Gaussian Mixture Variational Autoencoder for Semi-Supervised Topic Modeling

CANGQI ZHOU1, HAO BAN2, JING ZHANG1, (Senior Member, IEEE), QIANMU LI3,4, 
AND YINGHUA ZHANG5

1School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
2School of Information Science and Engineering, Southeast University, Nanjing 210096, China
3School of Cyber Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
4Information Department, Nanjing University of Science and Technology, Nanjing 210094, China
5SenseDeal Intelligent Technology Company Ltd., Beijing 100084, China

Corresponding authors: Hao Ban (banhao@seu.edu.cn) and Yinghua Zhang (yhzh@sensedeal.ai)

This work was supported in part by the National Natural Science Foundation of China under Grant 61902186 and Grant 91846104, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20180463, in part by the 4th Project of the National Key Research and Development Program under Grant 2020YFB1804604 and Grant 2020YFB1804600, in part by the Fundamental Research Funds for the Central Universities under Grant 30920010008 and Grant 30918012204, in part by the 2020 Industrial Internet Innovation and Development Project from Ministry of Industry and Information Technology of China, in part by the 2018 Jiangsu Province Major Technical Research Project, and in part by the Cooperative Research Project Commissioned by Zhongtian Technology.

ABSTRACT Topic models are widely explored for summarizing a corpus of documents. Recent advances in Variational AutoEncoder (VAE) have enabled the development of black-box inference methods for topic modeling in order to alleviate the drawbacks of classical statistical inference. Most existing VAE based approaches assume a unimodal Gaussian distribution for the approximate posterior of latent variables, which limits the flexibility in encoding the latent space. In addition, the unsupervised architecture hinders the incorporation of extra label information, which is ubiquitous in many applications. In this paper, we propose a semi-supervised topic model under the VAE framework. We assume that a document is modeled as a mixture of classes, and a class is modeled as a mixture of latent topics. A multimodal Gaussian mixture model is adopted for latent space. The parameters of the components and the mixing weights are encoded separately. These weights, together with partially labeled data, also contribute to the training of a classifier. The objective is derived under the Gaussian mixture assumption and the semi-supervised VAE framework. Modules of the proposed framework are appropriately designated. Experiments performed on three benchmark datasets demonstrate the effectiveness of our method, comparing to several competitive baselines.

INDEX TERMS Topic model, variational autoencoder, semi-supervised learning, Gaussian mixture model, deep generative learning.

I. INTRODUCTION

Topic models [1], [2] provide us with methods to discover abstract word and phrase patterns that best summarize and characterize a corpus of documents. Applications of topic models ranges from organizing daily documents such as emails [3], news [4] and social media posts [5], to understanding professional documents such as scientific papers [6], medical records [7] and technical reports [8]. Latent Dirichlet Allocation (LDA) [9] and its variants [10]–[12] are among the most popular models for extracting topic structure from documents. These probabilistic generative models adopt Markov China Monte Carlo (MCMC) and Variational Inference (VI) to approximate the true Bayesian inference that is often intractable due to high dimensionality of latent variables [13]. The standard inference approaches suffer a lot from high computation cost and tedious mathematical derivations, which hinders practitioners to explore large datasets [14].

The limitations motivate neural network-based black-box inference methods [15]. Variational AutoEncoder (VAE) [16] provides us a framework to alleviate the above-mentioned limitations, by training an inference network that maps the representations of documents to an approximate posterior distribution directly. Recently, several studies have
endeavoured to re-develop topic models under the VAE framework [17]–[20]. In this paper, we study topic modeling under the framework of VAE. And we aim to overcome two drawbacks in most of the existing studies.

First, despite that VAE is powerful for posterior inference, the posterior distribution is usually chosen to be a known and simple one, which limits the flexibility of the model [21]. For topic modeling, latent variables can be considered higher and more abstract concepts that could generate the words of documents. The usually chosen multivariate diagonal Gaussian is a rigid choice for generating documents due to its unimodal structure.

Second, the framework of VAE is mainly designed for unsupervised learning. The extension to supervised or semi-supervised learning is non-trivial [22]–[24]. Labeling data is admittedly very expensive for text data in reality, but obtaining a small subset of labeled data is much more practicable. And these labeled data have proven to be valuable to improve the performance of learning algorithms [25]–[27]. Semi-supervised topic modeling has attracted much attention recently. However, adopting the VAE framework for semi-supervised topic learning is still rare [28].

We propose a Semi-supervised Variational AutoEncoder with Gaussian Mixture posteriors (S-VAE-GM) to address the above challenges in topic modeling. We assume that each document is endowed with both class labels and latent topics. Topics can be seen as a relatively high dimensional vector of latent variables that follow a multivariate Gaussian distribution. Each class label, which can be observed for a subset of the data, corresponds to a Gaussian with its specific parameters. This assumption means that topics weigh differently for different classes. A document corresponds to a discrete distribution over classes. And this naturally results in a mixture of Gaussians for a document. The Gaussian mixture model for latent space is rational for the assumption of the class-topic hierarchy in a document, and it alleviates the unimodal limitation. The assumptions are illustrated in Figure 1. Specifically, the representation vectors of a document will be input into two networks to encode the parameters of latent variables, one network for the parameters of each Gaussian component, and the other one for the mixture weights of Gaussians. Latent variables are sampled from this Gaussian mixture density, and they are then used to generate the original document by independently generating the representation of each word in the document. We carefully derive the objective for semi-supervised topic learning. The objective takes into consideration the KL divergence between two Gaussian mixture densities and the cross entropy term for training the classifier with labeled data. Experiments performed on three standard datasets demonstrate the effectiveness of our model.

II. RELATED WORK

In this section, we review previous studies on topic models using the VAE framework, and VAE models with multimodal approximate posterior distributions.

A. VAE-BASED TOPIC MODELS

The computation burden of inferring the posteriors in topic models challenges the application and development of them. Several previous studies adopted variational autoencoder, which trains a black-box inference network for approximating posteriors, to avoid variational updates.

