IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 20, 2018, accepted December 9, 2018, date of publication January 10, 2019, date of current version February 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2891902

Hierarchical Topic Modeling of Twitter Data for Online Analytical Processing

DONGJIN YU, (Member, IEEE), DENGWEI XU, DONGJING WANG, AND ZHIYONG NI
School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China

Corresponding author: Dongjin Yu (yudj@hdu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61472112 and Grant 61702144, in part by the Key Science and Technology Project of Zhejiang Province under Grant 2017C01010, and in part by the Natural Science Foundation of Zhejiang Province under Grant LY12F02003.

ABSTRACT Social platforms, such as Twitter, reveal much about the tastes of the public. Many studies focus on the content analysis of social platforms, which assists in product promotion and sentiment investigation. On the other hand, online analytical processing (OLAP) has been proven to be very effective for analyzing multidimensional structured data. The key purpose of applying OLAP to text messages, (e.g., tweets), called text OLAP, is to mine and construct the hierarchical dimension based on the unstructured text content. In contrast to the plain texts which text OLAP usually handles, the social media content includes a wealth of social relationship information which can be employed to extract a more effective dimensional hierarchy. In this paper, we propose a topic model called twitter hierarchical latent Dirichlet allocation (thLDA). Based on hierarchical latent Dirichlet allocation, thLDA aims to automatically mine the hierarchical dimension of tweets' topics, which can be further employed for text OLAP on the tweets. Furthermore, thLDA uses word2vec to analyze the semantic relationships of words in tweets to obtain a more effective dimension. We conduct extensive experiments on huge quantities of Twitter data and evaluate the effectiveness of thLDA. The experimental results demonstrate that it outperforms other current topic models in mining and constructing the hierarchical dimension of tweeters' topics.

INDEX TERMS Twitter, online analytical processing, topic modeling, hierarchical latent Dirichlet allocation, social media analysis.

I. INTRODUCTION

During the past few years, Twitter has become increasingly popular as an emerging social platform for messaging and communication among individuals. The huge quantities of Twitter data accumulated so far make it possible to discover the distribution and drift of mass tastes and opinions, which greatly assists in product recommendation, target marketing and so on. On the other hand, OLAP, or online analytical processing, enables analysis to interactively view data from all aspects in layered granularities, which has already been proven especially useful for business intelligence. Unfortunately, OLAP techniques are successful in dealing with cube data which are structured and formalized, but face difficulties in processing textual content such as Twitter data. To successfully apply OLAP techniques to Twitter, it is critical to mine the hidden representative dimensions from its extensive content.

As a typical unsupervised topic model, the Latent Dirichlet Allocation (LDA) model is efficient at statistically analyzing textual data for the underlying topics. In [1] and [2], we proposed a LDA-based model, called MS-LDA, to detect the hidden layered interests from the Twitter data. As the extension of LDA, MS-LDA integrated tweets and the social relationships among tweeters. Nevertheless, the primitive LDA model can only mine monolayer topics, rather than the hierarchical ones which OLAP requires. On the other hand, as an unsupervised hierarchical topic model, hLDA can obtain the sibling-sibling relationships between topics and can organize the topics into a hierarchical tree automatically. In fact, Twitter data contain abundant social behavioral information about tweeters, such as mentioning, retweeting and following. In addition, there exist some semantic relationships among the words in tweets, which may affect the effectiveness of the modeling process. In other words, to effectively discover the hidden layers of topics from Twitter data for constructing the hierarchical dimension for OLAP, we need to propose a new topic model which leverages the characteristics of Twitter in its modeling process.

In this paper, we focus on how to discover the underlying topics of tweets from tweeters' social behaviors and from their published tweets. Such topics can be then organized into one very important hierarchical dimension, or topic dimension, for applying OLAP to Twitter data. We present a model called thLDA to extract the hidden-layer topics from Twitter data for the multidimensional analysis of tweets' topics. The process is briefly described as follows. Firstly, we collect a primitive corpus through Twitter's APIs. Then, we preprocess the Twitter data by removing stop words and irrelevant data such as short links, short tweets and junk information. Subsequently, we analyze the social relationships of tweeters and the semantic relationships between words in tweets. Finally, we mine the topics from the Twitter data and organize them into a hierarchical structure based on thLDA.

The main contribution of this paper is threefold. (1) We introduce a novel hierarchical model called thLDA to construct a dimension hierarchy of tweets' topics, incorporating social relationships and semantic relationships into the modeling process. (2) We make use of word2vec, which is a two-layer neural network model, to obtain the semantic relationships between words in tweets, to improve the mining of the topics. (3) We conduct extensive experiments on our model with large quantities of Twitter data and find that the results demonstrate its effectiveness.

The remainder of this paper is organized as follows. After introducing the state-of-the-art related works in Section 2, we present some preliminaries necessary for understanding the paper in Section 3. Section 4 elaborates our proposed thLDA model and demonstrates the mathematical derivation process of thLDA, and Section 5 presents the experimental results and the comparison with other models, undertaken to verify the effectiveness of thLDA. Finally, we draw conclusions about our model and outline future work in Section 6.

II. RELATED WORKS

OLAP is an approach to answering multidimensional analytical queries over the cube data. It provides the operations such as rolling up, drilling down and slicing [3]. The goal of OLAP is to provide decision support or ad-hoc reporting. Its core technology is the concept of ''dimensions,'' which are usually multiple and hierarchical. Based on dimensions, OLAP aggregates the ''measured'' data by averaging, counting, totaling and so on.

Traditional OLAP can effectively analyze structured multidimensional data. However, it cannot handle unstructured data such as tweets [4]. In order to apply OLAP technology to the analysis of unstructured textual data, the concept of text OLAP is proposed [5]. Based on traditional OLAP technology, text OLAP aims to provide aggregative functions that summarize unstructured text data [6], [7]. For instance, Azabou *et al.* [8] present a novel model which serves as a basis for semantic OLAP for documents.

How to accurately and effectively mine tweets' topics from social data has long been the focus of research in the field of natural language processing. For example, Michelson and Macskassy *et al.* [9] present a topic profile to characterize tweets' topics. Cuzzocrea *et al.* [10] introduce an aggregation operator for tweets' content by using formal concept analysis theory. Liu *et al.* [11] propose a text cube approach to learning various types of social, human and cultural behaviors embedded in the Twitter data. Rehman *et al.* [12] focus on incorporating extensive natural language process technology in OLAP, to analyze multidimensional social data.

In addition, many researchers employ machine learning techniques to analyze social media. Siswanto *et al.* [13] propose a model that utilizes supervised learning-based classification based on tweeters' labels and specific accounts. Pennacchiotti and Popescu [14] propose a generic machine learning framework for tweeter classification, based on four general feature types: tweeter profile, tweeting behavior, linguistic content of tweeter's message and tweeter social network features. Pu *et al.* [15] present a mixed method which combines text mining and Wikipedia to mine tweeters' topics in Twitter data. Vathi *et al.* [16] propose a model based on a topic model to mine tweeters' clustered discussion topics and to design a method for excluding trivial topics. Furthermore, combining a topic model with analysis of Twitter data, Zhao *et al.* [17] propose a method called Twitter-LDA which aims to mine tweeters' topics from a typical sample of Twitter as a whole. However, this can only mine the topics from the Twitter data and does not take into consideration the hierarchical aspects of the topics. Based on LDA, Blei and Mcauliffe [18] propose sLDA (supervised Latent Dirichlet Allocation). In sLDA, Blei *et al.* add to LDA a response variable associated with each document. In order to find latent topics that will best predict the response variables for future unlabeled documents, sLDA jointly model the documents and the responses.

In order to obtain the topic hierarchy from the textual data, some researchers have focused on how to extend the traditional topic modeling techniques to obtain hierarchical information on the topics. The technique of hLDA [19], based on the notion of nCRP (nested Chinese restaurant process) [20], can simultaneously mine topics and construct the topic hierarchy by analyzing the relationships of topics without supervision. On the basis of hLDA, Mao *et al.* [21] propose a semi-supervised hierarchical topic model which aims to explore new topics automatically in the data space while incorporating the information from observed hierarchical labels into the modeling process, called Semi-Supervised Hierarchical Latent Dirichlet Allocation (SSHLDA). Wang *et al.* [22] also propose a semi-supervised hierarchical topic model, which aims to explore more reasonable topics in the data space by incorporating some constraints into the modeling process that are extracted automatically, denoted as constrained hierarchical Latent Dirichlet Allocation (constrained-hLDA). Dai and Storkey [23] propose the sHDP (supervised hierarchical Dirichlet process) process, which is a nonparametric generative model for the joint distribution of a group of observations and a response

variable directly associated with that whole group. Chien [24] presents a HPYD (hierarchical Pitman-Yor-Dirichlet) process as the nonparametric priors to infer the predictive probabilities of the smoothed n-grams with the integrated topic information. Teh [25] proposes a new hierarchical Bayesian n-gram model of natural languages, which makes use of a generalization of the commonly used Dirichlet distributions called Pitman-Yor processes which produce power-law distributions more closely resembling those in natural languages.

