2016 IEEE 16th International Conference on Data Mining

Topic Discovery for Short Texts Using Word Embeddings

Guangxu Xun, Vishrawas Gopalakrishnan, Fenglong Ma
Yaliang Li, Jing Gao, Aidong Zhang
Department of Computer Science and Engineering
SUNY at Buffalo, NY, USA
Email: {guangxux, vishrawa, fenglong, yaliangl, jing, azhang}@buffalo.edu

***Abstract*—Discovering topics in short texts, such as news titles and tweets, has become an important task for many content analysis applications. However, due to the lack of rich context information in short texts, the performance of conventional topic models on short texts is usually unsatisfying. In this paper, we propose a novel topic model for short text corpus using word embeddings. Continuous space word embeddings, which is proven effective at capturing regularities in language, is incorporated into our model to provide additional semantics. Thus we model each short document as a Gaussian topic over word embeddings in the vector space. In addition, considering that background words in a short text are usually not semantically related, we introduce a discrete background mode over word types to complement the continuous Gaussian topics. We evaluate our model on news titles from data sources like *abcnews*, showing that our model is able to extract more coherent topics from short texts compared with the baseline methods and learn better topic representation for each short document.**

***Keywords*-short texts, topic model, word embeddings**

I. Introduction

With more than five Exabytes of data being generated in less than two days [1], recent researches in Internet and social media focus on effective ways for data management and content presentation. Social networks on their part attempt to handle this by trying to provide a cohesive yet real-time view on a topic by partitioning the data into "Trending Topics" by hashtag or text mentions. However, such explicit categorization is either not possible or comes at a high cost in other domains like news titles, text advertisements, questions/tasks in crowd sourced applications, etc. To this end, topic models have proven to be a useful tool in unsupervised text analyses and pattern discovery in a corpus. Extracting meaningful topics helps us better analyze the documents, reduce the dimensionality of documents (allowing faster analyses) and is also crucial for many content analysis tasks, e.g. dynamic topic detection and topic expertise discovery [2], [3], [4], [5], [6]. However, the efficacy of conventional topic models is limited by the lack of rich context in short texts. The limitation stems from the fact that each individual document, by itself, is too short for effective topic extraction.

Conventional topic models, such as Probabilistic Latent Semantic Analysis (PLSA) [7] and Latent Dirichlet Allocation (LDA) [8], follow the bag-of-word assumption and model documents as mixtures of latent topics, where topics are multinomial distributions over words. Bayesian methods are then employed to learn the topic distribution for each document based on the document-word frequency matrix of the corpus. However, compared with regular documents, short texts are suffering from the lack of rich context. Short texts like news titles or tweets usually span only 10-30 word long, e.g. Twitter imposes a limit of 140 characters on each tweet. From a statistical point of view, this problem will heavily limit the quality of topics extracted from short texts by conventional topic models.

To overcome the lack of context information in short text corpus and exploit external semantics, we develop a new topic model for short text using word embeddings [9] in continuous vector space. Word embeddings, also known as word vectors and distributed representations of words, have proven to be effective at capturing semantic regularities in language: words with similar semantic and syntactic attributes are projected into the same area in the vector space. More specifically, first we use Wikipedia as an external source to train word embeddings upon it. The resulting semantic regularities are then used as a supplementary information to overcome the limitation of context information in short texts. Second, in the vector space of word embeddings, we formulate topics using Gaussian distributions to handle the "continuous" space of word embeddings. The primary motivation behind this modeling is that since we are now in vector space and semantically related words are located close to each other, Gaussian distribution over the word embeddings denotes the semantic centrality. Third, instead of viewing each short text as a mixture of topics, we assume each such text focuses on only one Gaussian topic. This assumption is plausible as the size of text is in the range of 10-30 words. Fourth, considering the fact that most background words are not semantically related, we add the background mode with discrete multinomial distribution of words to complement the Gaussian topics. Thus, we are able

