2014 IEEE International Conference on Data Mining

Hashtag Graph based Topic Model for Tweet Mining

Yuan Wang*, Jie Liu*, Jishi Qu†, Yalou Huang†, Jimeng Chen* and Xia Feng‡
*College of Computer and Control Engineering, Nankai University, Tianjin, China
Email: {yayaniuzi23@mail., jliu@, jeanchen@mail.}nankai.edu.cn
†College of Software, Nankai University, Tianjin, China
Email: {2120120447@mail., huangyl@}nankai.edu.cn
‡Information Technology Research Base of Civil Aviation Administration of China, Civil Aviation University of China, Tianjin, China
Email: xfeng@cauc.edu.cn

***Abstract*—Mining topics in Twitter is increasingly attracting more attention. However, the shortness and informality of tweets leads to extreme sparse vector representation with a large vocabulary, which makes the conventional topic models (e.g., Latent Dirichlet Allocation) often fail to achieve high quality underlying topics. Luckily, tweets always show up with rich user-generated hashtags as keywords. In this paper, we propose a novel topic model to handle such semi-structured tweets, denoted as *Hashtag Graph based Topic Model* (*HGTM*). By utilizing relation information between hashtags in our hashtag graph, HGTM establishes word semantic relations, even if they haven't co-occurred within a specific tweet. In addition, we enhance the dependencies of both multiple words and hashtags via latent variables (topics) modeled by HGTM. We illustrate that the user-contributed hashtags could serve as weakly-supervised information for topic modeling, and hashtag relation could reveal the semantic relation between tweets. Experiments on a real-world twitter data set show that our model provides an effective solution to discover more distinct and coherent topics than the state-of-the-art baselines and has a strong ability to control sparseness and noise in tweets.**

I. Introduction

Microblogging platforms, such as Twitter, have gone global. Bursts of world news, entertainment gossip, and discussions over recently released products are all collected in Twitter. It has been well recognized that uncovering topics of these user-generated content is crucial for a wide range of content analysis tasks[1][2][3]. Meanwhile, achieving good representation of document content could benefit tasks as organizing, classifying, or searching a collection of documents. In recent years, topic models, such as Probabilistic Latent Semantic Indexing (PLSI)[4] and Latent Dirichlet Allocation (LDA)[5], have been recognized as a powerful method of learning topic semantic representation of document corpus.

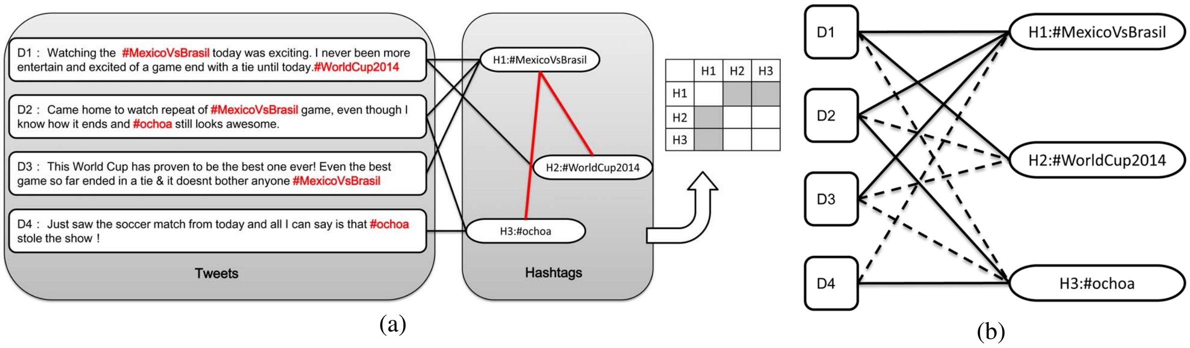
Although PLSI and LDA have achieved success in uncovering topic for normal documents, the characteristics of tweets pose new challenges. Firstly, the severe sparsity problem of tweet corpus invalidates traditional topic modeling techniques. Secondly, conventional LDA is designed for flat texts without structure, while hashtags, shown as "#hashtag", are a community-driven convention making tweets as semi-structured texts. Last but not least, such crowd wisdom information clashes with the assumption of independent identical distribution (i.i.d) of documents, because weakly-supervised information provided by hashtags builds semantic relations between tweets, so that tweets have more complex topical relationships than plain texts[2][6]. Hence, the i.i.d. assumption does not hold any more.

Therefore, it is crucial to consider the semantic and structured context information conveyed by hashtags in addition to bag-of-words methods within a tweet. Hashtags meet user requirements on Twitter, from associating tweets with particular events (e.g., "#WorldCup" and #Superbowl) to sharing memes (e.g., #bestsportsrivalry and "#ifyouknowmeyouknow").