A further challenge is that many classification tasks on short text, such as tweet, fail to achieve high accuracy due to data sparseness. Up to now, several works have been done in the field to solve the problem by finding more effective word embedding models. Li *et al.* [26] present several tweet topic classification methods by exploiting different types of data: tweet text, tweet text plus entity knowledge base, word embeddings derived from tweet text, distributed representations of tweets, and topical word embeddings. A follow-up study by Ganguly *et al.* [27] focus on the use of word embeddings for enhancing retrieval effectiveness. In particular, they construct a generalized language model. Enríquez *et al.* [28] show how a vector-based word representation obtained via word2vec helps to improve the results of a document classifier based on bags of words. They have also performed cross-domain experiments in which word2vec has shown much more stable behavior than bag of words models. Zhang *et al.* [29] propose a method for sentiment classification based on word2vec and SVMperf in order to obtain the semantic features. The experimental results show the superior performance of their method in sentiment classification.

In contrast to the studies mentioned above, the thLDA model proposed in this paper can mine tweets' topic hierarchy automatically from Twitter data while considering the semantic relationships between words in tweets and the social relationships between tweeters. The final hierarchy of topics has proven to be suitable for the multidimensional analysis of Twitter data.

III. PRELIMINARIES

A. TWITTER DATA

Twitter involves two entities, i.e., tweets and tweeters. Here, the term ''tweets'' refers to the content published by tweeters together with properties such as ''id,'' ''place'' and ''FavoriteCount,'' whereas ''tweeters'' have their own properties like ''uid,'' ''location'' and ''name'' and a set of behaviors including ''retweeting,'' ''mentioning'' and ''following''. On the other hand, Twitter data can also be divided into two parts, i.e., the structured and unstructured parts. The structured data, such as ''id'' and ''location,'' do not require additional preprocessing for OLAP. However, the unstructured data, including text messages, emoticons, short links, etc., require special treatment for OLAP. In particular, the topics that tweeters discuss must be extracted as one of the dimensions which OLAP may employ to explore the Twitter data.

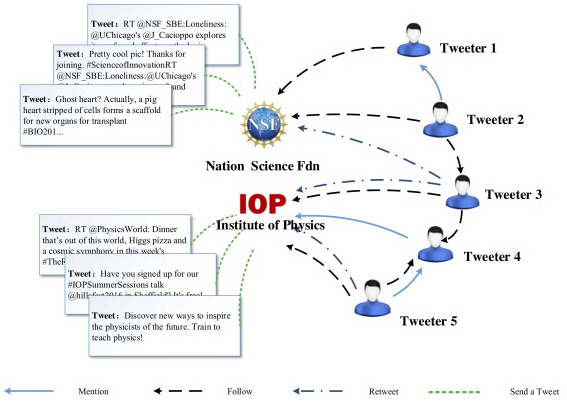


FIGURE 1. An example of social relationships among tweeters.

Figure 1 shows the social relationships between tweeters. In fact, tweeters possess abundant social behaviors, including following, mentioning and retweeting. These social behaviors are of great significance in mining the topics of tweeters. As shown in Figure 1, when the *Institute of Physics* sends a tweet, *Tweeters 3* and *5*, who follow it, will receive a notification, and may retweet the tweet if they are interested in it. Meanwhile, *Tweeter 5* can also mention it to his friend, *Tweeter 4*, when sending or retweeting tweets.

B. APPLYING OLAP TO TWITTER DATA

Online analytical processing, or OLAP, provides an intuitive form that is suitable for exploring Twitter data from multiple dimensions. As shown in Figure 2, from the perspective of the conceptual model of OLAP, the fact table ''UserFact'' includes measures such as ''FriendsCount'' and ''FollowerCount,'' which can be obtained directly by attribute mapping from the tweeter entity. Similarly, the dimension tables ''UserDIM,'' ''LocationDIM'' and ''TimeDIM'' can be obtained by attribute mapping from the tweeter entity. However, the tweets' topics, or the tweeter's interests, are implicitly embedded in the tweets. Such topics or interests establish a dimension hierarchy for OLAP, which must be extracted from the Twitter data.

OLAP provides users with operations such as the roll-up, drill-down, slicing and dicing operations which can analyze Twitter data from multiple perspectives. The overall process of exploring Twitter data based on the OLAP technique can be described as follows (Figure 3):

- **Data acquisition**: Obtain tweeters' profiles, tweets and social relationships through the REST APIs provided by Twitter.
- **Data preprocessing**: Remove the short words (the most common, short function words, such as *the*, *is*, *at*, *which*, and *on*) and the web links and carry out a parts of speech analysis to leave only nouns and verbs in the unstructured tweets.
- **Text modeling**: Identify the relationship between tweeters and tweets based on text modeling.

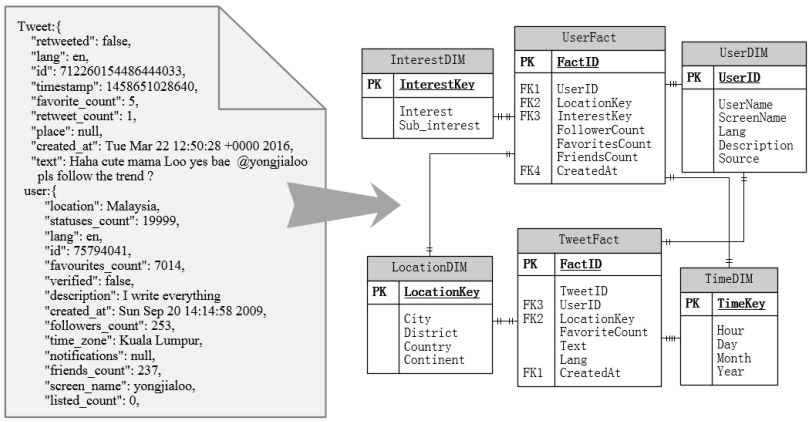


FIGURE 2. The galaxy schema for twitter data.

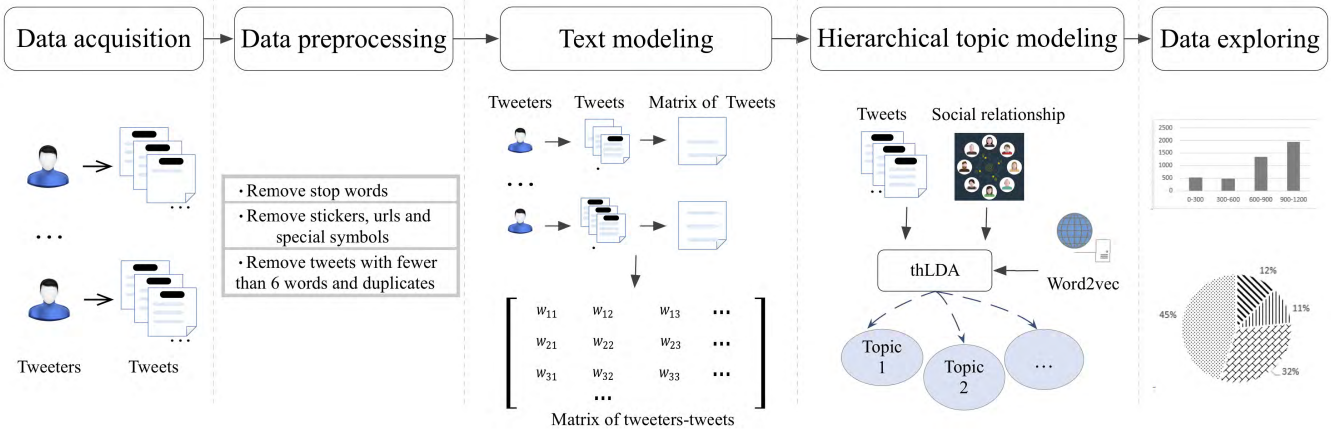


FIGURE 3. The overall process of exploring Twitter data based on the technique.

