4.3 Privacy Protection

The performance of FTM with different ε reflects the utility of FTM after different levels of privacy protection. In Figure 5, we show the performance of FTM with regard to 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, FTM usually achieves higher likelihood on the testing data. For example, FTM with $\varepsilon = 8$ and $\tau = 0.2$ achieves a log likelihood of -4.59×10^7. When ε increases to 11, FTM 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. Empirically, $\tau = 0.2$ demonstrates fairly good performance with moderate noise and we utilize it by default.

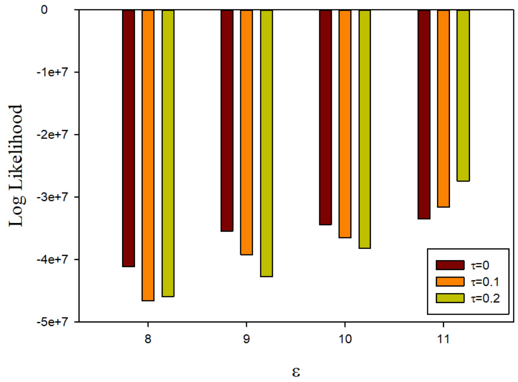


Figure 5: Performance of Privacy Protection

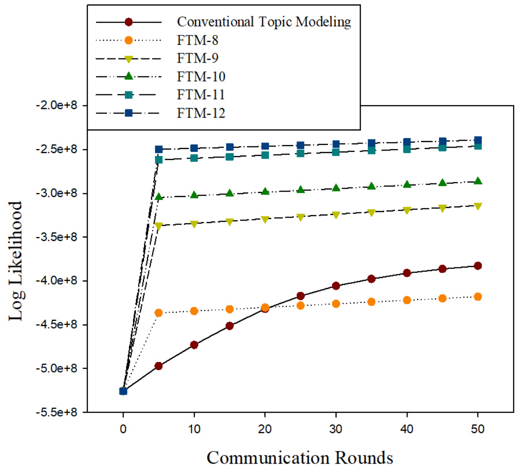


Figure 6: Likelihood versus Communication

4.4 Communication Cost

Figure 6 presents the communication costs of conventional topic modeling and FTM with different ε. The baselines conventional topic modeling is a LightLDA model trained on a dataset consisting of the training data from P_1, P_2 and P_3. We observe that FTM converges quickly within several rounds of communication while conventional topic modeling demonstrates much slower speed of convergence. FTM with higher ε demonstrates superior performance in terms of model quality and communication efficiency. When ε is lower than 9, the final model of FTM 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 FTM in less than 5 rounds. These results verify the superiority of FTM in low-bandwidth environment. Another interesting observation is that introducing moderate noise is beneficial for improve the quality of the model under training. When ε is set to a value larger than 8, the models trained by FTM achieves higher likelihood than that trained by conventional topic modeling on original data.

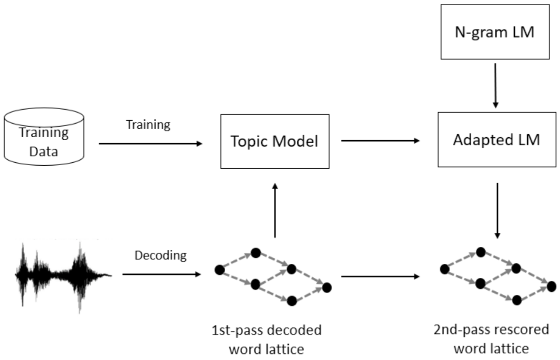


Figure 7: The Pipeline of Applying Topic Model in ASR

4.5 FTM in Automatic Speech Recognition

Topic models are known for effectively improving the performance of Automatic Speech Recognition (ASR) systems through providing richer contextual information for the language model (LM) component in ASR [9, 49]. Specifically, topic models are utilized to calculate the probability of seeing a word given the context:

$$P_{TM}(w|C) = \sum_{z} P(w|z)P(z|C) \tag{18}$$

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 semantic information. In practice, we conduct a linear interpolation between the traditional backoff n-gram language model and that produced by Eq. 18 to generate the adapted language model $P(w|C)$:

$$P(w|C) = \lambda P_{TM}(w|C) + (1 - \lambda)P_{LM}(w|C) \tag{19}$$

where $P_{LM}(w|C)$ is the probability given by the traditional backoff n-gram language model and λ is a trade-off parameter. The pipeline of applying topic models in ASR is illustrated in Figure 7.

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 FTM 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 FTM model according to those discussed in Section 3.

As a testbed, an 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 [49]. The performances the ASR system with different language model components are evaluated by the standard metric Word Error Rate (WER) [27]. The lower the WER, the better the performance of the ASR system. A data set of 10-hour audio recordings is used for testing. The experimental results are shown in Table 2. We observe that all topic models are effective in reducing WER but the models trained on larger data are of higher quality. Even with the perturbation caused by privacy protection, FTM still achieves the best performance in term of reducing WER, since harnessing comprehensive data significantly increases the quality of topic model. This application-oriented evaluation verifies our assumption that FTM can solve problems that plague real-life applications and improve their performance to a level that can not be achieved before.