Miao et al. [17] proposed a Neural Variational Document Model (NVDM). The model approximates the posterior distribution of latent variables by building an inference network, which is implemented by a deep neural network conditioned on text. Experiments on two standard news corpora show the lowest perplexity compared to several baseline methods. The NVDM is a fully unsupervised document model. And the approximate posterior of latent variable is assumed to be Gaussian. To tackle the drawback of costly computation of the posterior distribution in applying traditional topic models, Srivastava and Sutton [18] proposed the AVITM model for latent Dirichlet allocation. The Dirichlet prior of topic proportions of documents hinders the development of parameterization trick. It is solved by constructing a Laplace approximation to the Dirichlet prior. The authors also solved the component collapsing problem for their model. Experiments on two datasets demonstrated the effectiveness and efficiency of their model. AVITM can be seen as a neural extension of LDA, while NVDM does not consider the Dirichlet prior. Yang et al. [19] investigated a new type of decoder, a dilated CNN in VAE-based text modeling. By controlling the size of context, the model can outperform LSTM language models. Xiao et al. [20] introduced a topic distribution variable to explicitly involve Dirichlet latent variable. Based on LSTM specifications, the authors reported gains over several baselines on the tasks of text reconstruction and classification.
The studies mentioned above are mainly designed for unsupervised learning. And the posteriors of the latent variables are uni-modal Gaussian or Dirichlet. In this paper, we aim to develop a semi-supervised deep variational model for text data. In addition, to enhance the interpretability and flexibility of the model, we use a multimodal distribution for the posteriors of latent variables.

B. VAE WITH MULTIMODAL POSTERIORS

In order to increase the flexibility of variational autoencoder, some studies aim to enrich the form of the posterior distributions of latent variables. A natural choice is using multimodal posteriors.

Dilokthanakul et al. [29] proposed a VAE-based supervised clustering method that assumes the observed data is generated from a mixture of Gaussians. The generation process is analogous to Gaussian discriminant analysis. Their method selects a cluster label at first, and then the Gaussian corresponding to this label is activated. Experiments on image datasets demonstrate the effectiveness of clustering and image generation. Liu et al. [30] adopted Gaussian mixture distribution to capture the true posterior of latent variables. During implementation, the weights of the components of Gaussian mixture are set to be uniform. The authors reported state-of-the-art experimental results on several image datasets. In order to attenuate the effect of outliers in VAE-based clustering, Zhao et al. [31] proposed a truncated Gaussian-mixture VAE to model major and minor clusters separately. Experiments on a synthetic image dataset and a medical image dataset demonstrated the effectiveness in disentangling minor and major clusters.

All the studies mentioned above adopted some forms of mixture models for the posteriors of latent variables. However, a majority of existing studies aim to design unsupervised models for image data. While in this paper, we focus on semi-supervised learning for text data.

III. MODEL FRAMEWORK

We now describe our novel S-VAE-GM in which we use a Gaussian mixture model (GMM) for the approximate posterior distribution of latent variables.

A. PROBLEM DESCRIPTION

A corpus of documents can be denoted as \( X = \{x_1, x_2, \ldots, x_N\} \). \( x_i \in \mathbb{R}^{|V|} \) is the representation vector of the \( i \)-th document where \( V \) denotes the vocabulary. \( x_i \) is assumed to be i.i.d. sampled from some distribution \( p(x) \). And we assume that the generation of \( x \) is conditioned on a continuous latent variable \( z \in \mathbb{R}^K \), where \( K \) is the dimension of latent space and it indicates the number of latent topics. The dataset can be divided into two parts, a subset with labels \( X_1 \) and a subset without labels \( X_0 \). The set \( X_1 \) can be further represented as \( X_1 = \{(x_i, y_i)\} \), where \( y_i \in \{y_1, y_2, \ldots, y_M\} \) denotes the label of document \( x_i \).

We aim to learn a) the distribution of labels for each document \( i \), denoted as \( p(y_i|x_i) \), \( (j = 1, \ldots, M) \), b) the distribution of topics for each document \( i \), denoted as \( p(z_k|x_i) \), \( (k = 1, \ldots, K) \), and c) the distribution of words over the vocabulary for each topic \( k \), denoted as \( p(v_t|z_k) \), \( (t = 1, \ldots, |V|) \) where \( v_t \) is the \( t \)-th word in vocabulary \( V \).

B. GENERATIVE PROCESS

By leveraging the framework of VAE, we propose a probabilistic model to describe the generation of the text data under semi-supervised settings. The objective is to model \( p(x) \). Latent variables are often incorporated in the model to encode intuitive features associated to observed ones. Similar to the work of Kingma et al. [23], we introduce a latent class variable \( y \) to the unlabeled data and a continuous feature variable \( z \) to all the data. According to the laws of probability, \( p(x|y, z) \) and \( p(y, z) \) will be modeled instead in order to involve these latent variables. And we assume that the joint distribution is factorized as \( p(y)p(z) \). The generation model is described as \( p(x, y, z) = p(y)p(z)p_0(x|y, z) \). And the observed data can be generated from latent variables \( y \) and \( z \) according to the following process, which corresponds to the “decoder” in VAE.

\[
p(\pi) = \text{SymDir}(\pi|\alpha) \hspace{1cm} p(y) = \text{Cat}(y|\pi) \hspace{1cm} p(z) = N(z|\theta, I) \hspace{1cm} p_0(x|y, z) = f(x; y, z, \theta)
\]

The variable \( \pi \) follows a symmetric Dirichlet distribution with hyper parameter \( \alpha \). \( \pi \) is a \( M \)-dimensional vector, in which each element \( \pi_j \) represents the weight of the corresponding label \( y_j \). Label \( y \) is often modeled as a categorical distribution based on \( \pi \) due to the Dirichlet-multinomial conjugacy. Latent variable \( z \) follows a standard normal distribution, and we will use a GMM to approximate its posterior distribution. Data \( x \) can be either a continuous embedding vector or a discrete bag-of-words vector. \( f \) is a likelihood function parameterized by \( \theta \). The likelihood is usually approximated by a neural network based on the latent variables \( y \) and \( z \).

C. VARIATIONAL OBJECTIVE

Typically, the log likelihood of the data \( \log p(x) \) will be maximized to obtain an applicable model. However, the integral of the marginal likelihood \( p(x) = \int \int p_0(x|y, z)p(y, z) \, dy \, dz \) is computational intractable since that we have to sample over a massive number of latent variables which are often with high dimensionality.

The framework of VAE aims to deal with the above problem [16]. The conditional likelihood \( p_0(x|y, z) \) will be modeled as an decoder. And the priors of latent variables \( p(y, z) \) will be inferred by the posterior \( p(y, z|x) \). To comply with the principles of variational inference, we use a simpler distribution \( q_0(y, z|x) \) to approximate the true posterior. The approximate posterior is evaluated by an encoder of VAE parameterized by \( \phi \) using variation inference. The objective is therefore minimizing the KL divergence (KLD) between the true and approximate posteriors. A few mathematical
derivations result in a lower bound of $\log p(x)$ for a single data point, namely the **Evidence Lower Bound (ELBO)**. The optimization objective of VAE is then transformed to the maximization of the lower bound.

Specifically, the forms of the ELBO for labeled and unlabeled data points of our model, are identical to the forms of Equation (6)-(9) of Kingma’s work. The differences are the implementations details of the approximate posteriors and the decoder network. Inspired by Kingma [23] and Keng’s work [32], we elaborate the details of the objective in the following.