- **Hierarchical topic modeling**: Extract the topics (or interests) from the Twitter data, and construct the hierarchical topic dimension based on the probability distribution of various topics and subtopics.
- **Data exploring**: Analyze tweeters from multiple dimensions using OLAP.

Although the OLAP technology provides an intuitive inquiry form that is consistent with human custom, it can only handle structured data, and fails to deal with scenarios related to unstructured text data like tweets. Therefore, the key to applying the OLAP technology to Twitter data is to identify and construct the dimension hierarchy from the Twitter data automatically. However, this still remains a difficult problem. The main issue this paper tries to resolve can be described as follows: *how to automatically mine and construct the hierarchical dimension of tweets' topics (or tweeters' interests) from the unstructured tweet data to achieve effective multidimensional analysis.*

C. WORD2VEC

Word2vec [30], [31] is a group of related models that are used to produce word embeddings. These models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2vec takes as its input a large corpus of text and produces a vector space, typically of several hundred dimensions, with each unique word in the corpus being assigned a corresponding vector in the space. Word vectors are positioned in the vector space such that words that share common contexts in the corpus are located in close proximity to one another in the space. Word2vec was created by a team of researchers led by Tomas Mikolov at Google, and has been subsequently analyzed and explained by other researchers. Embedding vectors created using the Word2vec algorithm have many advantages compared to earlier algorithms such as latent semantic analysis. Word2vec can utilize either of two model architectures to produce a distributed representation of words: continuous bag-of-words (CBOW) or continuous skip-gram. In the continuous bag-of-words architecture, the model predicts the current word from a window of surrounding context words. However, the order of context words does not influence prediction (bag-of-words assumption). In the continuous skip-gram architecture, the model uses the current word to predict the surrounding window of context words.

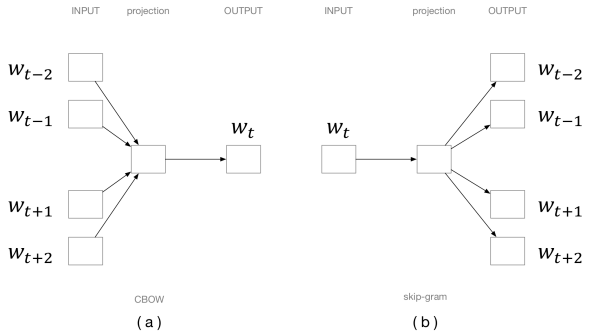


FIGURE 4. The models CBOW and skip-gram [30].

Figure 4(a) show the CBOW model, where w_t represents the central word and $w_{t\pm i}$ represents the context word of w_t. Figure 4(b) shows the skip-gram model, which weighs nearby context words more heavily than more distant context words. CBOW is considered to be faster than skip-gram but more suitable for infrequent words.

During the calculation process of word2vec, we usually express the semantic correlation between words by calculating the cosine similarity of two word vectors. The greater the cosine similarity, the stronger the correlation between two words. In addition, as the dimension increases, the model effectiveness tends to be steady. To ensure the high efficiency and good effectiveness, we choose 300 as the number of dimensions of the word vector in our approach.

D. CRP AND HLDA

The current topic models can be employed to mine the tweets' topics from large quantities of Twitter data. As a classical topic model, the standard LDA model considers that each word in an article is obtained by the following process: choose a topic with a certain probability in the article, and choose a word from the chosen topic. In the framework of the LDA model, all words in all articles represent observable data, and the topics of articles are implicit random variables which can only be obtained through a process of several iterations of sampling.

However, one of the disadvantages of the standard LDA model is that we must specify the number of topics in advance in the modeling process. In fact, the number of topics is unknown in different articles, and a fixed topic number may cause malign effects on the modeling process. In addition, the standard LDA model is unable to analyze the relationships between topics. In other words, by leveraging standard LDA, we can only retrieve topics in one single layer rather than in a topic hierarchy.

Fortunately, a probability distribution model based on the partition of integers, CRP (Chinese restaurant process) and its extension called nCRP (nested Chinese restaurant process), can organize topics into a hierarchical structure, and allow the data to continue to change and accumulate, by creating a hierarchical division of the sampling process.

CRP is a discrete-time stochastic process, analogous to seating customers at tables in a Chinese restaurant. It assumes that there is a Chinese restaurant which owns an unlimited number of infinite tables that can take an infinite number of customers at the same time. All customers come into the restaurant and choose their own tables with a certain probability. Here, the customers are regarded as an infinite collection, i.e., customer $= \{m|0 \leq m \leq N_M\}$. Given that the first $m-1$ customers have selected their tables, the collection of occupied tables is expressed as $R = \{r_j|0 < j < m\}$, and the corresponding number of customers at each table is $N = \{n_j|0 < j < m\}$. The probability that the next customer m chooses an occupied, or unoccupied table is given by the following distributions:

$$P(\textit{occupied table } r_j|\textit{Previous } m-1 \textit{ Customer}, \gamma) = \frac{n_j}{\gamma + m - 1} \quad (1)$$

$$P(\textit{unoccupied table}|\textit{Previous } m-1 \textit{ Customer}, \gamma) = \frac{\gamma}{\gamma + m - 1} \quad (2)$$

Here, γ is the parameter which aims to control the probability of the customer selecting a new table.

The nCRP model is derived from CRP, and is a distribution over hierarchical partitions. The nCRP model can be illustrated by the following situation. Supposing in a city there is an infinite number of Chinese restaurants, each of which has an infinite number of tables. The first restaurant is regarded as the root restaurant and each table in this restaurant corresponds to a card which refers to another restaurant. In the other words, each restaurant is associated with other restaurants. Consequently, all the restaurants can be organized into a tree with an infinite number of branches, while every level of the tree is associated with an infinite number of restaurants.

Consider a certain number of customers coming to the city for L days of holiday. On the first day, a customer comes into the root restaurant and chooses a table according to Equation (1). On the second day, he goes to the second restaurant which is associated with the table chosen previously by himself, and then chooses a table according to Equations (1) and (2). All customers choose restaurants according to Equations (1) and (2), repeatedly for L days. In other words, all customers follow a path which starts from the root restaurant and ends at level L. After all customers have finished their L-day holiday, the paths followed by each customer constitute a collection which can be regarded as an L-level tree. As an extension of CRP, nCRP can be applied to illustrate the uncertainty in the hierarchical structure (see Figure 5 for an example of such a tree)

The hLDA model mines the topics in the same way as LDA, but applies nCRP to organize the topics into a hierarchical structure rather than a flat structure. During the modeling process of hLDA, a certain document first chooses a path which starts from the root node and ends on a leaf node by nCRP, and then samples topics at every node in the chosen

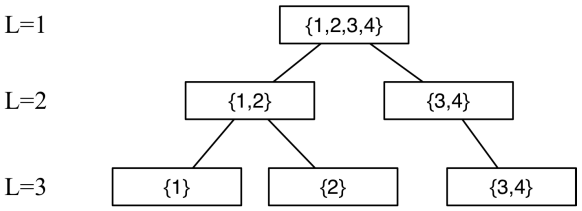


FIGURE 5. The paths of four tourists through the infinite tree of Chinese restaurants ($L = 3$).

path, and samples each word of the documents from the chosen topics. In this way, hLDA obtains a hierarchical structure whose every node is related to a topic and where each topic is regarded as a distribution of words after a certain number of iterations. Consequently, a topic hierarchy is obtained, which contains the underlying relationships between topics and simultaneously reflects the universality and specificity of the words.

Compared with the LDA model, hLDA generates a priori distribution of Bayesian non-parametric models through nCRP. In addition, the number of topics generated from hLDA is automatically changed according to changes in the corpus. Indeed, hLDA can adapt to the dynamic growth of the data set, and can distribute the topics into multiple abstraction levels. As a hierarchical topic model, hLDA is a pure data-driven approach that not only implements deep semantic analysis, but also identifies relationships between topics, namely, abstract and specific topics. In general, the topics that are close to the top are more abstract, whereas the topics that are close to the bottom are more specific. Consequently, the hierarchical organization of topics accords with human cognition of vocabulary and semantics.