Since hashtags are topic indicators generated by users, the tweets sharing the same hashtags have highly overlapping underlying topics. For example, as shown in Figure 1(a), users involve in the discussion of topic "World Cup 2014" by adding the hashtags "#MexicoVsBrasil" in tweet D1, D2 and D3 respectively, which bridges tweets with semantic relationship by the same hashtag. Furthermore, a number of hashtag co-occurrences in tweets indirectly contribute semantic relationships between tweets. In Figure 1(a), "#MexicoVsBrasil" and "#ochoa" ("#WorldCup2014") co-occur in a single tweet D2 (D1), such explicit co-occurrence indicates similar or related topics on tweets containing one of the two hashtags. Obviously, the word "soccer" in tweet D4 and the word "tie" in tweet D1 are semantically related. Unluckily, methods considering a single tweet or aggregated documents[2][6] couldn't model such a semantic relationship. While we can connect these two words through the path "D4"-"#ochoa"-"#MexicoVsBrasil"-"D1" based on the hashtag co-occurrences in the whole dataset, as shown in Figure 1(a). That means D4 should have a potential relationship with "#MexicoVsBrasil" (in a dotted link, as Figure 1(b) shows). These connections tackle the spareness problem in tweets as weakly-supervised information, and make a meaningful semantic relation between words.

Inspired by the observation mentioned above, it is vitally important to explore how to utilize the noisy wisdom of crowds in hashtag usage to face challenges in tweet topic modeling. In this paper, we propose a novel framework of Hashtag Graph based Topic Model (HGTM), which is contributed in a crowdsourcing manner and acquired with little cost. The basic idea of HGTM is to project tweets into a coherence semantic space by using latent features via user-contributed hashtags. HGTM is a probability generative model that incorporates weakly-supervised information based on a weighted hashtag graph. The model links tweets via both *explicit hashtags* from explicit tweet-hashtag relationships and *potential hashtags* from potential tweet-hashtag relationships, achieving in connecting semantic words that even haven't appeared in the

1550-4786/14 $31.00 © 2014 IEEE
DOI 10.1109/ICDM.2014.60



D1 : Watching the #MexicoVsBrasil today was exciting. I never been more entertain and excited of a game end with a tie until today.#WorldCup2014

D2 : Came home to watch repeat of #MexicoVsBrasil game, even though I know how it ends and #ochoa still looks awesome.

D3 : This World Cup has proven to be the best one ever! Even the best game so far ended in a tie & it doesnt bother anyone #MexicoVsBrasil

D4 : Just saw the soccer match from today and all I can say is that #ochoa stole the show !

	H1	H2	H3
H1		▨	▨
H2	▨		
H3	▨		

(a)

(b)

Fig. 1. An illustration of semantic relationship in tweets. (a) Explicit Relationships. One is inclusion relation between tweets and hashtags (marked with black links), the other one is co-occurrence relation between hashtags (marked with red links). The hashtag relationship can be formulated as a relation graph represented by a matrix. (b) Potential Relationships. Dotted links indicate the potential inclusion relation between tweets and hashtags.

same tweet. In such a way, we alleviate the sparse and noise problem in short texts, and also get coherent topic distributions of hashtags. Our contributions are summarized as follows.

- With HGTM, we alleviate the sparse problem in mining tweets and exploit the wisdom of crowds conveyed by hashtags as weakly-supervised information.
- In HGTM, we exploit the relations among hashtags, and establish a mathematical model of these relations.
- Experiments conducted on real-world Twitter data show the effective short text modeling ability of HGTM on clustering and topic quality evaluation.

The remainder of the paper is organized as follows. Section II gives a summary of related work and draws a comparison to HGTM. In Section III, we describe HGTM for short text topic understanding. Experimental results and analysis are given in Section IV. Section V concludes and discusses future work.

II. Related Works

Topic models have been widely used to discover the latent semantic structures of the corpus. Many powerful topic models for document analysis have been proposed, such as LSA[7], PLSI[4], LDA[5]. They have been successful in traditional tasks for the long document understanding[5]. However, traditional topic models fail in modeling tweets due to the severe sparseness and noise in short tweets[2][6].

Two kinds of methods have been proposed to tackle the serious sparseness and noise in tweets. One is to aggregate tweets as a large document. Typically, Hong et al.[6] aggregated tweets by the same user, the same word or the same hashtag. Mehrotra et al.[2] investigated different pooling schemes for LDA process. Yan et al.[8] clustered tweets by a non-negative matrix factorization before modeling topics. The other alternative is to inflate or to link short texts with additions from auxiliary long texts to enrich short texts. Hu et al.[9] organized tweets by transforming them to a semantic structure tree via term relationship defined in Wikipedia and WordNet. Besides content mining, a few works have used semi-structured information (hashtags or labels) for tweet modeling. Labeled LDA[10] was introduced to control relationship between tweets via manual defined supervision labels.