We assume that the prior of latent variables can be fully factorized as

$$ p(y, z) = p(y)p(z) $$  

We also make the assumption that the approximate posterior of latent variables can be factorized as

$$ q_\phi(y, z|x) = q_{\phi_1}(y|x)q_{\phi_2}(z|x, y) $$  

### 1) LABELED DATA

For labeled data, $y$ can be observed. The latent variable is only $z$. And the ELBO is analogous to the vanilla VAE. The ELBO for a data point is written as

$$ \log p(x, y) \geq E_{q_\phi(z|x, y)}[p_\theta(x|y, z)] - D_{KL}[q_\phi(z|x, y) \| \ p(y)p(z)] $$

$$ = - \mathcal{L}(x, y) $$  

The objective that we want to minimize for all $(x, y) \in X_1$ is

$$ \mathcal{J}_l = E_{\tilde{p}_l(x, y)}\mathcal{L}(x, y) $$

where $\tilde{p}_l$ is the empirical distribution of labeled data.

To enable our model to train a classifier using partially labeled data, we involve a cross entropy term between the empirical distribution and the approximate posterior distribution of $y$ into our objective.

$$ \mathcal{J}_l = E_{\tilde{p}_l}(\mathcal{L}(x, y)) + E_{\tilde{p}_l}[-\log q_{\phi_1}(y|x)] $$

$$ = E_{\tilde{p}_l}(\mathcal{L}(x, y)) + \mathcal{H}[\tilde{p}_l, q_{\phi_1}] $$  

where $\mathcal{H}$ denotes the cross entropy between two distributions. The classifier $q_{\phi_1}$ produces predicted class labels by learning a mapping from $x$ to $y$. The added term will be used to supervise the training of the classifier.

### 2) UNLABELED DATA

For unlabeled data points, $y$ is considered as a latent variable. The ELBO is

$$ \log p(x) \geq E_{q_\phi}[\log p_\theta(x|y, z)] - D_{KL}[q_\phi(y, z|x) \| \ p(y, z)] $$

$$ = E_{q_\phi}[\log p_\theta(x|y, z) + \log p(y, z) - \log q_\phi(y, z|x)] $$  

The last line holds because of the definition of KLD. Here in Equation (6) the approximate posterior $q_\phi$ has the form of $q_\phi(y, z|x)$. Consider Equation (2), then Equation (6) can be written as

$$ \log p(x) \geq E_{q_{\phi_1}, q_{\phi_2}}[\log p_\theta(x|y, z) + \log p(y) $$

$$ + \log p(z) - \log q_{\phi_1}(y|x) - \log q_{\phi_2}(z|x, y) ] $$

$$ = E_{q_{\phi_1}}[E_{q_{\phi_2}}[\log p_\theta(x|y, z) ] $$

$$ - \log q_{\phi_2}(y|x) $$

$$ = E_{q_{\phi_1}}[-\mathcal{L}(x, y) - \log q_{\phi_1}(y|x) ] $$

The objective that we aim to minimize for all unlabeled data $x \in X_u$ is

$$ \mathcal{J}_u = E_{\tilde{p}_u(x)}[E_{q_{\phi_1}}[\mathcal{L}(x, y)] - E_{q_{\phi_1}}[-\log q_{\phi_1} ] ] $$

$$ = E_{\tilde{p}_u(x)}E_{q_{\phi_1}}\mathcal{L}(x, y) - H[y|x] $$  

where the second term on the right side of the last line is actually the conditional entropy of $y$ given unlabeled data $x$.

### 3) THE FINAL OBJECTIVE

The objective function for all data is

$$ \mathcal{J} = \mathcal{J}_l + \mathcal{J}_u $$

This is our final objective to be minimized with our S-VAE-GM model.

### IV. MODEL IMPLEMENTATION

In this section, we describe the details of implementations of the encoder and decoder in our S-VAE-GM model. The overall framework of the S-VAE-GM model is illustrated in Figure 2.

### A. BASIC ASSUMPTIONS

Our latent variables aim to capture some high-level feature space information in order to model the generation of text data based on latent topics. To achieve this goal, we make the following assumptions

1) Each class $y_j$, ($j = 1, \ldots, M$) corresponds to a normal distribution $\mathcal{N}(\mu_j, \text{diag}(\sigma_j^2))$ over latent topics $z$. The dimension of the random vector $z$ corresponds to the number of topics.

2) For a document, the weight or probability of the appearance of $y_j$ in the document is specified by $\pi_j$.

3) A document is generated by a decoder network based on latent variables $y$ and $z$, where $y$ is sampled from a multinomial distribution parametrized by $\pi$ and $z$ is assumed to be sampled from a mixture of Gaussian distributions.

The first assumption defines a class label as a probability mass function over $K$ discrete topics. It is analogous to LDA’s idea that a document can be generated from the distributions over topics. Different classes allocate different weights on topics. Hence, they can be semantically differentiated. The second assumption is the key to semi-supervised learning as
well as to Gaussian mixture modeling of \( z \). The dimension of \( \pi \), which is denoted as \( M \), is set to as the same as the number of possible \( y \). On the one hand, \( \pi \) can be produced by training another encoder \( q_{\phi_1} \), which can be treated as a classifier to achieve transductive semi-supervised learning. On the other hand, learning the probabilities of different classes for a document just provides us an adequate way for learning the weights of Gaussian mixture components. The third assumption tells us that \( z \), which will be used to generate the document \( x \), is actually a random variable sampled from a mixture of Gaussians.

**B. IMPLEMENTATION DETAILS**

1) ENCODER

The approximate posterior distribution \( q_{\phi} \) is implemented as an inference model, which is parameterized by a neural-network-based inference network (aka encoder) under the framework of VAE. According to Equation (2), we implement two different inference networks for \( q_{\phi_1} \) and \( q_{\phi_2} \), respectively.

For latent variable \( y \), we use a neural network, which is shown as the network with blue nodes in Figure 2, to produce the parameter \( \pi \) of the multinomial distribution of \( y \).

\[
q_{\phi_1} = \text{Cat}(y|\pi_{\phi_1}(x)) \tag{9}
\]

where \( \pi_{\phi_1}(x) \) can be represented as an MLP network (parameterized by \( \phi_1 \)) that takes dense embedding vectors as inputs data \( x \). The network acts as a classifier to produce predicted labels of unlabeled data. In addition, the produced parameter \( \pi \) is also used for modeling \( z \).