IV. THLDA

A. OVERVIEW

In contrast with hLDA, thLDA integrates tweets and social relationships among tweeters into the modeling process. In addition, it considers semantic relationships between words in tweets. Figure 6 shows the Bayesian process of thLDA. During the modeling process, we first sample the path c_m for each tweeter, and then sample $z_{m,w}$ which denotes the topic allocation of each word associated with the level in the path.

Table 1 presents the symbols used throughout the paper. For simplicity, we do not distinguish between topics and interests in the Twitter data.

B. DATA PREPROCESSING

Before the actual topic modeling, we need to preprocess the text by transforming the disordered text into an easy-to-handle text-word matrix.

The traditional LDA and hLDA require documents with clear structure and rigorous style. Unfortunately, tweet texts

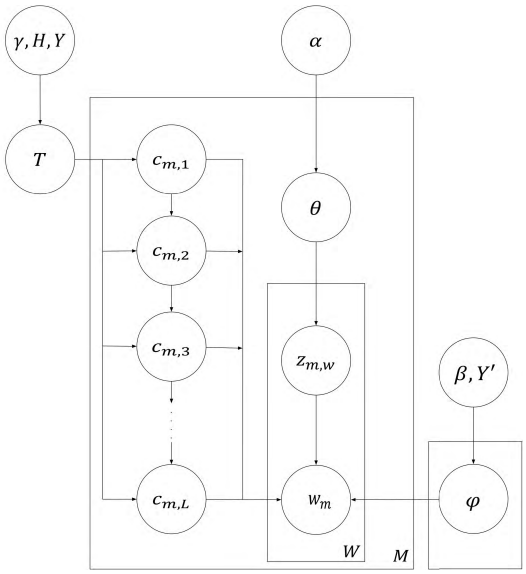


FIGURE 6. The graphical description of thLDA.

TABLE 1. Symbols used throughout the paper.

Symbol	Definition
M	collection of all tweeters
N_M	number of collection M
L	height of topic tree
D	collection of tweets of all tweeters
T	collection of paths drawn from nCRP
K	collection of topics acquired from the model
N_K	number of collection K
c	paths or topic collections of all tweeters
c_m	paths or topic collections of tweeter m
c_{-m}	collection of all tweeters' paths leaving out c_m
θ_m	distribution of topic over tweeter m
φ_k	distribution of words over topic k
W	collection of all words of all tweeters
W_m	collection of all words of tweeter m
α	vector of hyper parameter over θ_m
β	vector of hyper parameter over φ_k
γ	vector of hyper parameter over nCRP
$z_{m,w}$	topic assignment of word w of tweeter m
z_m	collection of $z_{m,w}$ for all words of tweeter m
z	collection of $z_{m,w}$ for all words of all tweeters
v_k	vector of words of topic k
v_m	vector of words of tweeter m
$w_{m,n}$	n^{th} word of tweeter m
$H_{m,k}$	social impact on tweeter m choosing topic k
$Y_{k_l,k_{l+1}}$	semantic impact of choosing topic k_l on choosing topic k_{l+1}
$Y'_{k,w}$	semantic impact of assigning word w to topic k

tend to be short and simple. When a single tweet text is treated as a document that is modeled as an input to LDA or hLDA, we often can not obtain good results. Therefore, in this paper, we treat all tweet texts of a Twitter user as the input document to thLDA.

As shown in Figure 7, we combine all the tweet data of the Twitter user $Twitter_m$ into a tweet document, and then obtain the tweet document collection $TDC = \{TweetDoc_m | m \in M\}$, in which $TweetDoc_m = \{w_{m,1}, w_{m,2}, \ldots, w_{m,n}\}$.

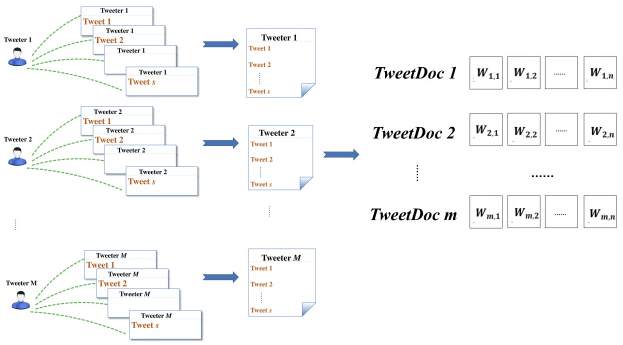


FIGURE 7. Schematic diagram of text modeling for twitter data.

C. THE THLDA MODEL

In the thLDA model, we need to sample two parameters via Gibbs sampling: the path c_m of each tweet document $TweetDoc_m$ in the topic tree and the topic number $z_{m,w}$ of all words in the tweet document collection. The joint probability distribution of the path c_m and the topic number $z_{m,w}$ is shown in Equation (3):

$$\begin{aligned} &P(c_m, z_{m,w}|\alpha, \beta, \gamma, Y, Y', H, W_m) \\ &\quad = P(c_m|W_m, c_{-m}, z, \gamma, \beta, Y, H) \\ &\qquad \times P(z_{m,w}|z_{-(m,w)}, W_m, Y', \alpha, \beta) \end{aligned} \tag{3}$$

Equation (3) describes the joint probability distribution between the observable words of tweeter and the latent topic in the thLDA model, in which γ is the hyperparameter of the nCRP model, and α, β are hyperparameters defined in the topic sampling process. During the path sampling process, γ is used to control the probability of each tweet document selection path; during the topic sampling process, β is used to control the probability of selecting a topic for each word. They are used together for the size of the subject tree. If γ is larger and β is smaller, more topics will be obtained, and a larger topic tree will eventually be generated. A smaller β value will result in fewer high-probability words for each topic, and more topics to describe the data. On the other hand, a larger γ will lead to a higher probability that the tweet document will select a new path.

We use a special MCMC (Markov Chain Monte Carlo) method to infer the posterior probability distribution of thLDA. Since it accepts all the data in the sample space, its acceptance rate achieves 100%. The MCMC needs to estimate multiple potential variables, but only considers a single latent variable in a single sample and treats the remaining variables as observable variables. When sampling the path c_m and the topic number $z_{m,w}$, the main work is as follows:

(1) According to $P(c_m|W_m, z, c_{-m}, \gamma, \beta, Y, H)$, randomly sample the state of c_m at the next moment c'_m.
(2) According to $P(z_{m,w}|z_{-(m,w)}, W_m, Y', \alpha, \beta)$, randomly sample the state of $z_{m,w}$ at the next moment $z'_{m,w}$.

In the following two sections, we detail the derivation of path sampling and topic sampling, respectively.

D. PATH SAMPLING

The distribution of path c_m which conditions on all observed words is expressed as follows:

$$\begin{aligned} P(c_m|W, c_{-m}, z, \gamma, \beta, Y, H) &= P(c_m|c_{-m}, \gamma, Y, H) \\ &\quad \times P(W_m|c, W_{-m}, z, \beta) \end{aligned} \tag{4}$$

According to Equation (4), two elements affect the probability that a tweeter selects a certain path. On the one hand, the first factor $P(c_m|c_{-m}, \gamma, Y, H)$ is implied by the nCRP model, with the extra consideration of social relationships and the semantic impact of words.

The generation process of nCRP model is described as follows:

(1) When the node $k_{l,j}$ of the topic tree has been selected, the probability that we choose a non-empty node $k_{l+1,j'}$ is defined as follows:

$$\begin{aligned} &P(c_m|c_{-m}, \gamma, Y_{k_l,k_l+1}, H) \\ &\qquad = \frac{N(k_{l+1,j'})}{\gamma + m - 1} \times Y_{k_{l,j},k_{l+1,j'}} \times H \end{aligned} \tag{5}$$

(2) When the node $k_{l,j}$ of the topic tree has been selected, the probability that we choose an empty node $k_{l+1,j'}$ is defined as follows:

$$P(c_m|c_{-m}, \gamma) = \frac{\gamma}{\gamma + m - 1} \tag{6}$$

Equations (5) and (6) describe the probability distribution of the tweet document $TweetDoc_m$ when selecting the next layer of nodes in the topic tree, where $N(k_{l+1,j'})$ represents the number of tweet documents selecting node $k_{l+1,j'}$. Each node in the topic tree consists primarily of two pieces of data: the topic and the tweet document which selects the node. In order to make full use of these two parts of data, we define H and $Y_{k_l,k_{l+1}}$ in the equations, as explained in the following.