2374-8486/16 $31.00 © 2016 IEEE
DOI 10.1109/ICDM.2016.33

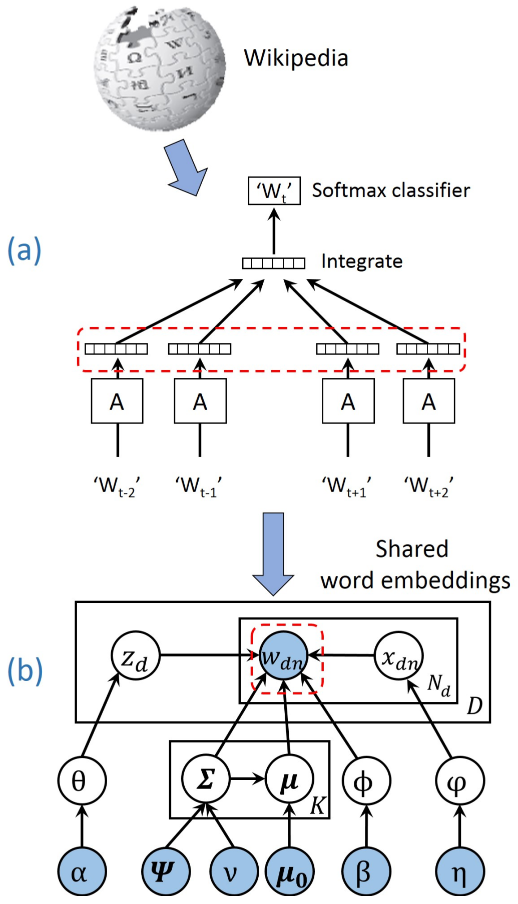

Figure 1. Schematic illustration of topic discovery for short texts. Part (a) represents the word embedding learning process. Part (b) represents the topic modelling in presence of word embedding for short texts.

to extract better topics from short text.

II. Methodology

In this section, we discuss the proposed methodology to extract high quality topics from short texts. An end-to-end framework is shown in Figure 1.

A. Learning Word Embeddings from Wikipedia

In our approach, we learn word embeddings using Wikipedia as the external source. The motivation of using Wikipedia lies in the sheer range of topics and subjects that are covered. Extracting word embeddings from Wikipedia allows us to "enrich" the short text with additional semantics. The part (a) of Figure 1 illustrates the training of Continuous Bag of Words (CBOW) word embeddings using Word2Vec tool [10].

Having learnt the word embeddings, given a word w_{dn}, which is the n^{th} word in d^{th} document, we can enrich that word by replacing it with the corresponding word embedding (red blocks in Figure 1). The following section describes how this enrichment is used in a generative process to extract a single topic for a given short document.

B. Strategies and Generative Process

Wikipedia word embeddings give us useful additional semantics, which is crucial due to the lack of context information in short texts. However, as the documents are now sequences of word embeddings instead of sequences of word types, conventional topic models no longer are applicable. Since the discrete word types are now replaced by continuous space of word embeddings, and those word vectors are allocated in space based on their semantics and syntax, we consider them as draws from several Gaussian distributions. Hence, each topic is characterized as a multivariate Gaussian distribution in the vector space. The choice of Gaussian distribution is justified by the observations that Euclidean distances between word embeddings correlate with their semantic similarities.

Another important observation about short texts is that each short text usually consists of only one topic instead of a combination of multiple topics. Inspired by Twitter-LDA [11], we assume each document is about one single Gaussian topic. Thus, the words in a document either belong to the document's topic or to the background mode.

However, it is not accurate to continue using word embeddings for the background mode. This is because background words are not semantically interrelated and hence, we cannot find a semantically correspondent Gaussian distribution to their physical locations in the vector space. Thus, in the background mode, we use discrete word types rather than continuous word embeddings to represent words.