For latent variable \( z \), we use a Gaussian mixture inference network. Specifically, the parameters of each Gaussian component are specified by another neural network, which is shown as the network with red nodes in Figure 2. And \( \pi_{\phi_1}(x) \) acts as the mixture weights. The approximate distribution of \( z \) is specified as

\[
q_{\phi_2}(z|x, y) = \sum_{j=1}^{M} \pi_j(x; \phi_1)N(z|\mu_j(x, y; \phi_2), \text{diag}(\sigma_j^2(x; \phi_2))) \tag{10}
\]

where \( \mu_j \) and \( \sigma_j^2 \) can be produced by an MLP network (parameterized by \( \phi_2 \)) that takes bag-of-words representation of documents as inputs data \( x \). The network outputs \( M \) pairs of the parameters. Equation (10) tells how the latent variable \( z \) is produced by the encoder network. \( z \) will be sampled from \( q_{\phi_2} \) using the re-parametrization trick.

The difference between our method and Kingma’s M2 model [23] is that, in our method both network \( q_{\phi_1} \) and \( q_{\phi_2} \) are used to produce the parameters of a Gaussian mixture model that is used for sampling \( z \), rather than directly using a Gaussian variable and the label \( y \) to generate \( x \) in the M2 model. The difference between our method and the VAEGH [30] is that the value of \( \pi \) is learnt from the encoder network, rather than set to \( 1/M \). And unlike VAEGH, our model is developed for semi-supervised learning. In addition, our model is particularly designed for learning text data. The key to this task involves the above mentioned GMM modeling of the latent space, and the implementation of the decoder. The latter will be detailed as follows.

2) DECODER

Since that the generation of \( z \) from the GMM distribution already uses the parameters of \( p(y) \), here we simplify the decoder network as \( p_{\theta_0}(x|z) \) by only using \( z \) to generate \( x \).

We adopt the implementation of the decoder in NVDM model [17] for our decoder network \( p_{\theta_0}(x|z) \). The input data \( x \in \mathbb{R}^{|V|} \) for encoder \( q_{\phi_2} \), which is used to produce the continuous latent variable \( z \), is a sparse bag-of-words representation of a document. Hence, \( x \) can be decomposed into...
word vectors \( w_d \), where \( D \) is the number of words in the document. \( w_d \) is the one-hot vector of the \( d \)-th word in the document. If \( w_d \) is at the \( t \)-th position of the vocabulary \( \mathcal{V} \), only the element \( w_d(t) \) equals 1. The decoding process of a document can be seen as independently decoding each word. The decoder can be represented as

\[
p_{\theta}(x|z) = \prod_{d=1}^{D} p_{\theta}(w_d|z)
\]

which is shown as the network with yellow nodes in Figure 2.

We aim to obtain a topic-word matrix \( R_{K \times |\mathcal{V}|} \) to represent the semantic elements of topics. NVDM assumes that matrix \( R \) represents linear correlations between topics and words, and it will be used to evaluate a score and further the conditional probability of \( w_d \) given \( z \) as

\[
\text{Score}(w_d; z; \theta) = -z^T R w_d - b w_d
\]

\[
p_{\theta}(w_d|z) \propto \exp\left[\text{Score}(w_d, z; \theta)\right]
\]

This softmax regression model for words is shared by all documents. After training, the above model will learn the probability mass function over vocabulary for each topic.

To optimize our objective, there is another issue needs to be addressed. Since the prior and approximate posterior in vanilla VAE are both Gaussian, the KLD in the objective can be computed with a closed form solution. However, under our GMM assumption, we need to derive a tight bound for the KLD of two Gaussian mixture densities.

### C. KLD BETWEEN TWO GMMS

Now we consider the computation of the KLD in Equation (3). Since that \( p(x) \) is a symmetric Dirichlet distribution and \( p(y) \) is a Dirichlet-multinomial distribution, \( p(y) \) can be treated as a constant in (3). Given the GMM implementation of \( q_{\phi_2}(x|y, z) \), we need to deal with the KLD between a Gaussian mixture distribution and a standard Gaussian distribution. Since standard Gaussian can be rewritten as a mixture of standard Gaussians with any normalized set of weights, i.e.,

\[
\mathcal{N}(0, I) = \sum_i \pi_i \mathcal{N}(0, I), \quad \sum_i \pi_i = 1
\]

The KLD in the objective can then be evaluated between two Gaussian mixtures. Several studies [30], [33], [34] have demonstrated that the KLD between two mixtures does not have a closed form solution, but it has an upper bound. Since that we aim to maximize the ELBO, which means the KLD between two mixtures \( \mathbb{D}_{KL}[q_{\phi_2}(x, y) || p(z)] \) should be minimized, we can then transform the minimization of the KLD to the problem of minimizing the upper bound of the KLD between two mixtures.

Specifically, the KLD between two probability mixture densities \( p = \sum_{i=1}^{n} \pi_i f_i \) and \( \bar{p} = \sum_{i=1}^{n} \bar{\pi}_i f_i \) is upper bounded as follows [30]

\[
\mathbb{D}_{KL}[p || \bar{p}] \leq \mathbb{D}_{KL}[\pi || \bar{\pi}] + \sum_{i=1}^{n} \pi_i \mathbb{D}_{KL}[f || \bar{f}]
\]

The details of the proof of (14) can be found in [30].

For two Gaussian mixture densities, we only need to derive the analytical form of the KLD between two multivariate Gaussians to obtain the analytical form of the upper bound of the KLD between two Gaussian mixtures. Fortunately, the KLD between two Gaussians has a closed form solution. Suppose we have two Gaussians \( \mathcal{N}(\mu_1, \Sigma_1) \) and \( \mathcal{N}(\mu_2, \Sigma_2) \). According to the definition of KLD, we have

\[
\mathbb{D}_{KL}[\mathcal{N}(\mu_1, \Sigma_1) || \mathcal{N}(\mu_2, \Sigma_2)]
\]

\[
= \int \mathcal{N}(z|\mu_1, \Sigma_1) \log \frac{\mathcal{N}(z|\mu_1, \Sigma_1)}{\mathcal{N}(z|\mu_2, \Sigma_2)} dz
\]

\[
= \frac{1}{2} \log \frac{\Sigma_2}{\Sigma_1} + \frac{1}{2} tr(\Sigma_2^{-1} \Sigma_1)
\]

\[
+ \frac{1}{2}(\mu_1 - \mu_2)^T \Sigma_2^{-1}(\mu_1 - \mu_2) - \frac{1}{2} n
\]

where \( n \) is the dimension of variable \( z \). Substitute the \( j \)-th component of our GMM implementation of (10) into (15), we have the KLD between a diagonal multivariate Gaussian distribution and a standard Gaussian distribution as follows

\[
\mathbb{D}_{KL}[\mathcal{N}(\mu_j, \sigma_j^2) || \mathcal{N}(0, I)]
\]

\[
= -\frac{1}{2} \sum_{k=1}^{K} \left( 1 + \log \sigma_j^2 - \sigma_j^2 - \mu_j^2 \right)
\]

where \( \mu_j \) and \( \sigma_j^2 \) are the \( k \)-th component of \( \mu_j(x, \phi_2) \) and \( \sigma_j^2(x; \phi_2) \), respectively.