During the process of sampling a path, a tweeter at a given level will choose a index which is related to the node at next level. As we know, each node at each level is associated with a topic. We hold the view that the semantic similarity of the topics affects the nCRP process. Therefore, $Y_{k_l,k_{l+1}}$ is introduced to indicate the *semantic impact* between two topics (or nodes) k_l and k_{l+1}. The higher the value of $Y_{k_l,k_{l+1}}$, the higher the probability that the topics k_l and k_{l+1} will be assigned to the sample path.

To calculate $Y_{k_l,k_{l+1}}$, we extract the top n words as $Q_{k_l} = \{q_{k_l,i}|1 \le i \le n\}$. We use $F_{k_l} = \{f_{k_l,i}|1 \le i \le n\}$ to represent the collection of their frequencies, where each item gives the number of occurrences of the corresponding word. Thus, $Y_{k_l,k_{l+1}}$ is calculated according to the following :

$$Y_{k_l,k_{l+1}} = \frac{\sum_{i=1}^{n} f_{k_l,i} \times \frac{\left(\sum_{j=1}^{n} f_{k_{l+1},j} \times sim(q_{k_l,i}, q_{k_{l+1},j})\right)}{\sum_{j=1}^{n} f_{k_{l+1},j}}}{\sum_{i=1}^{n} f_{k_l,i}} \tag{7}$$

Here, to calculate $\text{sim}(q_{k_{l,i}}, q_{k_{l+1,j}})$, we employ word2vec, an efficient tool for training words as an x-dimensional vector space. Supposing there are two words w_1 and w_2, we can obtain the similarity between w_1 and w_2 using the following expression:

$$\begin{aligned} Sim\,(w_1, w_2) &= cos\,(V_1, V_2) \\ &= \frac{\sum_{i=1}^{x} (V_{1,i} \times V_{2,i})}{\sqrt{\sum_{i=1}^{x} V_{1,i}^2} \times \sqrt{\sum_{i=1}^{x} V_{2,i}^2}} \end{aligned} \tag{8}$$

Here, V_1 and V_2 are the vectors of w_1 and w_2 obtained using word2vec, and x is the number of dimensions.

Further, we introduce $H_{m,k}$, or the *social impact*, to represent the degree to which social relationships affect tweeter m in choosing topic k.

Supposing $S_m = \{u_1, u_2, u_3, \cdots, u_{N_m}\}$ represents the social list of tweeter m where u_i represents the i^{th} tweeter in the social list S_m and N_m represents the number of all tweeters. The social impact is calculated using the following equation, where $P_{u_i,k}$ represents the probability that tweeter u_i selects topic k in the previous iteration:

$$H_{m,k} = \frac{\sum_{j=1}^{N_m} P_{u_i,k}}{N_m} \tag{9}$$

On the other hand, the second factor of Equation (4), or the probabilistic distribution $P\{W_m|c, W_{-m}, z, \beta\}$, represents the probability of obtaining the words for tweeter m with a certain choice of path, which can be calculated as follows:

$$\begin{aligned} P(W_m|c, W_{-m}, z, \beta) = &\prod_{l=1}^{L} \frac{\Gamma(\sum_{w\in W} (n^w_{c_{m,l},-m} + \beta))}{\prod_{w\in W} \Gamma(n^w_{c_{m,l},-m} + \beta)} \\ &\times \frac{\prod_{w\in W} \Gamma(n^w_{c_{m,l},-m} + n^w_{c_{m,l},m} + \beta)}{\Gamma(\sum_{w\in W} (n^w_{c_{m,l},-m} + n^w_{c_{m,l},m} + \beta))} \end{aligned} \tag{10}$$

where $n^w_{c_{m,l},-m}$ represents the number of words assigned to $c_{m,l}$, excluding those in the tweet document $TweetDoc_m$.

E. TOPICS SAMPLING

After path sampling, we sample the words of each tweeter, i.e., allocate the topic, or the level of the topic tree, to each word.

The joint probability of the whole corpus of tweets is calculated as follows:

$$\begin{aligned} P\left(z_{m,w}\middle|z_{-(m,w)}, W_m, Y', \alpha, \beta\right) &= p(W_m|z, \beta, Y') \times p(z|\alpha) \\ &= p\left(W_m, z\middle|\alpha, \beta, Y'\right) \end{aligned} \tag{11}$$

Here, we need to utilize the collapsed Gibbs sampling to sample the variables W_m and z. The main sampling steps are described as follows:

(1) Initialization. We assign a topic to each word according to the multinomial distribution.
(2) Sampling. For each word, we utilize collapsed Gibbs sampling to assign a topic to each word according to the semantic relationship between word and topic and the Dirichlet distribution between word and topic.
(3) Iteration. Repeat Step (2) until the result converges to a steady value.

We assume that N_M tweeters are associated with N_M independent Dirichlet-multinomial conjugated structures and N_K topics are associated with N_K independent Dirichlet-multinomial conjugated structures also. The main process of assigning a topic to each tweet word of tweeter m is presented as follows:

(1) $\alpha \rightarrow \theta_m \rightarrow z$: When generating the tweet of tweeter m, we first obtain θ_m which is the probability distribution of topics over tweeter m according to the hyper-parameter α. Afterwards, we generate z_m, the collection of $z_{m,w}$ for all words of tweeter m. Here, $\alpha \rightarrow \theta_m$ is associated with a Dirichlet process, and $\theta_m \rightarrow z$ is associated with a multinomial distribution. On the whole, $\alpha \rightarrow \theta_m \rightarrow z$ is associated with a Dirichlet-multinomial conjugated structure.
(2) $\beta, Y'_{k,w}, z_m \rightarrow \varphi_k \rightarrow W_m$: Given z_m, we first obtain k which is the probability distribution of words over topic k according to the hyper-parameter β and the semantic impact between topic k and word w. Afterwards, we generate W_m, the collection of all words of tweeter m. Here, $\beta, Y'_{k,w}, z_m \rightarrow \varphi_k$ is associated with a Dirichlet process, and $\varphi_k \rightarrow W_m$ is associated with a multinomial distribution. As a whole, $\beta, Y'_{k,w}, z_m \rightarrow \varphi_k \rightarrow W_m$ is associated with a Dirichlet-multinomial conjugated structure.

We obtain the probability distribution of topics as follows:

$$\begin{aligned} p\,(z|\alpha) &= \int p(z|\theta) \times p(\theta|\alpha) d\theta \\ &= \prod_{m\in M} \frac{(v_m + \alpha)}{(\alpha)} \\ &= \frac{\Gamma(\sum_{k\in K} \alpha_k)}{\prod_{k\in K} \Gamma(\alpha_k)} \prod_{m\in M} \frac{\prod_{k\in K} \Gamma(v_{m,k} + \alpha_k)}{\Gamma(\sum_{k\in K} (v_{m,k} + \alpha_k))} \end{aligned} \tag{12}$$

Furthermore, the probability distribution of words is obtained as follows:

$$\begin{aligned} &p\left(W_m\middle|\beta, Y'_{k,w}, z\right) \\ &= \int p(W_m|z, Y'_{k,w}, \varphi) \times p(\varphi|\beta) d\varphi \\ &= \frac{\prod_{k\in K} (Y'_{k,w} (v_k + \beta))}{(\beta)} \\ &= \frac{\Gamma(\sum_{w\in W_m} \beta_w)}{\prod_{w\in W_m} \Gamma(\beta_w)} \prod_{k\in K} \frac{\prod_{w\in W_m} \Gamma(Y'_{k,w} \times (v_{k,w} + \beta_w))}{\Gamma(\sum_{w\in W_m} Y'_{k,w} (v_{k,w} + \beta_w))} \end{aligned} \tag{13}$$

We hold the view that the semantic similarity of the words and topics influences the topic sampling process. The higher the semantic similarity of the words and topics, the greater the probability that the words will be assigned to the topic. We use $Y'_{k,v}$ to represent the degree of word-topic semantic impact of word v belonging to topic k. To calculate the word-topic semantic impact, we pick out the top n words which belong to topic k to constitute a collection

$Q_k = \{q_{k,i} | 1 \leq i \leq n\}$. The word-topic semantic impact can thus be obtained as follows:

$$Y'_{k,w} = \frac{\sum_{i=1}^{n} (f_{k,i} \times Sim(w, q_{k,i}))}{\sum_{i=1}^{n} f_{k,i}} \tag{14}$$

Here, $f_{k,i}$ denotes the frequency of occurrence of word i with respect to topic k.