Besides tweets, many approaches take advantage of tags or labels for normal text mining. Typically, Tag-Weighted Dirichlet Allocation (TWDA)[11] introduced label prior on the topic distribution of each document. The idea of tag weighting in [11] is related to ours to some extent, but our hashtag weighting information is based on the wisdom of crowds rather than a prior determined by academic experience and testing. Meanwhile, the Author-Topic Model (ATM)[12] also can be seen as a way of modeling texts via tags, by treating tags as authors. Hence, we compare with ATM as a strong baseline for our model.

III. Hashtag Graph based Topic Model

A. Notations

A hashtag graph is an undirected graph, denoted as $\mathscr{G} = (V,E)$ where nodes $V = \{h\}_{h=1}^{H}$ are hashtags, and edges $E = \{(e_{ij})\}_{i,j\in V, i\neq j}$ are hashtag relationships. The edge e_{ij} is weighted based on the association between hashtag i and hashtag j. There are various hashtag relations in the tweet corpus, such as two hashtags appearing in the same tweets or being added in tweets with the same URLs, which reflect semantic relevancy between hashtags. Such relationships can be modeled as a hashtag relation matrix G, in which the entry at the i^{th} row represents hashtag i's incident vector, g_{ij} is the association weight measuring the co-occurrences of hashtag i and j.

In HGTM, we define a corpus as $D = \{d\}_{d=1}^{M}$, with a word dictionary $\{w\}_{w=1}^{W}$ and a hashtag dictionary $\{h\}_{h=1}^{H}$. Suppose that document d is related to a word sequence $\mathbf{w}_d = \{w_{d1},\ldots,w_{di},\ldots,w_{dN_d}\}$ and a hashtag sequence $\mathbf{h}_d = \{h_{d1},\ldots,h_{di},\ldots,h_{dH_d}\}$, where N_d (H_d) is the number of words (hashtags) in document d. Similar to LDA, each topic from T topics is typically represented by a distribution over words as ϕ with β as the hyperparameter. In particular, we characterize each hashtag by a distribution over topics as θ with α as the hyperparameter. We allocate a topic assignment z_{di} and a hashtag assignment y_{di} for each word w_{di} in the document d. Here, we use boldface letters $\mathbf{z}$ and $\mathbf{y}$ to denote all of the topic and hashtag assignments respectively, which are both an N-dimensional vector. The part of $\mathbf{z}$ and $\mathbf{y}$, $\mathbf{z}_d$ and $\mathbf{y}_d$, are topic assignment and hashtag assignment vectors for a specific document d.

B. The Generative Process of Tweets in HGTM

HGTM is a probabilistic generative model that describes a process of generating a semi-structured tweet collection with weakly-supervised information from both explicit hashtag appearance and hashtag relation graphs. HGTM associates each word with a "hashtag-topic" assignment pair. Parameterizations of HGTM are given as follows:

$$\begin{gathered}\theta_i|\alpha \sim Dirichlet(\alpha)\\ \phi_i|\beta \sim Dirichlet(\beta)\\ y_{di}|\mathbf{h}_d, g_{\mathbf{h}_d}, \tau \sim Benoulli(\tau)\\ z_{di}|\theta_{y_{di}} \sim Multinomial(\theta_{y_{di}})\\ w_{di}|\phi_{z_{di}} \sim Multinomial(\phi_{z_{di}})\end{gathered}$$

We introduce a Benoulli variable τ to decide whether to assign hashtags in tweet d for current word w_{di}, and connect different words in semantically related tweets. The generative process for HGTM is given as follows, shown in Figure 2.

1 T, α, β, τ have been predefined

2 For each of the hashtags $h = 1:H$, sample the mixture of topics $\theta_h \sim Dir(\alpha)$

3 For each of the topics $t = 1:T$, sample the mixture of words $\phi_t \sim Dir(\beta)$

4 For each of the documents $d = 1:D$, sample its length N_d, and give its hashtag set $\mathbf{h_d}$

   For each word w_{di}, $\quad i = 1:N_d$

      1) sample a hashtag y^1_{di}, $\quad y^1_{di} \sim Uniform(\mathbf{h_d})$

      2) sample r, $\quad r \sim Bernoulli(\tau)$

      3) if $r = 1$, sample $y_{di} = y^1_{di}$,
      if $r = 0$,
      sample $y_{di} \sim Multinomial(norm(g_{y^1_{di}}))$

      4) sample a topic $z_{di} \sim Multinomial\left(\theta_{y_{di}}\right)$

      5) sample a word $w_{di} \sim Multinomial\left(\phi_{z_{di}}\right)$

In step 3), $norm(g_{y^1_{di}})$ is an H-dimensional association probability vector, where the j^{th} element is:

$$p(y_j|y^1_{di}) = \frac{g_{y^1_{di},y_j}}{\sum_{j'} g_{y^1_{di},y_{j'}}}. \qquad (1)$$

Equation (1) reflects the compactness of semantic relationships between hashtags, and indirectly tells the degree of words' semantic relationship in different tweets containing related hashtags. In HGTM, the association weight shows us similarity between topic distributions of different hashtags. The statistics of hashtag relations avoid the necessity of word co-occurrence, and transmit semantic information, and furthermore enhance topic modeling. The phenomenon is in accord with how users exchange information and communicate with others in the microblogging platforms to a great extent.