Substitute our implementation of the two Gaussian mixtures (10) and (13) into (14), we obtain the upper bound \( \mathcal{L}(x, y) \) of \( \mathcal{L}(x, y) \) as

\[
\mathcal{L}(x, y) \leq \tilde{\mathcal{L}}(x, y)
\]

\[
= \sum_{j=1}^{M} \pi_j \left[ \log \frac{\pi_j}{\tilde{\pi}_j} - \frac{1}{2} \sum_{k=1}^{K} \left( 1 + \log \sigma_j^2 - \sigma_j^2 - \mu_j^2 \right) \right]
\]

And the final objective that we aim to minimize becomes

\[
\mathcal{J} = \mathbb{E}_{\tilde{p}_1} \tilde{\mathcal{L}}(x, y) + \mathbb{E}_{\tilde{p}_2} \mathbb{E}_{\theta_1} \tilde{\mathcal{L}}(x, y) + \mathcal{H}(y|x) - \mathcal{H}([\tilde{p}_1, q_{\phi}])
\]

### D. OPTIMIZATION

By using the reparameterization trick, the derivatives of the objective \( \mathcal{J} \) w.r.t. model parameters \( \theta, \phi_1 \) and \( \phi_2 \) can be estimated. Then the objective can be optimized by back-propagating the stochastic gradients w.r.t. model parameters.

### V. EXPERIMENTAL SETUPS

In this section, we describe the details of the datasets used to perform experiments, the metrics for performance evaluation, the baselines for comparison, and the settings of experiments.

### A. DATASETS

We use three standard, public available datasets to verify the effectiveness of our proposed model.
We use perplexity and pointwise mutual information to evaluate data statistics. The number of total samples, training samples, and testing samples for each dataset are listed. For all of the three datasets, only 20% of the training samples retain their labels. Now the datasets for semi-supervised topic learning have been established.

Table 1 summarizes the statistics of the three datasets. For the whole dataset, perplexity is designated as a classifier of documents. Hence, the number of classes is 2. The dataset is divided equally for training and testing.

C. BASELINES

We use both non-neural network-based and neural network-based baseline methods to compare the performances of our model.

- LDA. Latent Dirichlet Allocation (LDA) [9] is one of the most popular generative Bayesian probabilistic models for text corpora. A document is modeled as a mixture of latent topics. And a topic is modeled as a distribution over vocabulary. LDA utilizes Dirichlet-Multinomial conjugacy and several inference techniques to cluster documents according to topic distributions. It has been proved that LDA performs well on several tasks [39], [40].

- CTM. Correlated Topic Model (CTM) [10] is an extension of LDA. It incorporates correlations into topic models by using the logistic normal distribution to model the topic mixture of a document. Several studies demonstrate its effectiveness [41], [42].

- NVDM. Neural Variational Document Model (NVDM) [17] is among the first a few studies that utilize deep generative neural network model for text data learning. The model shows significant improvement, under the perplexity measure, over traditional LDA.

D. SETTINGS

To encode the weight parameters of GMM, we use dense embeddings of documents as the inputs of encoder $\phi_1$. The embeddings of the words are obtained from the pre-trained Glove model. And the embeddings of documents are obtained by summing the embeddings of the words in the documents. The dimension of embedding vector is set to 100. Encoder $\phi_2$ uses bag-of-words representation of documents as its inputs. For datasets 20NewsGroup, IMDB and AGNews, the vocabulary size is set to 2,000, 10,000 and 5,000, respectively. These words are the most frequent words in the corresponding datasets, after deleting the stop words using the NLTK toolkit [43]. The words that are not contained in the vocabulary are omitted.

The encoder $\phi_1$ is designated as a classifier of documents. We perform experiments with the TextCNN [44] model and

C. Zhou et al.: Gaussian Mixture VAE for Semi-Supervised Topic Modeling

106849

VOLUME 8, 2020
the MLP model. The TextCNN model tends to overfitting the data before the end of the training process of \( \phi_2 \). Hence, we select MLP for both \( \phi_1 \) and \( \phi_2 \). For \( \phi_1 \), the size of the input layer is the dimension of the embedding vector, which is 100. The size of the hidden layer is set to 500. And the size of the output layer is constrained to be the same as the number of labels of a dataset, since we aim to obtain the weights for each class. The size of the number of topics varies in \{25, 50, 100, 150, 200\} in our experiments. For encoder \( \phi_2 \), the size of the input layer is the same as the size of the vocabulary of a dataset. During implementation, the inputs of \( \phi_2 \) are encoded through the hidden layer into a token vector \( \rho = g(\phi_2(x)) \) where \( g \) is the nonlinear active function and \( f^{\text{MLP}}_{\phi_2} \) denotes the MLP with parameter \( \phi_2 \). The size of the hidden layer is set to 500, which is the dimension of the token vector. Then the token is used to produce the parameters of GMM through two different linear networks, i.e., \( \mu = l_1(\rho) \) and \( \log \sigma^2 = l_2(\rho) \). The sizes of \( l_1 \) and \( l_2 \) both equal in the number of classes, which is the number of components of the GMM model. The active function used in our experiments is \( \tanh(\cdot) \).

The decoder takes the hidden variable, which is sampled from the GMM and with the dimension of the number of topics, to be its input. It uses a softmax network to transform the hidden variable into the one-hot representation of words. The weights of the decoder network correspond to the \( K \times |V| \) matrix \( R \) where \( R \in \mathbb{R}^{|V|} \) produces the logit scores for each word, and after the softmax layer, the probabilities of each word are obtained. The probabilities, together with the bag-of-words vector \( x \), are used to compute the reconstruction error.

We use the Xavier uniform initializer in Tensorflow [45] to initialize the weights of our networks. We use the Adam optimizer with a learning rate of \( 10^{-4} \). During training, the batch size is set to 64. The maximum epoch is 1,000. After the training of each epoch, we compute the metrics mentioned in Section V-B. We save our model if the newly recorded metrics are better than the existed ones. The training process is terminated when the metrics in 10 consecutive epochs are not better than the recorded ones.

VI. EXPERIMENTAL RESULTS
In this section, we perform experiments with the proposed model on the above-mentioned three datasets. We report the results by comparison with the baselines.