According to Equations (12) and (13), we obtain the joint probability distribution of W and z as fallows:

$$p\left(W_m, z | \alpha, \beta, Y'_{k,w}\right) = \frac{\prod_{k \in K} \left(Y'_{k,w} \left(v_k + \beta\right)\right)}{(\beta)} \times \prod_{m \in M} \frac{(v_m + \alpha)}{(\alpha)} \tag{15}$$

According to the Gibbs sampling method, we iterate over Equation (15) and sample the topic of all words until the sampling result becomes stable. Finally, we obtain the probability distribution θ_m of document-topic of the tweet and the probability distribution φ_k of topic-word of the tweet. The results are as follows:

$$\theta_m = \frac{(v_{m,k} + \alpha)}{(v_{m,\cdot} + K\alpha)} \tag{16}$$

$$varphi_k = \frac{Y'_{k,w}(v_{k,w} + \beta)}{Y'_{k,\cdot}(v_{k,\cdot} + V\beta)} \tag{17}$$

Combining c_m, θ_m, and φ_m, we know the distribution of the various themes of *TweetDoc* in the path c, and the distribution probability of various words of *TweetDoc* in the topic. In this way, we obtain a complete topic tree. Algorithm 1 describes the formal modeling process for thLDA.

Algorithm 1 Formalized Modeling Process of thLDA

Input: *TDC* - The set of Twitter document;
α, β, γ - hyperparameters;
L - the height of topic tree;
I - the iteration number of Gibbs sampling;
Output: *TopicTree*;

// Associate topic with node based on Dirichlet dist
for each $t \in TopicTree$ **do**
  draw a Dirichlet Process $\varphi \sim Dir(\beta)$;
end for
// Generate a path for $TweetDoc_m$ based on nCRP
for each $TweetDoc_m \in TDC$ **do**
  let c_1 be the root node;
  for each level $l \in 1, 2, \ldots, L$ **do**
    draw the current level for each $Tweet_{m,s}$;
    draw a occupied path c_l using Eq. (5);
    draw a unoccupied path c_l using Eq. (6);
  end for
  obtain c_m;
  draw a L-dim. topic proportion vector θ_m from $Dir(\alpha)$;
  for $i = 1$ to I **do**
    for each word $w \in W$ **do**
      draw topic $z \in 1, 2, \ldots, L$ from $Mult(\theta)$;
      draw w from the topic z;
    end for
  end for
end for
return *TopicTree*;

V. EXPERIMENT

A. DATA AND ENVIRONMENT

To verify the effectiveness and efficiency of our model, we conducted extensive experiments on large quantities of Twitter data collected through the Twitter REST API. We first chose 15 Twitter users with the largest amount of attention as the seeds and then obtained all tweeters who followed the seeds, retrieving their profiles, tweets, and social relationships (including following lists and followed lists). The number of tweets reached a total of 21,213,000. Subsequently, we removed the short tweets of less than 6 words, because we think such tweets generally have no clear semantics. In addition, we also removed the duplicated tweets. Finally, we obtained 10,160,317 tweets from 6,907 tweeters. Figure 8 shows the distribution of tweeters and tweets. Due to the limitations of the Twitter REST API, we could only acquire at most 3,200 tweets for each tweeter. The experimental data and results are published on the website for reference (http://dbsi.hdu.edu.cn/twitter_data/).

The word2vec model we employed in our paper was downloaded from https://code.google.com/archive/p/word2vec/. This repository hosts the word2vec model (three million 300-dimensional English word vectors) trained on the Google

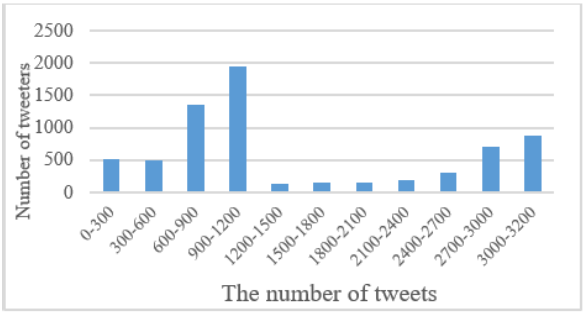


FIGURE 8. The distribution of tweeters over different numbers of tweets.

News dataset. The experiments were executed on a computer with eight E5-2620 2.10GHz cores, 16GB memory, and Windows 7.

B. EVALUATING EFFECTIVENESS BASED ON PMI

We used PMI-score (Pointwise Mutual Information score) to evaluate the effectiveness of our model. To check whether a topic was reasonable, we judged the number of odd words which were irrelevant to the specific topic.

We calculated the PMI values for pairs of the top 20 frequent words relevant to topic. The larger the PMI value between two words, the stronger the relationship between them. If two words are completely unrelated, their PMI value

TABLE 2. The perplexity of thLDA, LDA, and hLDA over different heights of topics with different numbers of iterations.

	thLDA					LDA					hLDA				
Iterations	H=2	H=3	H=4	H=5	H=6	K=14	K=26	K=31	K=45	K=55	H=2	H=3	H=4	H=5	H=6
0	4,293	3,974	4,227	4,106	5,008	6,877	7,470	7,721	8,291	8,906	4,159	3,906	4,293	4,556	4,956
50	3,533	2,981	3,055	3,055	3,408	6,672	3,246	3,414	7,659	3,433	3,601	3,047	3,099	3,198	3,459
100	3,518	2,972	3,026	3,026	3,388	6,429	3,277	3,423	6,411	3,439	3,597	3,041	3,087	3,179	3,477
150	3,418	2,979	3,031	3,031	3,423	3,499	3,302	3,429	3,369	3,444	3,597	3,039	3,088	3,179	3,536
200	3,418	2,970	3,028	3,028	3,412	3,508	3,313	3,420	3,390	3,442	3,597	3,036	3,084	3,178	3,565
250	**3,418**	**2,968**	**3,026**	**3,026**	**3,425**	3,505	3,321	3,417	3,390	3,442	3,599	3,038	3,088	3,182	3,534

Note: bold type denotes the lowest value of perplexity.

is set to zero. We set the PMI-score of topic k to the median value of all the PMI values of its word pairs, as shown in Equation (18).

$$PMI - SCORE^k = median\{PMI(w_i^k, w_j^k)\} \quad i, j \in [1, 20] \tag{18}$$

in which

$$PMI(w_i^k, w_j^k) = \log \frac{p(w_i^k, w_j^k)}{p(w_i^k)p(w_j^k)} \tag{19}$$

As we know, when applying LDA the number of topics must be assigned in advance. However, the number of topics can be determined during the modeling process when applying either our model or hLDA. To ensure a fair comparison, we first conducted the experiments on thLDA and obtained the specific topic number for different heights of the topic trees, and based on these we then ran the experiments on LDA. The relation between the heights of topic trees (used by thLDA) and the corresponding topic number (used by LDA) is shown in Table 2.

TABLE 3. The height of topic tree and its corresponding topic number.

Height of topic tree (H)	Corresponding topic number (K)
2	14
3	26
4	31
5	45
6	55

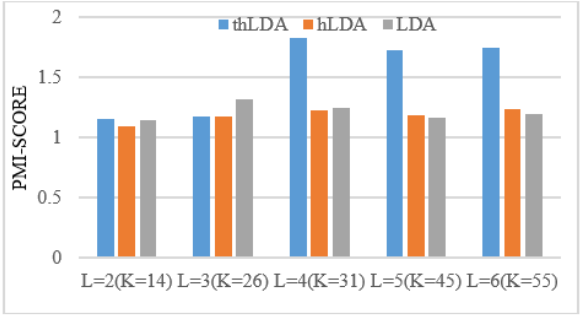


FIGURE 9. The comparison of PMI-score of thLDA, hLDA and LDA over different heights and topic numbers.

As Figure 9 shows, the PMI score of our model is slightly higher than those of the other two models for a height value of two, and the PMI score of our model is slightly lower than those of the other two models for a height value of three, when the height is too small, the corresponding number of topics will be small, and unrelated words will be assigned to the same topic, consequently, the PMI score of our model is similar to that of the other two models a height value of two and three. However, our model outperforms than other two models for height values four, five and six.