More formally, a document d is construed to be of a single Gaussian topic, represented by z_d in Figure 1 part (b). The corresponding parameter that controls the latent topic distribution is θ and the hyper-parameter for that distribution is α. Word w_{dn} can either be a topic word or a background word, we consider both factors. For a topic, it is represented by a multivariate Gaussian distribution in the word vector space and $\boldsymbol{\mu}_k$ denotes the mean and $\boldsymbol{\Sigma}_k$ denotes the covariance for the k^{th} topic. $\boldsymbol{\Psi}$ is hyper-parameter covariance matrix and ν is the hyper-parameter denoting the initial degree of freedom. $\boldsymbol{\mu}_0$ is the hyper-parameter for mean. ϕ represents the multinomial distribution for background words for which the corresponding hyper-parameter is β. The fact that whether the word w_{dn} is a background word or not is depicted by an indicator variable x_{dn}, whose parameter is φ representing the Bernoulli distribution. The corresponding hyper-parameter for that distribution is η. Variables in bold font mean they are either vectors or matrices. The generative process is as follows:

1) Draw $\theta \sim Dir(\alpha)$.
2) Draw $\phi \sim Dir(\beta)$.
3) Draw $\varphi \sim Dir(\eta)$.
4) For each topic $k = 1, 2, \cdots, K$:

a) Draw topic covariance $\mathbf{\Sigma}_k \sim \mathcal{W}^{-1}(\mathbf{\Psi}, \nu)$.
b) Draw topic mean $\boldsymbol{\mu}_k \sim \mathcal{N}(\boldsymbol{\mu}_0, \frac{1}{\tau}\mathbf{\Sigma}_k)$.

5) For each text $d = 1, 2, \cdots, D$:
   a) Draw a topic $z_d \sim Multinomial(\theta)$.
   b) For each word index $n = 1, 2, \cdots, N_d$:
      i) Draw a word category $x_{dn} \sim Bernoulli(\varphi)$.
      ii) Draw a word. If $x_{dn} = 1$, Draw topic word $\boldsymbol{w}_{dn} \sim \mathcal{N}(\boldsymbol{\mu}_{z_d}, \mathbf{\Sigma}_{z_d})$; otherwise, draw background word $w_{dn} \sim Multinomial(\phi)$.

Note that τ in step 4 (b) is a constant factor. We use the following conjugate priors: a Gaussian distribution $\mathcal{N}$ for the mean and an inverse Wishart distribution $\mathcal{W}^{-1}$ for the covariance.

C. Model Details

When $x_{dn} = 1$, the current word $\boldsymbol{w}_{dn}$ is a topic word and it corresponds to a Wikipedia word embedding; otherwise, the current word w_{dn} is a discrete background word. Hence, the conditional probability of the current word w_{dn} is:

$$p(\boldsymbol{w}_{dn}|x_{dn}, z_d, \phi, \boldsymbol{\mu}, \mathbf{\Sigma}) \propto \left(f(\boldsymbol{w}_{dn}|\boldsymbol{\mu}_{z_d}, \mathbf{\Sigma}_{z_d})\right)^{x_{dn}} (\phi_{w_{dn}})^{1-x_{dn}}$$

where function $f(\boldsymbol{w}|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k)$ is the probability density of topic k's Gaussian distribution. Thus, for document d of N_d words, the conditional probability is:

$$\begin{aligned} & p(\boldsymbol{w}_d|\theta, \phi, \varphi, \boldsymbol{\mu}, \mathbf{\Sigma}) \\ & = \sum_z p(z|\theta) \left(\prod_{n=1}^{N_d} p(x_{dn}|\varphi) p(\boldsymbol{w}_{dn}|x_{dn}, z, \phi, \boldsymbol{\mu}, \mathbf{\Sigma})\right). \end{aligned} \tag{1}$$

Thus, for the corpora $\mathcal{D} = \{d\}_1^D$, we can obtain the overall probability $p(\mathcal{D}|\alpha, \beta, \eta, \mathbf{\Psi}, \nu, \boldsymbol{\mu}_0)$ by integrating out the intermediate variables. Furthermore, the objective function to minimize is the log likelihood of the corpora:

$$O = -\log p(\mathcal{D}|\alpha, \beta, \eta, \mathbf{\Psi}, \nu, \boldsymbol{\mu}_0). \tag{2}$$