Table 2: Introducing Topic Models into ASR

Models	WER
ASR without Topic Model	33.183%
ASR with Topic Model trained on P_1	31.401%
ASR with Topic Model trained on P_2	32.324%
ASR with Topic Model trained on P_3	33.035%
ASR with FTM-11	30.063%

5 CONCLUSION

In this paper, we propose a novel framework named Federated Topic Modeling (FTM) to solve two critical problems faced by industrial topic modeling: data scarcity and data privacy. By seamlessly combining techniques such as differential privacy, MCMC sampling and meta learning, FTM significantly alleviates the problem of data scarcity while providing a principled approach for protecting data privacy. With the federated architecture in FTM, a master and a series of parties work collectively to train high-quality topic models with low communication cost. Our quantitative experiments show that FTM has significant promise, as high-quality topic models can be trained in federated setting. Empirical evaluation of FTM on automatic speech recognition show that that it truly solves some real-life problems that have not been successfully handled before. Future work involves implementing more topic models based upon FTM.

ACKNOWLEDGMENTS

We are grateful to anonymous reviewers for their constructive comments. This work is partially supported by the National Science Foundation of China (NSFC) under Grant No. 61822201, U1811463 and 61532004. Yongxin Tong is the corresponding author of this paper.

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. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

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

[5] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2016. Practical secure aggregation for federated learning on user-held data. *arXiv preprint arXiv:1611.04482* (2016).

[6] Theodora S Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. 2018. Federated learning of predictive models from federated Electronic Health Records. *International journal of medical informatics* 112 (2018), 59–67.

[7] Peter Carey. 2018. *Data protection: a practical guide to UK and EU law*. Oxford University Press, Inc.

[8] 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.

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

[10] 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). arXiv:1901.08755 http://arxiv.org/abs/1901.08755

[11] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. In *Theory and Applications of Models of Computation, 5th International Conference, TAMC 2008, Xi'an, China, April 25-29, 2008. Proceedings*. 1–19.

[12] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[13] 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).

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

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

[16] Walter R Gilks, Sylvia Richardson, and David Spiegelhalter. 1995. *Markov chain Monte Carlo in practice*. Chapman and Hall/CRC.

[17] Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences* 101, suppl 1 (2004), 5228–5235.

[18] Xiawei Guo, Quanming Yao, WeiWei Tu, Yuqiang Chen, Wenyuan Dai, and Qiang Yang. 2018. Privacy-preserving Transfer Learning for Knowledge Sharing. *arXiv preprint arXiv:1811.09491* (2018).

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

[20] 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).

[21] 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).

[22] 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).

[23] 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.

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

[25] Yohan Jo and Alice H Oh. 2011. Aspect and sentiment unification model for online review analysis. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 815–824.

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

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

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

[29] Aaron Q Li, Amr Ahmed, Sujith Ravi, and Alexander J Smola. 2014. Reducing the sampling complexity of topic models. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 891–900.

[30] 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).

[31] Yang Liu, Tianjian Chen, and Qiang Yang. 2018. Secure Federated Transfer Learning. *CoRR* abs/1812.03337 (2018). arXiv:1812.03337 http://arxiv.org/abs/1812.03337

[32] Jon D Mcauliffe and David M Blei. 2008. Supervised topic models. In *Advances in neural information processing systems*. 121–128.

[33] 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).

[34] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2018. Exploiting unintended feature leakage in collaborative learning. In *Exploiting Unintended Feature Leakage in Collaborative Learning*. IEEE, 0.

[35] David Newman, Arthur Asuncion, Padhraic Smyth, and Max Welling. 2009. Distributed algorithms for topic models. *Journal of Machine Learning Research* 10, Aug (2009), 1801–1828.

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

[37] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2010), 1345–1359.

[38] 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).

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

[40] Ronald L Rivest, Len Adleman, Michael L Dertouzos, et al. 1978. On data banks and privacy homomorphisms. *Foundations of secure computation* 4, 11 (1978), 169–180.

[41] 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. *The Annals of Applied Statistics* 7, 2 (2013), 613–639.

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

[43] W Gregory Voss. 2016. European union data privacy law reform: General data protection regulation, privacy shield, and the right to delisting. *Business Lawyer* 72, 1 (2016), 221–233.

[44] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K Leung, Christian Makaya, Ting He, and Kevin Chan. 2018. Adaptive federated learning in resource constrained edge computing systems. *learning* 8 (2018), 9.

[45] 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.

[46] Yang Wang, Quanquan Gu, and Donald Brown. 2018. Differentially Private Hypothesis Transfer Learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 811–826.

[47] Yu-Xiang Wang, Stephen E Fienberg, and Alexander J Smola. 2015. Privacy for Free: Posterior Sampling and Stochastic Gradient Monte Carlo.. In *ICML*, Vol. 15. 2493–2502.

[48] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. 2018. Beyond Inferring Class Representatives: User-Level Privacy Leakage From Federated Learning. *arXiv preprint arXiv:1812.00535* (2018).

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

[50] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated Machine Learning: Concept and Applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 10, 2 (2019), 12.

[51] 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.

[52] Andrew Chi-Chih Yao. 1982. Protocols for secure computations. In *FOCS*, Vol. 82. 160–164.

[53] Dong Yu and Li Deng. 2016. *AUTOMATIC SPEECH RECOGNITION*. Springer.

[54] 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.

[55] 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.

RIGHTSLINK