The key step of generative process is to sample a correlated hashtag for the current word. Under the observation of hashtags, we model the process of tweets by using a two-step procedure of hashtag selection (called *two-step sampling*). Firstly, we sample a hashtag y^1_{di} uniformly from $\mathbf{h}_d$. Secondly, we sample an r from a Bernoulli distribution to decide whether the current word is related to *explicit hashtags* $\mathbf{h}_d$. If so, y_{di} equals to y^1_{di}; if not, sample y_{di} from the multinomial distribution of $norm(g_{y^1_{di}})$, which means to sample from *potential hashtags* from g_{h_d}. After hashtag assignment, we draw the topic assignment z_{di} from the multinomial distribution with parameter $\theta_{y_{di}}$. Now, we finish the assignment of two latent variables (a "hashtag-topic" pair).

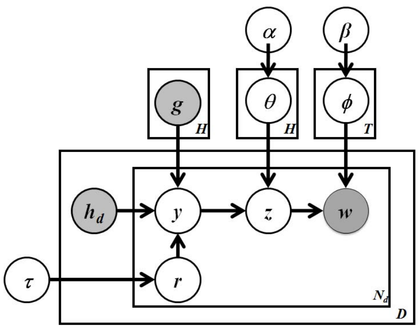


Fig. 2. Graphical model representation of HGTM. The boxes represent replicates. The outer plate represents tweets, while the inner plate represents the repeated assignment of hashtags, topics and words within a tweet.

In HGTM, we model users' arbitrary of hashtag selection via a sticky factor $\tau \sim [0,1]$. The hyperparameter τ defines the possibility that hashtag assignments are from hashtags in the current document, i.e. $\tau \sim p(y \in \mathbf{h}_d)$. The lower the τ is, the higher randomness is, and vice versa. Through the sampling process, we simulate the users' randomness during hashtag selection and seamlessly integrate correlated hashtags to enhance the semantic relationship between short texts with lack of co-occurrence. When τ less than 1, the model adds word co-occurrence via latent hashtag assignments even if they have less or no co-occurrence.

C. Parameter Estimation

In HGTM, the words, hashtags and hashtag graphs in microblogs are observed while the hidden variables (i.e. the hashtag assignment $\mathbf{y}$ and the topic assignment $\mathbf{z}$) are guided by latent distribution parameters, the H hashtag-topic distribution θ and the T topic-word distribution ϕ. In order to infer the hidden variables, we compute the posterior distribution of the hidden variables given the observed variables. Generating probability of the whole corpus is:

$$p(\mathbf{w}|\theta,\phi,r,\mathbf{h},G) = \prod_{d=1}^{D} p(\mathbf{w}_d|\theta,\phi,r,\mathbf{h}_d,\mathbf{g}_{h_d}). \qquad (2)$$

Following the above procedure, we assume that "topic-word" distribution and "hashtag-topic" distribution are conditionally independent. For each document d, we have:

$$\begin{aligned} p(\mathbf{w}_d|\theta,\phi,r,\mathbf{h}_d,\mathbf{g}_{h_d}) &= \prod_{i=1}^{N_d} p(w_{di}|\theta,\phi,r,\mathbf{h}_d,\mathbf{g}_{\mathbf{h}_d})\\ &= \prod_{i=1}^{N_d}\sum_{s=1}^{H}\sum_{t=1}^{T} p(w_{di},z_{di}=t,y_{di}=s|\theta,\phi,r,\mathbf{h}_d,\mathbf{g}_{\mathbf{h}_d})\\ &= \prod_{i=1}^{N_d}\sum_{s=1}^{H}\sum_{t=1}^{T} p(w_{di}|z_{di}=t,\phi)p(z_{di}=t|y_{di}=s,\theta)p_{sy_{di}}\\ &= \prod_{i=1}^{N_d}\sum_{s=1}^{H}\sum_{t=1}^{T} \phi_{w_{di}t}\theta_{ts}p_{sy_{di}}, \end{aligned} \qquad (3)$$

where $p_{sy_{di}} = p(y_{di} = s|r, \mathbf{h}_d, g_{\mathbf{h}_d})$ indicates the probability of hashtag assignment s conditioned on the current *explicit hashtags* $\mathbf{h}_d$ and related *potential hashtags*. From the two-step sampling discussed earlier, we assign hashtags to words according to co-occurrence and statistic correlation relationship between hashtags:

$$\begin{aligned} & p(y_{di} = s|r, \mathbf{h}_d, g_{\mathbf{h}_d}) \\ & \quad = [p(y^1_{di} = s|\mathbf{h}_d)p(y_{di} = s|y^1_{di})]^r \cdot \\ & \quad \quad [\sum_{j=1}^{H_d} p(y^1_{di} = h_{d,j}|h_d)p(y_{di} = s|y^1_{di} = h_{d,j}, g_{h_{d,j}})]^{1-r}, \end{aligned} \tag{4}$$

where $h_{d,j}$ is the j^{th} hashtag in document d. When $r = 1$, there is no need of using related hashtags. When applied to probabilistic topic models[4], this method is susceptible to local maxima and is computationally inefficient[5]. Hence, we employ an alternative parameter estimation strategy, Gibbs sampling procedure[13] - a fast and efficient MCMC (Markov Chain Monte Carlo) algorithm, to carry out approximated parameters instead of estimating the model parameters directly. By applying Euler integration for Equation (3), we can obtain the sample posterior distribution:

$$\begin{aligned} & p(z_{di} = t, y_{di} = s, r_{di} = u|w_{di} = w, \mathbf{z}_{-di}, \mathbf{y}_{-di}, \mathbf{w}_{-di}, H, G, \alpha, \beta, \tau) \\ & \quad \propto \frac{C^{WT}_{wt,-di} + \beta}{\sum_{w'} C^{WT}_{w't,-di} + W\beta} \cdot \frac{C^{TH}_{ts,-di} + \alpha}{\sum_{t'} C^{TH}_{t's,-di} + T\alpha} \cdot p_s^{y_{di}}, \end{aligned} \tag{5}$$

where C^{WT} is the count matrix of the number of times a specific word assigned to a specific topic, C^{TH} is the count matrix of the number of times a specific topic assigned to a specific hashtag, and $-di$ means assignments except that for the current word.

After iterative sampling, it reaches the convergence. The final results of θ and ϕ are:

$$\begin{aligned} \theta_s & \propto \frac{C^{TH}_{ts} + \alpha}{\sum_{t'} C^{TH}_{t's} + T\alpha}, \\ \phi_t & \propto \frac{C^{WT}_{wt} + \beta}{\sum_{w'} C^{WT}_{w't} + W\beta}. \end{aligned} \tag{6}$$

Thus, according to the topic structure detected, HGTM can conclude distinguishable topics in microblogs and find out the clear representative words for each topic. Besides, we find out hashtags' meaning and the key hashtags under each topic.

D. New Document's Topic Distribution Inference

After parameter estimation, we can get hashtags' probability distribution over topics and topics' probability distribution over words. For a new tweet with foregone hashtags, we infer its topic distribution by sampling process as parameters inference, but latent variable probability distributions are static by using the parameters obtained from the stage of training.

IV. EXPERIMENTAL ANALYSIS

Based on our hashtag graph, we enhance short text topic modeling. In this section, we conduct experiments to verify the effectiveness of our model and fulfill the task of text (hashtag) clustering on an English microblogging data set published in TREC 2011 microblog track, named "TweetData"[1]. It contains nearly 16 million tweets sampled from January 23rd and February 8th in 2011. To reduce low quality tweets, we process the raw dataset via the similar normalization steps as [3] does. For a better analysis, we conduct our experiments on tweets containing hashtags and remove the retweets. At last, we get 333,491 tweets, 12,420 distinct words, 106,682 hashtags and 20,311 users. The average document length is 5.22. The average number of hashtags in a single tweet is 1.42. The training data set containing tweets from January 23rd to February 6th is used for inferring model's parameters, and the remaining tweets are used as our testing data set.

We compare HGTM[2] with six other models: 1) VSM, the Vector Space Model, which represents a tweet using word frequencies; 2) LSA, the Latent Semantic Analysis model, which decomposes the "document-word" matrix by Singular Value Decomposition; 3) LSAH, which aggregates all the tweets containing the same hashtag to a pseudo-document before training; 4) LDA, the Latent Dirichlet Allocation, which takes each tweet as a document; 5) LDAH, which shares the aggregation strategy with LSAH and learns parameters with pseudo documents; 6) ATM, the Author Topic Model, which treats hashtags as "authors".

In experiments, we construct our hashtag relation graph by weighing the edges with the number of same URLs that two hashtags co-occur with. The number of topics T is fixed at 60 and the hyperparameters in generative models (LDA, LDAH, ATM, HGTM) are set at $50/T$ for α and 0.01 for β respectively. We run 5 independent Gibbs sampling chains for 2000 iterations each. In HGTM, we set τ as 0.7 for a little preference to select hashtags in the current tweet.