D. Estimation and Parameter Inference

The observed variables are documents consisting of word types and word embeddings, and our goal is to infer the posterior distributions over the Gaussian topics and background mode along with topic assignments of words. We use Gibbs-EM to infer the parameters [12]. We begin by first fixing the other variables and derive a collapsed Gibbs sampler that samples document topic z_d document by document. The probability for sampling document topic z_d is:

$$\begin{aligned} & p(z_d = k|\boldsymbol{z}_{-d}, D, \boldsymbol{\mu}, \mathbf{\Sigma}, \alpha, \beta, \boldsymbol{x}) \\ & \propto (n^k_{-d} + \alpha) \cdot \prod_{n=1}^{N_d} \left(T_r(\boldsymbol{w}_{dn}|\boldsymbol{\mu}_k, \frac{\tau_k + 1}{\tau_k}\mathbf{\Sigma}_k)\right)^{x_{dn}}, \end{aligned} \tag{3}$$

where n^k_{-d} denotes the number of times that topic k is sampled, without counting current document d. $T_r(\boldsymbol{w}|\boldsymbol{\mu}, \mathbf{\Sigma})$ is the multivariate Student's t-distribution for Gaussian sampling with $(r = \nu_k - dim + 1)$ being its degrees of freedom and dim being the dimensionality of word embeddings. $(\nu_k = \nu + n^k)$ and $(\tau_k = \tau + n^k)$ are the parameters of topic k, where n^k represents the total number of words that are assigned to topic k. x_{dn} is the topic/background indicator for word w_{dn}.

Then after the document topic z_d is sampled, for each word w_{dn} in document d, we sample the topic/background indicator x_{dn} according to the Bernoulli distribution:

$$\begin{aligned} & p(x_{dn}|\boldsymbol{w}, \boldsymbol{x}_{-w}, z_d = k, D, \boldsymbol{\mu}, \mathbf{\Sigma}, \alpha, \beta) \\ & \propto \left(\frac{n^{-w}_{x=1} + \eta}{n^{-w}_{x=1} + n^{-w}_{x=0} + 2\eta} \cdot T_r(\boldsymbol{w}_{dn}|\boldsymbol{\mu}_k, \frac{\tau_k + 1}{\tau_k}\mathbf{\Sigma}_k)\right)^{x_{dn}} \\ & \cdot \left(\frac{n^{-w}_{x=0} + \eta}{n^{-w}_{x=1} + n^{-w}_{x=0} + 2\eta} \cdot \frac{n^{w}_{x=0} + \beta}{\sum_{w'=1}^{V} n^{w'}_{x=0} + V\beta}\right)^{1-x_{dn}}, \end{aligned} \tag{4}$$

where $n^{-w}_{x=1}$ and $n^{-w}_{x=0}$ denote the number of topic words and background words respectively, without considering the current word. $n^{w}_{x=0}$ is the number of times the current word sampled as a background word, and V is the vocabulary size. Every time z_d or x_{dn} is re-sampled, the involved Gaussian topics would change and needs to be updated. Following the idea of [13], we can derive the updates for $\boldsymbol{\mu}_k$ and $\mathbf{\Sigma}_k$ of the posterior Gaussian distributions for topic k.

III. Experiments

In this section, we conduct experiments on real-world short texts to demonstrate the effectiveness of our model. This section details the dataset, the evaluation metric, baselines and the performance of our proposed model.

A. Dataset and Baselines

The dataset used for topic discovery is crawled from *abcnews*[1]. Based on the news categories in *abcnews* website, the documents in this dataset are divided into ten groups: Entertainment, Health, U.S., International, Law, Money, Politics, Sports, Technology, and Travel. In each category, there are 1000 news documents. Each document has a title and an optional description of the corresponding news article. The average length of the description, when available, is around 20 words - very short as compared to a regular document.