C. EVALUATING EFFECTIVENESS BASED ON PERPLEXITY

As a conventional evaluation index of topic models, *perplexity* is normally used to evaluate the ability of a topic model for generating texts. For a set of tweets, a lower *perplexity* denotes better effectiveness of the topic model and a stronger ability for predicting texts. For a set of tweets D, the *Perplexity* is calculated as follows:

$$P(D) = exp^{-\frac{\sum_{m=1}^{M} log(p(w_m))}{\sum_{m=1}^{M} N_m}} \tag{20}$$

in which w_m denotes the word of tweeter m and N_m denotes the number of words of tweeter m, respectively, and M denotes the number of tweets in the set D.

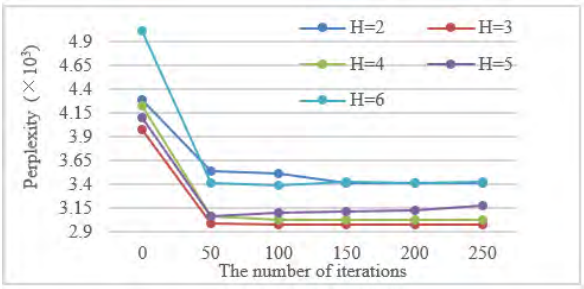


FIGURE 10. The perplexity of thLDA over different heights with different numbers of iterations.

Figure 10 shows that the *perplexities* of thLDA over all heights decrease with an increasing number of iterations and eventually converge to the steady values. Table 3 compares the *perplexities* of thLDA with those of LDA and hLDA. It is clearly seen that in all cases the *perplexities* of thLDA are lower than those of LDA when the modeling process becomes stable. Similarly, in most cases (H=3, 4, 6), the *perplexities* of thLDA are clearly lower than those of hLDA when the process becomes stable. Overall, the experiment

TABLE 4. Part of distribution of topic numbers at different levels over different heights for thLDA.

Topic	Parent-Topic	Level	Words
Topic-1 of Level-1	N/A	Level-1	People, School, Life, Read, News, Women, Watch, Live, Story, Change...
Topic-1 of Level-2	Topic-1 of Level-1	Level-2	Science, Physics, Live, Learn, Space, Students, Earth, Energy, Video, Watch...
Topic-2 of Level-2			Steps, Traveled, Miles, Ballgame, Royals, City, Fitness, Fortress, Solitude, Sporting...
Topic-1 of Level-3	Topic-1 of Level-2	Level-3	Game, Team, Watch, USA, Live, Win, Check, Play, Set, volleyball...
Topic-2 of Level-3			Job, Jobs, Theaters, Wage, Check, Employment, Labor, Lol, Workers, Interview...
Topic-3 of Level-3			Cell, Issue, Technology, Online, Cells, Stem, Top, Content, Stories, Science...
Topic-4 of Level-3			Brexit, Stocks, Oil, Market, Markets, Investors, Bank, China, CEO, Fed...
Topic-5 of Level-3			Volleyball, Support, Dot, Game, Ball, Set, Play, Life, Team, Park...
Topic-6 of Level-3	Topic-2 of Level-2		Game, Report, Score, Deal, Curry, Win, Play, Knee, Draft, Warriors, Dose, Pick...
Topic-1 of Level-4	Topic-1 of Level-3	Level-4	History, Soldiers, Pope, Trump, War, ISIS, Army, Muslim, Religion, Francis...
Topic-2 of Level-4			Euro, Goal, Wales, United, France, Cristiano, Manchester, Football, Player, Transfer...
Topic-3 of Level-4			Food, Recipe, Cake, Recipes, Chocolate, Chicken, Cream, Cheese, Dinner, Eat...
Topic-4 of Level-4			Photo, Climbing, Photos, Photography, climb, Shot, Shots, Video, Submit, Check...
Topic-8 of Level-4	Topic-2 of Level-3		Trump, Clinton, President, Sanders, Campaign, Email, House, Gop, Police, Lynch...
Topic-9 of Level-4			Music, Watch, Album, Listen, Live, Video, Song, Songs, Playlist, Top...
Topic-10 of Level-4			Travel, Trip, Visit, Summer, Beach, Park, Top, Flight, Vacation, Hotel...
Topic-11 of Level-4	Topic-3 of Level-3		Energy, Gas, Oil, Climate, Power, Blog, Industry, Future, Production, Emissions...
Topic-13 of Level-4	Topic-4 of Level-3		Fed, Trump, Vote, Orlando, China, Brexit, Clinton, Economy, Referendum, Shoot...
Topic-14 of Level-4			Book, Books, Reading, Read, Author, Writing, Writers, Fiction, Life, Story...
Topic-15 of Level-4	Topic-5 of Level-3		Fishing, Captain, Fish, Bass, Report, Lake, China, Boat, Catch, Sea...
Topic-16 of Level-4			University, Study, Source, Read, Psychology, People, Health, Brain, Life, Children...
Topic-20 of Level-4	Topic-6 of Level-3		Game, Stat, Sheet, Win, Season, Star, Baseball, Live, Team, Hit...

demonstrates that thLDA outperforms LDA and hLDA as far as *perplexity* is concerned.

D. OVERALL EFFECT

Table 4 shows part of word distribution of the discovered topics and the hierarchical relationships between them over different levels when the height is set to four.

One advantage of applying OLAP to Twitter data is that we can conduct multi-dimensional analysis using operations such as rolling up and drilling down. As shown in Figure 11, with regard to topic-1 of level-2, when we drill down into it, the distributions of the tweets' topics are different in different cities. However, in all cases, topic-1 of level-3, which may be described as "sports" in accordance with the hot words given in Table 4, attracts the most attention.

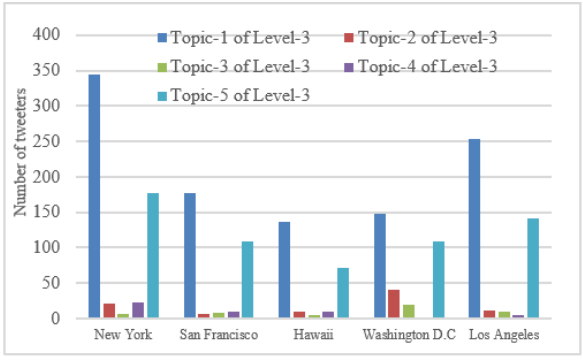


FIGURE 11. The distribution of child-topics of topic-1 of level-2 over different cities.

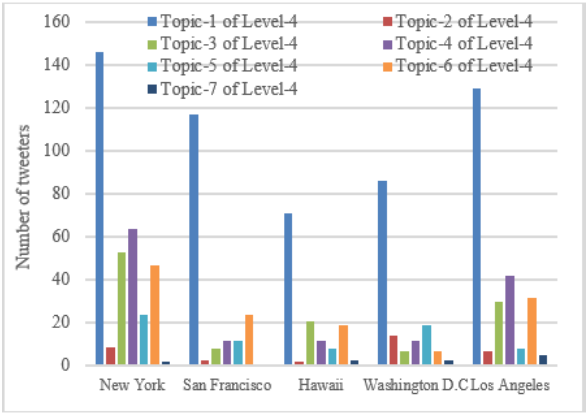


FIGURE 12. The distribution of child-topics of topic-1 of level-3 over different cities.

Figure 12 shows the results when we drill down into the topic-1 of level-3, whereas Figure 13 shows the results when aggregating the number of tweeters by rolling up the "location" dimension from city to country. It indicates that in

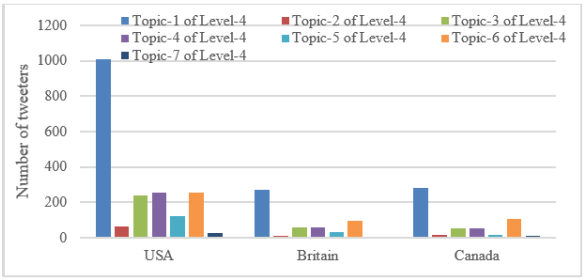


FIGURE 13. The distribution of child-topics of topic-1 of level-3 over different countries.

most cases tweeters in the "USA" are more active than other countries' tweeters.

VI. CONCLUSIONS AND FUTURE WORK

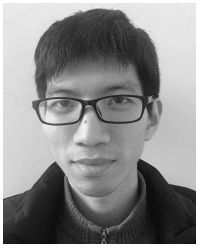
In this paper, we put forward a novel hierarchical topic model, i.e., thLDA, which is applied to mine the dimension hierarchy of tweets' topics from a large quantity amount of unstructured Twitter data. We conducted extensive experiments on real Twitter data to evaluate the effectiveness of thLDA. The results show that thLDA has a better recognition effect than the other models.