A. Qualitative Analysis of Topics

Here we study the quality of topics and representative hashtags discovered by our model. Table I tells examples of 6 topics (out of 60) learnt by HGTM. We use one word with the highest probability to denote the topic.

Referred to the results, the top-10 words and top-10 hashtags are highly related to the specific topic. For topic "egypt", we discover most important key words, such as "egypt, people, obama, mubarak and police". Meanwhile, HGTM finds out highly related hashtags, such as "#egypt", "jan25" and et al. For Topic "SONG", the top-10 words are much prominent and precise about music things. It also helps someone who is not familiar with Twitter to guess that "#nowplaying" is the expanded form of "#np". Topic "GAME" is oriented towards a popular sport event at that time-"Super Bowl", along with a hashtag list showing some favorites, "#nfl" (the National Football League), "#steelers" (the Steelers) and "#packers" (the Green Bay Packers). The results about "SNOW" and "LIFE" also show their own characteristics. However, we discover that some functional hashtags, such as "#fb" (Tweets ending in "#fb" are automatically imported to Facebook), dominate in multiple topics. This phenomenon results from hashtags usage and reflects hotter topics in Facebook to some extent, such as life related things ("GAME" and "SNOW"), inferring from hashtag probability under each topic.

[1] http://trec.nist.gov/data/microblog2011.html

[2] The code is available : http://kdd.nankai.edu.cn/sourcecode/HGTM.html

EGYPT		SONG		JOB		GAME		SNOW		LIFE	
WORD	PROB.	WORD	PROB.	WORD	PROB.	WORD	PROB.	WORD	PROB.	WORD	PROB.
egypt	0.01761	song	0.0363	job	0.0363	game	0.02755	snow	0.02804	life	0.035
people	0.01649	listen	0.03078	service	0.0185	super	0.02569	home	0.02462	people	0.02627
obama	0.01201	album	0.01527	sale	0.01667	bowl	0.02219	today	0.02214	god	0.02225
mubarak	0.01098	play	0.01339	manager	0.01479	show	0.019	feel	0.02159	thing	0.01882
egyptian	0.01073	radio	0.011	web	0.01267	fan	0.01696	morning	0.01636	friend	0.01586
police	0.0095	feat	0.01061	host	0.01153	year	0.01641	wind	0.0148	man	0.01353
protest	0.00882	sound	0.00897	project	0.0115	tonight	0.0131	weather	0.01353	heart	0.01255
president	0.00802	live	0.00862	business	0.01048	green	0.01307	cold	0.01205	true	0.0125
state	0.00794	mix	0.00836	design	0.01024	play	0.01268	tomorrow	0.01202	give	0.01074
news	0.00757	music	0.00831	website	0.00755	team	0.01231	feb	0.01069	mind	0.00964
HASHTAG	PROB.	HASHTAG	PROB.	HASHTAG	PROB.	HASHTAG	PROB.	HASHTAG	PROB.	HASHTAG	PROB.
#egypt	0.04174	#nowplaying	0.03413	#jobs	0.04728	#superbowl	0.01584	#weather	0.00656	#quote	0.00727
#25-Jan	0.02866	#np	0.01915	#job	0.00596	#fb	0.00445	#fb	0.00583	#damnitstrue	0.00653
#tcot	0.00745	#music	0.00422	#hiring	0.00307	#steelers	0.0039	#wdisplay	0.00182	#ihatequotes	0.00512
#news	0.007	#lastfm	0.00282	#fb	0.00285	#nfl	0.0038	#nowplaying	0.00179	#fb	0.0043
#sotu	0.00685	#fb	0.00255	#freelance	0.00263	#packers	0.00354	#tcyasi	0.00147	#zodiacfacts	0.00313
#p2	0.00401	#soundcloud	0.00242	#in	0.00231	#sb45	0.00246	#news	0.0014	#nowplaying	0.00288
#fb	0.00364	#itunes	0.00176	#hosting	0.0022	#bears	0.00214	#realestate	0.00133	#egypt	0.00271
#mubarak	0.00361	#blogtalkradio	0.00174	#news	0.00159	#jets	0.00186	#np	0.0013	#np	0.0024
#dearjohn	0.00284	#listeningto	0.00154	#careers	0.0013	#sotu	0.00141	#egypt	0.00114	#tls	0.00234
#tahrir	0.00249	#pandora	0.00125	#ebc	0.00118	#nowplaying	0.00128	#travel	0.00108	#viatumblr	0.00223

TABLE I. AN ILLUSTRATION OF 6 TOPICS FROM A 60-TOPIC SOLUTION. EACH TOPIC IS SHOWN WITH TOP-10 WORDS AND HASHTAGS RANKING IN THE HIGHEST PROBABILITY.