We use Latent Dirichlet Allocation (LDA) [8] and Gaussian-LDA [14] as the baselines to evaluate the performance of our topic discovery. Gaussian-LDA is first proposed for audio retrieval [13] and then used to leverage another kind of continuous data – word vectors to incorporate external semantics [14].

[1] http://abcnews.go.com/

Table I
Top 10 words in each topic for LDA

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10
know	one	new	new	can	year	us	week	look	news
clinton	america	apple	years	episode	hotel	see	says	said	abc
police	back	today	time	full	cruise	airline	trump	first	family
need	first	facebook	day	best	vacation	3	presidential	game	latest
shooting	world	video	found	15	high	ceo	debate	season	big
year	obama	people	past	10	car	flight	john	state	star
hillary	made	1	birth	top	report	home	donald	last	get
everything	said	two	things	11	satisfaction	letter	new	win	ways
life	man	old	now	20	city	mom	president	homes	pope
inside	around	co	google	travel	four	plane	carson	4	save

Table II
Top 10 words in each topic for Gaussian-LDA

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10
presidential	back	clinton	hotel	leading	news	shooting	abc	latest	president
debate	apple	family	people	jobs	big	made	year	top	america
world	full	birth	abc	travel	ways	bringing	week	today	episode
candidate	flight	police	yaz	news	report	america	trump	pope	presidential
pope	time	life	part	homes	letter	inside	airline	found	house
york	control	man	years	latest	high	colorado	obama	long	game
time	plane	state	kids	microsoft	case	back	ceo	refugees	star
save	woman	woman	video	founder	things	found	cruise	called	season
talks	million	home	women	back	million	dead	vacation	recently	hunter
news	car	safety	rielle	top	family	police	hillary	remains	paris

Table III
Top 10 words in each topic for our model

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10
presidential	apple	star	year	america	captain	airline	episode	children	flight
clinton	world	world	game	infrastructure	control	hotel	full	cancer	police
campaign	officer	jenner	back	taskrabbit	birth	letter	15	mysterious	plane
2016	garrido	swift	google	business	wreck	satisfaction	pope	students	paris
trump	search	stars	big	company	yaz	complaint	star	doctors	attacks
candidate	failed	wars	family	microsoft	leading	cruise	shooting	rare	woman
president	mars	williams	life	jobs	pill	suite	francis	brain	refugees
debate	photo	photos	win	nokia	credit	vacation	20	price	city
hillary	shooting	bruce	abc	homeowners	identity	report	09	disease	turn
week	latest	opens	woman	myspaces	card	serving	week	symptoms	attack

B. Experiment Setup

When learning word embeddings from Wikipedia, we set the dimensionality of word embeddings to 50, and the context window size to 12. This means when we are predicting the current word, its previous 6 words and subsequent 6 words contribute to the prediction. We train word embeddings with an iteration of 100 epochs.

As there are 10 categories in our news dataset, we are interested to see if the extracted topics can reveal a similar mixture. Hence we set the number of topics K to 10. For uniformity, all the baseline topic models are implemented with Gibbs sampling as well and we perform 100 iterations of Gibbs sampling for all the models. We set $\eta = 20$, $\beta = 0.01$, and $\alpha = 50/K$. For the hyper parameters regarding Gaussian topics, we set prior $\boldsymbol{\mu}_0$ to the sample mean of all the word embeddings, the initial degree of freedom ν to the dimensionality of word embeddings, and assign an identity matrix to prior $\boldsymbol{\Psi}$.

As our *abcnews* dataset and Wikipedia are two different corpora, it's inevitable to encounter out-of-vocabulary words when extracting topics for *abcnews* dataset, i.e., some words in *abcnews* do not have corresponding word embeddings learnt from Wikipedia. We generate their word embeddings using the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Psi})$.