When considering how social relationships impact on the hierarchical topic model, we focus only on direct social relationships and ignore indirect relationships. Furthermore, we ignore cases where two unrelated tweeters follow the same tweeters. In the future, we will analyze indirect social relationships among tweeters to enhance our current model. In addition, to improve the model effectiveness, we will consider taking advantage of bicliques to calculate the semantic impact of the topic of two tweets. Last but not least, we will focus on how the social impact factors and word semantic similarity influence the experimental results separately, and whether it is possible to improve the model using hashtags.

REFERENCES

[1] D. Yu *et al.*, "Mining hidden interests from Twitter based on word similarity and social relationship for OLAP," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 27, nos. 9–10, pp. 1567–1578, 2017.
[2] D. Yu, J. Sun, Y. Wu, Z. Ni, and Y. Li, "Discovering hidden interests from Twitter for multidimensional analysis," in *Proc. 29th Int. Conf. Softw. Eng. Knowl. Eng.*, 2017, pp. 329–334.
[3] S. Chaudhuri and U. Dayal, "An overview of data warehousing and OLAP technology," *ACM SIGMOD Rec.*, vol. 26, no. 1, pp. 65–74, 1997.
[4] A. Inokuchi and K. Takeda, "A method for online analytical processing of text data," in *Proc. 16th ACM Conf. Conf. Inf. Knowl. Manage.*, 2007, pp. 455–464.
[5] F. Ravat, O. Teste, R. Tournier, and G. Zurfluh, "Top_keyword: An aggregation function for textual document OLAP," in *Proc. Int. Conf. Data Warehousing Knowl. Discovery*. Berlin, Germany: Springer, 2008, pp. 55–64.
[6] C. X. Lin, B. Ding, J. Han, F. Zhu, and B. Zhao, "Text cube: Computing IR measures for multidimensional text database analysis," in *Proc. 8th IEEE Int. Conf. Data Mining (ICDM)*, Dec. 2008, pp. 905–910.
[7] D. Zhang, C. Zhai, J. Han, A. Srivastava, and N. Oza, "Topic modeling for olap on multidimensional text databases: Topic cube and its applications," *Stat. Anal. Data Mining, ASA Data Sci. J.*, vol. 2, nos. 5–6, pp. 378–395, 2009.
[8] M. Azabou, K. Khrouf, J. Feki, C. Soulé-Dupuy, and N. Vallès, "A novel multidimensional model for the OLAP on documents: Modeling, generation and implementation," in *Proc. Int. Conf. Model Data Eng.* Cham, Switzerland: Springer, 2014, pp. 258–272.
[9] M. Michelson and S. A. Macskassy, "Discovering users' topics of interest on Twitter: A first look," in *Proc. ACM 4th workshop Anal. Noisy Unstructured Text Data*, 2010, pp. 73–80.
[10] A. Cuzzocrea, C. De Maio, G. Fenza, V. Loia, and M. Parente, "OLAP analysis of multidimensional tweet streams for supporting advanced analytics," in *Proc. 31st Annu. ACM Symp. Appl. Comput.*, 2016, pp. 992–999.
[11] X. Liu *et al.*, "A text cube approach to human, social and cultural behavior in the Twitter stream," in *Proc. Int. Conf. Social Comput., Behav.-Cultural Modeling, Predict.* Berlin, Germany: Springer, 2013, pp. 321–330.
[12] N. U. Rehman, A. Weiler, and M. H. Scholl, "OLAPing social media: The case of Twitter," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2013, pp. 1139–1146.
[13] E. Siswanto, M. L. Khodra, and L. J. E. Dewi, "Prediction of interest for dynamic profile of Twitter user," in *Proc. IEEE Int. Conf. Adv. Inform., Concept, Theory Appl. (ICAICTA)*, Aug. 2014, pp. 266–271.
[14] M. Pennacchiotti and A.-M. Popescu, "A machine learning approach to Twitter user classification," in *Proc. ICWSM*, 2011, vol. 11, no. 1, pp. 281–288.
[15] X. Pu, M. A. Chatti, H. Thues, and U. Schroeder, "Wiki-LDA: A mixed-method approach for effective interest mining on Twitter data," in *Proc. 8th Int. Conf. Comput. Supported Edu.* Rome, Italy: ScitePress-Science and Technology Publications, 2016, pp. 426–433.
[16] E. Vathi, G. Siolas, and A. Stafylopatis, "Mining interesting topics in Twitter communities," in *Computational Collective Intelligence*. Cham, Switzerland: Springer, 2015, pp. 123–132.
[17] W. X. Zhao *et al.*, "Comparing Twitter and traditional media using topic models," in *Proc. Eur. Conf. Inf. Retr.* Berlin, Germany: Springer, 2011, pp. 338–349.
[18] D. M. Blei and J. D. Mcauliffe, "Supervised topic models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 3, 2010, pp. 327–332.
[19] T. L. Griffiths, M. I. Jordan, J. B. Tenenbaum, and D. M. Blei, "Hierarchical topic models and the nested chinese restaurant process," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 17–24.
[20] D. M. Blei, T. L. Griffiths, and M. I. Jordan, "The nested chinese restaurant process and Bayesian nonparametric inference of topic hierarchies," *J. ACM*, vol. 57, no. 2, 2010, Art. no. 3.
[21] X.-L. Mao, Z.-Y. Ming, T.-S. Chua, S. Li, H. Yan, and X. Li, "SSHLDA: A semi-supervised hierarchical topic model," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.* Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 800–809.
[22] W. Wang, H. Xu, W. Yang, and X. Huang, "Constrained-hLDA for topic discovery in chinese microblogs," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*. Cham, Switzerland: Springer, 2014, pp. 608–619.
[23] A. M. Dai and A. J. Storkey, "The supervised hierarchical Dirichlet process," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 2, pp. 243–255, Feb. 2015.
[24] J.-T. Chien, "Hierarchical Pitman–Yor–Dirichlet language model," *IEEE/ACM Trans. Audio, Speech, Language Process*, vol. 23, no. 8, pp. 1259–1272, Aug. 2015.
[25] Y. W. Teh, "A hierarchical Bayesian language model based on Pitman–Yor processes," in *Proc. 21st Int. Conf. Comput. Linguistics 44th Annu. Meeting Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006, pp. 985–992.
[26] Q. Li, S. Shah, X. Liu, A. Nourbakhsh, and R. Fang, "Tweet topic classification using distributed language representations," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. (WI)*, Oct. 2016, pp. 81–88.
[27] D. Ganguly, D. Roy, M. Mitra, and G. J. F. Jones, "Word embedding based generalized language model for information retrieval," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2015, pp. 795–798.
[28] F. Enríquez, J. A. Troyano, and T. López-Solaz, "An approach to the use of word embeddings in an opinion classification task," *Expert Syst. Appl.*, vol. 66, pp. 1–6, Dec. 2016.
[29] D. Zhang, H. Xu, Z. Su, and Y. Xu, "Chinese comments sentiment classification based on word2vec and SVM^{perf}," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 1857–1863, 2015.
[30] T. Mikolov, K. Chen, G. Corrado, and J. Dean. (2013). "Efficient estimation of word representations in vector space." [Online]. Available: https://arxiv.org/abs/1301.3781
[31] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

DONGJIN YU is currently a Professor with Hangzhou Dianzi University, China, where he is also the Director of the Institute of Big Data and the Institute of Computer Software. His research efforts include big data, business process management, and software engineering. He is a member of IEEE, a member of ACM, and a Senior Member of China Computer Federation (CCF). He is also a member of the Technical Committee of Software Engineering, CCF, and the Technical Committee of Service Computing, CCF.



DENGWEI XU is currently pursuing the degree with Hangzhou Dianzi University, China. His research interests include machine learning and information retrieval.

DONGJING WANG received the B.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 2012 and 2018, respectively. He is currently a Lecturer with Hangzhou Dianzi University, China. His current research interests include recommender systems, machine learning, and business process management.

ZHIYONG NI received the bachelor's and master's degrees in computer science from Hangzhou Dianzi University, China. He has participated in several government funded projects related with data mining. His current research interests mainly include online analytic processing and information retrieval.