B. Text Clustering

In order to see topic representation ability for tweets, we observe the performance of text clustering. Unluckily, there is no category information in tweet data. We take hashtags in tweets as cluster labels of tweets. We take 50 frequent hashtags (the same as [3]) as cluster labels. Except VSM, we map each tweet in a new space as a vector, denoted as **d**. By topic models (LDA, LDAH, ATM, HGTM), each tweet is represented by a probability distribution over topics. By LSA and LSAH, each tweet is mapped to a low-dimensional vector.

Evaluation is based on the fact that the clusters should have lower intra-cluster distance and higher inter-cluster distance. Here, we use cosine similarity to measure the similarity degree of two documents. So the distance between two tweets is:

$$dis\,(d_1,d_2) = 1 - \frac{\mathbf{d_1} \cdot \mathbf{d_2}}{\|\mathbf{d_1}\|\|\mathbf{d_2}\|}. \tag{7}$$

The average intra-cluster distance is:

$$IntraDis\,(C) = \frac{1}{K}\sum_{k=1}^{K}[\sum_{d_i,d_j \in C_k, i\neq j} \frac{2dis(d_i,d_j)}{\|C_k\|\|C_k-1\|}]. \tag{8}$$

The average inter-cluster distance is:

$$InterDis\,(C) = \frac{1}{K(K-1)} \sum_{C_k,C_{k'}\in C, k\neg k'} [\sum_{d_i\in C_k}\sum_{d_j\in C_{k'}} \frac{2dis(d_i,d_j)}{\|C_k\|\|C'_k\|}]. \tag{9}$$

If average intra-cluster distance is much less than average inter-cluster distance, the model achieves a more clear topic description. So we calculate the ratio H score between them, where a smaller value is better.

$$H = \frac{IntraDis(C)}{InterDis(C)} \tag{10}$$

From results shown in Table II, we have the following conclusions. (1) HGTM achieves the best performance (with p-value <0.001). (2) Aggregation strategy brings traditional models an obvious improvement. LDAH and LSAH outperform LDA by 0.1 and LSA by 0.04 respectively. The results verify that word co-occurrence frequency has an important impact on LSA and LDA. Hence, due to data sparsity, we could not get a satisfactory distinguish result even when aggregating tweets by user-contributed hashtags. (3) VSM is the worst for semantic similarity measure. It suggests that the arbitrary nature of language has a much severer impact on VSM than that on ATM and HGTM. (4) ATM also considers the hashtag co-occurrence in a single tweet, and then exceeds other models except HGTM. The results tell that it is more helpful to model the strength of word semantic relationship by considering *explicit hashtags* and *potential hashtags* via two-step sampling.

TABLE II. H SCORE FOR TEXT CLUSTERING ON THE TWEETDATA COLLECTION. THE SIGNIFICANCE LEVELS (P-VALUE BY T-TEST) ARE DENOTED AS 0.1*, 0.01**, 0.001***.

Model	H score	Significant differences
VSM	0.961	-
LSA	0.877	>VSM***
LSAH	0.838	>LSA*** >VSM***
LDA	0.817	>LSAH*** >LSA*** >VSM***
LDAH	0.718	>LDA** >LSAH*** >LSA*** >VSM***
ATM	0.477	>ALL*** except HGTM
HGTM	0.467	>ALL ***

C. Hashtag Clustering

Hashtag probability distribution over topics is a vital by-product of HGTM. In order to illustrate whether our method is susceptible to different categories of topics and events, we evaluate our model by the task of hashtag clustering. Our aim is to see the learning capacity of distinguish hashtags in different semantic domains. For each hashtag, we aggregate all tweets containing the same hashtag to construct a pseudo document for inferring its topic distribution. We use hashtag class information manually provided in [14] as cluster labels. We remove the ambiguous hashtags in the "NONE" cluster. Finally, we use 336 hashtags from 8 clusters in our experiment. The details of labels are shown in Table III.

We take the same metric (H score) in Section IV-B for evaluation. The results are shown in Table IV. The results show that

TABLE III. Label information of hashtags.

Class	#hashtag	Examples
IDIOMS	126	#ihate, #cantcandidateyou, #followback
POLITICAL	39	#Jan25, #tcot, #glennbeck, #obama, #hcr
TECHNOLOGY	57	#nikeplus, #teamautism, #amwriting
SPORTS	42	#golf, #yankees, #nhl, #cricket
MOVIES	32	#lost, #glennbeck, #bones, #newmoon
CELEBRITY	4	#mj, #brazilwantsjb, #regis, #iwantpeterfacinelli
GAMES	13	#mafiawars, #spymaster, #mw2, #zyngapirates
MUSIC	23	#thisiswar, #musicmonday, #pandora

TABLE IV. *H* score for hashtag clustering on the TweetData collection. The significance levels (P-value by t-test) are denoted as 0.1*, 0.01**, 0.001***.