C. Experimental Results on Topic Coherence

Usually perplexity is used as a measure to evaluate language models. But in our case, the probability of a word embedding is given by its probability density function rather than an exact probability. Furthermore, the probability of a background word is given by the discrete multinomial probability with respect to the background mode, and this disagreement between continuous probability density and discrete probability makes it incorrect and in fact infeasible to use perplexity in our analysis. Thus, we list top 10 words in each topic for LDA, Gaussian-LDA and our model on *abcnews* dataset with $K = 10$, as shown in Tables I, II and

III. The words are ranked based on their frequency in each topic in the last round of sampling.

From Table I, we can see that the topics extracted by LDA are not satisfying, and this is probably because of the limitation of document length. Only topic 6 and topic 8 are high-quality topics corresponding to Travel and Politics. Topic 1 looks to be loosely related to Law but with Politics mixed in. The other topics are not at all acceptable.

In Table II, with the help of Wikipedia word embeddings, the extracted topics get more coherent. This validates the use of word embeddings in topic modelling. However, one can observe that the topics are still not as crisp as the ones we want. This is because of the fact that this model still treats the text as a mixture of many topics rather than a single topic.

Lastly we present the top words for our model - Table III. As one can observe many news categories, Politics (topic 1), Entertainment (topic 3), Sports (topic 4), Technology (topic 5), Travel (topic 7), Health (topic 9) and International (topic 10), are successfully extracted with high quality keywords. It is worth noticing that Entertainment, Technology and Health have not been extracted by any of the baseline topic models, and that our model even captures the names of entertainers in topic 3. Also we can still tell that topic 2 is somewhat related to Law, and topic 6 to Money.

D. Quality of Topic Representation of Documents

We see that the topics extracted by our model are more reasonable and have better qualities. But are the topics extracted by our model really corresponding to the coherent news categories? To answer this, we compare the category labels with the document-topic labels to see if they are consistent. The category label of each news article comes from the dataset and is used as the ground truth. The document-topic label of each news article is assigned by the models. More specifically, for LDA and Gaussian-LDA , we can assign one single topic to document d according to:

$$z_d = argmax_z p(z|d).$$

For our model, each document has only one topic according to the model assumptions. To solve the cluster matching problem, e.g., news category 1 may correspond to topic 9 instead of topic 1, we use pairwise comparison [15] to measure the consistency between news categories and extracted topic representation of documents. The pairwise comparison defined as:

$$precision(E, G) = \frac{||pair_E \cap pair_G||}{||pair_E||},$$

$$recall(E, G) = \frac{||pair_E \cap pair_G||}{||pair_G||},$$

$$F1(E, G) = \frac{2 \times precision \times recall}{precision + recall},$$

Table IV
COMPARISON OF DOCUMENT-TOPIC DISTRIBUTION

Model	Precision	Recall	F1
LDA	0.162	0.163	0.163
Gaussian-LDA	0.117	0.140	0.128
Our Model	**0.223**	**0.271**	**0.244**

where E and G are two clustering results corresponding to ten document-topic groups and ten ground truth categories respectively in our case, and $pair_E$ denotes the set of pairs in clustering result E. The result of this comparison is reported in Table IV. We can see that, with respect to the consistency between news categories and extracted topics, our model outperforms the other baselines significantly.

IV. Related Works

Topic models have been proposed to reveal the latent semantic structure from text corpus. Latent Semantic Analysis (LSA) [16] first tries to uncover the latent semantic information in a corpus by applying singular value decomposition to the document-term matrix. Probabilistic Latent Semantic Analysis (PLSA) [7] and Latent Dirichlet Allocation (LDA) [8] further use a hidden topic variable to capture the latent semantic structure and model documents as mixtures of topics, while topics are probability distributions over words. PLSA, LDA and their variants, such as the author topic model [17], have achieved huge success in analyzing normal texts. However, for short texts, such as tweets and news titles, conventional topic models usually don't work well due to the lack of rich context.