A. COMPARATIVE RESULTS
The perplexity produced by our S-VAE-GM and other comparatives are shown in Figure 3. According to this figure, the following observations are made: (1) Our S-VAE-GM model achieves the lowest perplexity under almost all conditions. With the 20NewsGroup dataset, S-VAE-GM outperforms other baselines under every topic number. With datasets IMDB and AGNews, our method and NVDM achieve similar best performance. Non-NN methods have much higher perplexity values, especially when the number of topics increases. (2) The S-VAE-GM model is more robust, under the perplexity measure, against the change of the dimension of latent space. From these observations, we find that S-VAE-GM improves the performance of topic modeling, especially compared to non-NN methods.

The PMIs produced by S-VAE-GM and other methods are shown in Figure 4. With the 20NewsGroup dataset, both non-NN methods outperform NN-based methods. However, as the number of latent topics increases, the gap between these two kinds of methods is shrunk. With the other two datasets, S-VAE-GM and NVDM achieve similar best performance. We believe that the complexities among the results of PMI values have the following reasons: (1) The true structures of the latent space of the three datasets are very different. The interpretability of the produced topics with different methods varies a lot. (2) Topics may correlate with other topics. Hence, independent components assumption when modeling the latent space limits the performance of the methods. We will explore these issues in our future work.

The S-VAE-GM method and the NVDM method show similar performance except with the 20NewsGroup dataset. We believe this is due to similar implementations of the decoder. However, the semi-supervised learning framework
FIGURE 4. PMI (the higher the better). The Pointwise Mutual Information values are also recorded for different numbers of latent topics. Different lines represent different methods.

B. ABLATION EXPERIMENTAL RESULTS

In order to verify the effectiveness of the Gaussian mixture model for latent space, we change the GMM of $z$ to a single multivariate Gaussian and perform the experiments using the same settings of S-VAE-GM. The perplexity and PMI values using a single Gaussian under each number of topics are shown in Table 2 and 3, compared with multimodal Gaussian mixture model.

From the tables, we observe that with the 20NewsGroup dataset, using GMM significantly improves the performance under the criteria of perplexity and PMI. Perplexity and PMI are improved by an average of 21.30% and 6.49%, respectively. For all three datasets, perplexity is improved by an average of 7.61%, and PMI is improved by an average of 2.01%. Introducing GMM to latent space has improved the performance under a majority of conditions (23 out of 30).

C. LATENT TOPICS

Each dimension in the latent space is assumed to be associated with a latent topic. In Table 4, we list some randomly selected topics with their top 10 most relevant words.

D. CLASSIFIER EVALUATION

In order to evaluate the performance of classifier $\phi_1$ in S-VAE-GM, we compute the $F_1$-score of the prediction results. Since only 20% of the data are used for training, the remaining 80% are used for testing the predicted results produced by $\phi_1$. During experiments, we observe that the number of latent topics, which varies in $\{25, 50, 100, 150, 200\}$, does not affect the performance of the classifier. Hence, we do not distinguish the results under different number of latent topics. For comparison, we train an MLP particularly for classifying the text data, with $\{20\%, 40\%, 60\%, 80\%\}$ data as training samples, respectively.

In Figure 5, each box-plot indicates the distribution of $F_1$-scores computed for specific labels. The results of

---

TABLE 2. Perplexity (the lower the better). The perplexity values under each number of latent topics are shown above, for all three datasets. Uni- indicates the model with a single multivariate Gaussian, while Multi- indicates the model with a Gaussian mixture model. These two models are compared under each condition. The better results are marked with bold font.

| # of Topics | 20NewsGroup | IMDB       | AGNews     |
|-------------|-------------|------------|------------|
|             | Uni-        | Multi-     | Uni-       | Multi-     |
| 25          | 895         | 732        | 1970       | 1962       | 993        | 988        |
| 50          | 941         | 748        | 2018       | 1984       | 949        | 959        |
| 100         | 1016        | 801        | 2112       | 2064       | 935        | 943        |
| 150         | 1068        | 819        | 2166       | 2114       | 956        | 949        |
| 200         | 1098        | 842        | 2176       | 2153       | 947        | 942        |

TABLE 3. PMI (the higher the better). The PMI values under each number of latent topics are shown above, for all three datasets. Uni- and Multi- indicate the same as in Table 2. The better results are marked with bold font.

| # of Topics | 20NewsGroup | IMDB       | AGNews     |
|-------------|-------------|------------|------------|
|             | Uni-        | Multi-     | Uni-       | Multi-     |
| 25          | 0.27        | 0.31       | 0.40       | 0.43       | 0.56        | 0.59        |
| 50          | 0.28        | 0.30       | 0.51       | 0.42       | 0.57        | 0.57        |
| 100         | 0.28        | 0.29       | 0.44       | 0.46       | 0.54        | 0.53        |
| 150         | 0.29        | 0.30       | 0.45       | 0.43       | 0.54        | 0.53        |
| 200         | 0.29        | 0.30       | 0.44       | 0.45       | 0.52        | 0.54        |

As shown in the table, we can deduce corresponding topics based on the words. We observe that when the dimension of latent space increases, it is more likely to discover more precise topics. For example, we can associate the third column of words under dataset AGNews with the topic Sports. But after a careful examination, we find these words are all relevant to Baseball, a branch of Sports. We also observe some less relevant words. They are astronauts and NASA in the Political topic of IMDB, and chips in the Economy topic of AGNews. However, in view of that the topic Space is sometimes correlated with the topic Political in some movies, and that chips may represent the topic Industrial production, we believe that these words are not totally unrelated with their corresponding topics. Except for these words, all other words are closely related to their corresponding topics. Hence, the produced topics are locally interpretable and semantically coherent.
TABLE 4. The latent topics learned by S-VAE-GM for three datasets. The numbers in parentheses indicate the dimension of latent space that produces the corresponding topic words. Each column is an ordered list of the top 10 words with the highest weights in that dimension.

| 20NewsGroup | IMDB | AGNews |
|-------------|------|--------|
| **Computer (25)** | **Religion (50)** | **Sports (50)** | **Thril (25)** | **Political (25)** | **Action (50)** | **Attack (50)** | **Economy (100)** | **Baseball (150)** |
| toolkit | Jesus | players | zombie | Amir | martial | killing | Wall | homered |
| database | Christians | terns | troma | capitalism | Chan | rocked | Nasdaq |ining |
| interface | god | pitching | uwe | Chavez | arts | occurred | chips | Ramirez |
| Unix | sin | season | ball | democracy | Jackie | killed | profit | preseshon |
| X11 | scripture | NIH | slashers | astronauts | Sammo | bomb | street | touchdowns |
| libraries | faith | team | mutated | MacArthur | King | witness | PL C | Ortiz |
| xt | gods | player | bitten | presidential | Li | casualties | grew | Busch |
| platforms | Christ | league | cannibals | democratic | stunts | wounding | outlook | Manny |
| visual | Bible | playoffs | scarecrow | allies | plainly | exploded | lower | pitched |
| formats | atheism | puck | zombie | NASA | Fu | deadliest | customers | NL |

FIGURE 5. Classifier evaluation with F1 score. S-VAE-GM indicates the classifier in our method, which masks 80% data and leaves the rest 20% data with labels. We train another neural network as the baseline for comparison. The number marked on the x-axis means the percentage of labeled data used for training the baseline.