Model	H score	Significant differences
VSM	0.946	>LDA**
LSA	0.823	>VSM*** >LDA***
LSAH	0.751	>VSM*** >LSA*** >LDA***
LDA	0.991	-
LDAH	0.639	>VSM*** >LSA*** >LSAH*** >LDA** >ATM*
ATM	0.659	>VSM*** >LSA*** >LSAH*** >LDA***
HGTM	0.589	>VSM*** >LSA*** >LSAH*** >LDA*** >LDAH* >ATM**

HGTM performs significantly better than other models. Similar to tweet clustering, performance of unsupervised semantic methods (LDA and LSA) can be improved by aggregation strategy in the case of noisy text data. LSAH and LDAH achieve a lower H-score than LSA and LDA respectively. In particular, LDAH outperforms LDA by 33.01%. Comparing LSA-type methods to LDA-type methods, the latent topic structures discovered by generative probability models are more suitable for users hashtag semantic understanding.

We find LDA is a little worse than VSM. An explanation for this phenomenon is that modeling topic relevance information from a short and casual post probably only captures weak semantic description, but not the topic representation. However, aggregated messages give us topic illustration of higher quality[2][6]. Ambiguous and noisy topics learnt by LDA are accumulated and amplified in aggregated pseudo documents. LSA reduces noise via low dimensional approximation process. That's why LSA is much better than LDA.

V. Conclusion and Future Works

The HGTM proposed in this paper first introduces a hashtag relation graph as weakly-supervised information for effective tweet semantic modeling to handle both severe sparseness and noise in tweets. Compared to single document-oriented topic models and aggregation strategies, HGTM has a better ability to capture semantic relations between words within or without co-occurrence by utilizing the wisdom of crowds from user-generated hashtags. Besides, HGTM discovers more readable and distinguishable topics than previous models. Furthermore, we would like to explore different hashtag relations for tweets, and to model time-sensitive hashtag relations. Moreover, we want to explore some vital tasks under HGTM, such as hashtag recommendation, short text retrieval, event detection, etc.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (No. 61105049 and No. 61300166), the National Science Fund for Distinguished Young Scholars (No. 61222210), the State Key Program of National Natural Science of China (No. 61432011), the Open Project Foundation of Information Technology Research Base of Civil Aviation Administration of China (No. CAAC-ITRB-201303 and No. CAAC-ITRB-201204), the Natural Science Foundation of Tianjin (No. 14JCQNJC00600), the Science and Technology Planning Project of Tianjin (No. 13ZCZDGX01098) and the Tianjin Key Laboratory of Cognitive Computing and Application.

References

[1] Y. Chen, H. Amiri, Z. Li, and T.-S. Chua, "Emerging topic detection for organizations from microblogs," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '13. New York, NY, USA: ACM, 2013, pp. 43–52.

[2] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie, "Improving lda topic models for microblogs via tweet pooling and automatic labeling," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '13. New York, NY, USA: ACM, 2013, pp. 889–892.

[3] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22Nd International Conference on World Wide Web*, ser. WWW '13. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2013, pp. 1445–1456.

[4] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '99. New York, NY, USA: ACM, 1999, pp. 50–57.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[6] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the First Workshop on Social Media Analytics*, ser. SOMA '10. New York, NY, USA: ACM, 2010, pp. 80–88.

[7] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS*, vol. 41, no. 6, pp. 391–407, 1990.

[8] X. Yan, J. Guo, S. Liu, X.-q. Cheng, and Y. Wang, "Clustering short text using ncut-weighted non-negative matrix factorization," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, ser. CIKM '12. New York, NY, USA: ACM, 2012, pp. 2259–2262.

[9] X. Hu, L. Tang, and H. Liu, "Enhancing accessibility of microblogging messages using semantic knowledge," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, ser. CIKM '11. New York, NY, USA: ACM, 2011, pp. 2465–2468.

[10] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '09. Stroudsburg, PA, USA: ACL, 2009, pp. 248–256.

[11] S. Li, G. Huang, R. Tan, and R. Pan, "Tag-weighted dirichlet allocation," in *Proceedings of the 13th International Conference on Data Mining*, ser. ICDM'13. Los Alamitos, CA, USA: IEEE Computer Society, 2013, pp. 438–447.

[12] D. M. Blei, A. Y. Ng, and M. I. Jordan, "The author-topic model for authors and documents," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[13] T. Griffiths, "Gibbs sampling in the generative model of Latent Dirichlet Allocation," Stanford University, Tech. Rep., 2002. [Online]. Available: www-psych.stanford.edu/~gruffydd/cogsci02/lda.ps

[14] D. M. Romero, B. Meeder, and J. Kleinberg, "Differences in the mechanics of information diffusion across topics: Idioms, political hashtags, and complex contagion on twitter," in *Proceedings of the 20th International Conference on World Wide Web*, ser. WWW '11. New York, NY, USA: ACM, 2011, pp. 695–704.