S-VAE-GM, compared to the results of the baseline with an increase in the percentage of training samples, demonstrate that the performance of classifier φ1 is not markedly reduced in view of the combination of φ1 and φ2 and the incorporation of the cross entropy term in the combined objective.

VII. CONCLUSION

In this paper, we proposed a novel VAE-based semi-supervised topic model with a Gaussian mixture model for latent space. We used two separate encoders for the parameters of the Gaussian mixture components and the mixing weights. Latent variables were sampled from the Gaussian mixture to decode the words in documents. We also used the weights, together with partially labeled data for training the classifier. The objectives for both labeled and unlabeled data were carefully derived, and the modules of the framework were appropriately designed. Experiments were performed on three public-available benchmark datasets, 20NewsGroup, IMDB and AGNews. The results under the perplexity and the PMI criteria demonstrated the effectiveness of the proposed topic model, comparing to both non-neural network-based and neural network-based baseline methods.

In the future, we intend to explore topic correlations using both GMM and normalizing flows [46], [47] under the VAE framework.

REFERENCES

[1] D. M. Blei and J. D. Lafferty, “Topic models,” in Text Mining. Boca Raton, FL, USA: CRC Press, 2009, pp. 101–124.
[2] D. M. Blei, “Probabilistic topic models,” Commun. ACM, vol. 55, no. 4, pp. 77–84, Apr. 2012.
[3] A. McCallum, A. Corrada-Emmanuel, and X. Wang, “The author-recipient-topic model for topic and role discovery in social networks: Experiments with Enron and Academic email,” in Computer Science Department Faculty Publication Series, 2005, p. 44.
[4] D. Newman, C. Chemudugunta, P. Smyth, and M. Steyvers, “Analyzing entities and topics in news articles using statistical topic models,” in Proc. Int. Conf. Intell. Secur. Inform. Berlin, Germany: Springer, 2006, pp. 93–104.
[5] L. Hong and B. D. Davison, “Empirical study of topic modeling in Twitter,” in Proc. 1st Workshop Social Media Anal. (SOMA), 2010, pp. 80–88.
[6] Q. He, B. Chen, J. Pei, B. Qiu, P. Mitra, and L. Giles, “Detecting topic evolution in scientific literature: How can citations help?” in Proc. 18th ACM Conf. Inf. Knowl. Manage. (CIKM), 2009, pp. 957–966.
[7] C. Arnold and W. Speier, “A topic model of clinical reports,” in Proc. 35th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR), 2012, pp. 1031–1032.
[8] X. Xia, D. Lo, Y. Ding, J. M. Al-Kofahi, T. N. Nguyen, and X. Wang, “Improving automated bug triaging with specialized topic model,” IEEE Trans. Softw. Eng., vol. 43, no. 3, pp. 272–297, Mar. 2017.
[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet allocation,” J. Mach. Learn. Res., vol. 3, pp. 993–1022, Mar. 2003.
[10] J. D. Lafferty and D. M. Blei, “Correlated topic models,” in Proc. Adv. Neural Inf. Process. Syst., 2006, pp. 147–154.
[11] M. Hoffman, F. R. Bach, and D. M. Blei, “Online learning for latent Dirichlet allocation,” in Proc. Adv. Neural Inf. Process. Syst., 2010, pp. 856–864.

[12] A. J. Perotte, F. Wood, N. Elhadad, and N. Bartlett, “Hierarchically supervised latent Dirichlet allocation,” in Proc. Adv. Neural Inf. Process. Syst., 2011, pp. 2690–2698.

[13] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, “Variational inference: A review for statisticians,” J. Amer. Stat. Assoc., vol. 112, no. 518, pp. 859–877, 2017.

[14] A. Graves, “Practical variational inference for neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2011, pp. 2348–2356.

[15] R. Ranganath, S. Gerrish, and D. M. Blei, “Black box variational inference,” in Proc. 17th Int. Conf. Artif. Intell. Statist., 2014, pp. 1–9.

[16] D. P. Kingma and M. Welling, “Auto-encoding variational Bayes,” 2013, arXiv:1312.6114. [Online]. Available: http://arxiv.org/abs/1312.6114.

[17] Y. Miao, L. Yu, and P. Blunsom, “Neural variational inference for text processing,” in Proc. 33rd Int. Conf. Int. Conf. Mach. Learn., vol. 48, 2016, pp. 1727–1736.

[18] A. Srivastava and C. Sutton, “Autoencoding variational inference for topic models,” 2017, arXiv:1703.04188. [Online]. Available: http://arxiv.org/abs/1703.04188.

[19] Z. Yang, Z. Hu, R. Salakhutdinov, and T. Berg-Kirkpatrick, “Improved variational autoencoders for text modeling using dilated convolutions,” in Proc. 34th Int. Conf. Mach. Learn. Sydney, NSW, Australia: PMLR, 2017, pp. 3881–3890.

[20] Y. Xiao, T. Zhao, and W. Yang Wang, “Dirichlet variational autoencoder for text modeling,” 2018, arXiv:1811.00135. [Online]. Available: http://arxiv.org/abs/1811.00135.

[21] Y. Burda, R. Grosse, and R. Salakhutdinov, “Importance weighted autoencoders,” in Proc. 4th Int. Conf. Learn. Represent. (ICLR), 2016, pp. 1–14.

[22] J. Jagarlami, H. Daumé, III, and R. Udupa, “Incorporating lexical priors into topic models,” in Proc. 13th Conf. Eur. Chapter Assoc. Comput. Linguistics, 2012, pp. 204–213.

[23] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, “Semi-supervised learning with deep generative models,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 3581–3589.

[24] C. Li, S. Chen, J. Xing, A. Sun, and Z. Ma, “Seed-guided topic model for document filtering and classification,” ACM Trans. Inf. Syst., vol. 37, no. 1, pp. 1–37, Dec. 2018.

[25] K. Nigam, A. McCallum, and T. Mitchell, “Semi-supervised text classification using EM,” in Semi-Supervised Learning, Cambridge, MA, USA: MIT Press, 2011, pp. 2609–2617.

[26] X. Zha and A. B. Goldberg, “Introduction to semi-supervised learning,” Synth. Lect. Artif. Intell. Mach. Learn., vol. 3, no. 1, pp. 1–130, 2009.

[27] T. Miyato, A. M. Dai, and I. Goodfellow, “Adversarial training methods for semi-supervised text classification,” 2016, arXiv:1605.07725. [Online]. Available: http://arxiv.org/abs/1605.07725.

[28] W. Xu, H. Sun, C. Deng, and Y. Tan, “Variational autoencoder for semi-supervised text classification,” in Proc. 31st AAAI Conf. Artif. Intell., 2017, pp. 3338–3344.

[29] N. Dilokthanakul, P. A. M. Mediano, M. Garnelo, M. C. H. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan, “Deep unsupervised clustering with Gaussian mixture variational autoencoders,” 2016, arXiv:1611.02648. [Online]. Available: http://arxiv.org/abs/1611.02648.

[30] G. Liu, Y. Liu, M. Guo, P. Li, and M. Li, “Variational inference with Gaussian mixture model and householder flow,” Neural Netw., vol. 109, pp. 43–55, Jan. 2019.

[31] Q. Zhao, N. Honnorat, E. Adeli, and K. M. Pohl, “Truncated Gaussian mixture variational AutoEncoder,” 2019, arXiv:1902.03717. [Online]. Available: http://arxiv.org/abs/1902.03717.

[32] B. Keng. (2017). Semi-Supervised Learning With Variational Autoencoders. Accessed: Jan. 30, 2020. [Online]. Available: http://bijkeng.github.io/posts/semi-supervised-learning-with-variational-autoencoders/

[33] Y. Singer and M. K. Warmuth, “Batch and on-line parameter estimation of Gaussian mixtures based on the joint entropy,” in Proc. Adv. Neural Inf. Process. Syst., 1999, pp. 578–584.

[34] M. N. Do, “Fast approximation of Kullback-Leibler distance for dependence trees and hidden Markov models,” IEEE Signal Process. Lett., vol. 10, no. 4, pp. 115–118, Apr. 2003.

[35] X. Zhang, J. Zhao, and Y. LeCun, “Character-level convolutional networks for text classification,” in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 649–657.

[36] P. Resnik and J. Lin, “Evaluation of NLP systems,” in The Handbook of Computational Linguistics and Natural Language Processing. Oxford, U.K.: Blackwell, 2010, pp. 271–295.

[37] A. Mnih and K. Gregor, “Neural variational inference and learning in belief networks,” in Proc. 31st Int. Conf. Int. Conf. Mach. Learn., vol. 32, 2014, pp. 1719–1727.

[38] N. Aletras and M. Stevenson, “Evaluating topic coherence using distributional semantics,” in Proc. 10th Int. Conf. Comput. Semantics (IWCS), 2013, pp. 13–22.

[39] Z. Liu, M. Li, Y. Liu, and M. Porraj, “Performance evaluation of latent Dirichlet allocation in text mining,” in Proc. 8th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD), vol. 4, Jul. 2011, pp. 2695–2698.

[40] H. Jelodar, X. Wang, C. Yuan, X. Feng, X. Jiang, Y. Li, and L. Zhao, “Latent Dirichlet allocation (LDA) and topic modeling: Models, applications, a survey,” Multimedia Tools Appl., vol. 78, no. 11, pp. 15169–15211, Jun. 2019.

[41] D. M. Blei and J. D. Lafferty, “A correlated topic model of science,” Ann. Appl. Statist., vol. 1, no. 1, pp. 17–35, Jun. 2007.

[42] G. Sun, Y. Li, W. X. Zhao, J. Gao, and A. Zhang, “A correlated topic model using word embeddings,” in Proc. 26th Int. Joint Conf. Artif. Intell., Aug. 2017, pp. 4207–4213.

[43] J. Perkins, Python 3 Text Processing With NLTK 3 Cookbook. Birmingham, U.K.: Packt, 2014.

[44] Y. Kim, “Convolutonal neural networks for sentence classification,” in Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP), 2014, pp. 1746–1751.

[45] M. Abadi et al., “TensorFlow: A system for large-scale machine learning,” in Proc. 12th USENIX Symp. Operating Syst. Design Implement. (OSDI), 2016, pp. 265–283.

[46] J. M. Tomczak and M. Welling, “Improving variational auto-encoders using householder flow,” 2016, arXiv:1611.09630. [Online]. Available: http://arxiv.org/abs/1611.09630.

[47] L. Liu, H. Huang, Y. Gao, Y. Zhang, and X. Wei, “Neural variational correlated topic modeling,” in Proc. World Wide Web Conf., 2019, pp. 1142–1152.

CANGQI ZHOU received the B.S. degree in automation from the Beijing Institute of Technology, Beijing, China, in 2008, and the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, in 2015.

From 2015 to 2017, he was a Postdoctoral Research Fellow with the Center for Intelligent and Networked Systems (CFINS), Tsinghua University. Since 2017, he has been an Assistant Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His research interests include statistical learning for text data, social network data mining, network representation learning, and community detection.

HAO BAN received the B.S. degree in information engineering from Southeast University, Nanjing, China, in 2018, where he is currently pursuing the M.S. degree in information and communication engineering.

From 2019 to 2020, he was a Research Intern at SenseDeal, Nanjing. His research interests include topic modeling, sentiment analysis, and deception detection.

He was the Second Prize Winner of the Huawei Cup – the 14th China Post-Graduate Mathematical Contest in Modeling, in 2018.
JING ZHANG (Senior Member, IEEE) received the Ph.D. degree in computer science from the Hefei University of Technology, Hefei, China, in 2015. He is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. He has published dozens of articles in prestigious journals and international conferences. His research interests include data mining, machine learning, and their applications in business and industry. He serves as a PC member of a number of international conferences and a reviewer for more than 20 international journals.

QIANMU LI received the B.S. and Ph.D. degrees from the Nanjing University of Science and Technology, China, in 2001 and 2005, respectively. He is currently a Full Professor with the School of Cyber Science and Engineering, Nanjing University of Science and Technology. His research interests include information security and data mining. He received the China Network and Information Security Outstanding Talent Award, in 2016, and the Education Ministry Science and Technology Awards, in 2012.

YINGHUA ZHANG received the B.S. degree in computer science and technology from Shandong University, China, in 2007, and the Ph.D. degree in pattern recognition and intelligent systems from the University of Chinese Academy of Sciences, China, in 2013. From 2013 to 2018, she was an Associate Professor at the Institute of Automation, Chinese Academy of Sciences (CASIA). Since 2018, she has been the Vice President of Technology at SenseDeal Intelligent Technology Company Ltd. Her research interests include machine learning, natural language processing, and its application in financial field.

***