An intuitive way to handle this problem in short text corpus is to aggregate several short texts into one normal document based on auxiliary information before extracting topics. For instance, Weng et al. [18] utilize the user information of Twitter. They make an assumption similar to the author-topic model [17] that each user has a specific topic preference and then aggregate the tweets by the same user into one long document. Such aggregation methods can alleviate the lack of rich context problem and improve the performance of conventional topic models. However, such heuristic aggregation methods do not work in the scenarios where auxiliary information is not available. Take news titles as an example - there is no such auxiliary information as user name to utilize. Besides the assumption on user topic distribution, making assumption on the data is another way to tackle this problem. Zhao et al. [11] follow the assumption that a single tweet is usually about one single topic and further models each tweet as a variant of mixutre of unigrams. In [19], rather than each short document, they assume each sentence is about one topic.

However, these aforementioned models fail to leverage external semantics, which is quite helpful in dealing with the lack of rich context in short text corpus. Das et al. [14] first tries to combine topic modeling and word embeddings for regular texts, and further introduces a fast training method for it. Focusing on short texts, our model proposes a novel generative strategy utilizing word embeddings – modeling each short text as one single Gaussian distribution over topic words and complementing continuous Gaussian topics with discrete multinomial background model. The word embeddings we used in our model is derived from the language models based on distributed representations of words. Those language models are mostly built on neural network structures. The distributed representation of words, i.e., word embedding, is first introduced into natural language processing by NPLM [9]. Many distributed language models with speed-up strategies, such as using tree structures, have been proposed to reduce the time complexity of NPLM [20], [10]. Mikolov et al. [10] proposed a Huffman tree based hierarchical neural network called Word2Vec, which significantly shortens the training time and is one of the most popular distributed language models currently in use.

V. Conclusions

In this work, we have proposed a topic model for short texts using word embeddings. Word embeddings learnt from external sources, such as Wikipedia, can bring supplemental semantics to short texts to overcome its lack of rich context. Hence, we model each short document as a Gaussian topic in the word embedding vector space. A short text is composed of not only topic words but also background words, we incorporate an alternative background mode to complement Gaussian topics. Considering that background words are not semantically related, background mode is implemented with discrete multinomial distribution over word types rather than in the word embedding space. The experiments validate the effectiveness of our model at discovering coherent topics from short text corpus.

Acknowledgment

This work was supported in part by the US National Science Foundation under grants NSF IIS-1218393, NSF IIS 1319973 and IIS-1514204. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] J. Gantz and D. Reinsel, "The digital universe decade-are you ready," *IDC iView*, 2010.

[2] K. Lee, D. Palsetia, R. Narayanan, M. M. A. Patwary, A. Agrawal, and A. Choudhary, "Twitter trending topic classification," in *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 251–258.

[3] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann, "Who, where, when and what: discover spatio-temporal topics for twitter users," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 605–613.

[4] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. J. Gao, H. Qu, and X. Tong, "Textflow: Towards better understanding of evolving topics in text," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 17, no. 12, pp. 2412–2421, 2011.

[5] M. Chen, X. Jin, and D. Shen, "Short text classification improved by learning multi-granularity topics," in *IJCAI*. Citeseer, 2011, pp. 1776–1781.

[6] Q. Diao, J. Jiang, F. Zhu, and E.-P. Lim, "Finding bursty topics from microblogs," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 2012, pp. 536–544.

[7] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[8] 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.

[9] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain, "Neural probabilistic language models," in *Innovations in Machine Learning*. Springer, 2006, pp. 137–186.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[11] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li, "Comparing twitter and traditional media using topic models," in *Advances in Information Retrieval*. Springer, 2011, pp. 338–349.

[12] H. M. Wallach, "Topic modeling: beyond bag-of-words," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 977–984.

[13] P. Hu, W. Liu, W. Jiang, and Z. Yang, "Latent topic model based on gaussian-lda for audio retrieval," in *Pattern Recognition*. Springer, 2012, pp. 556–563.

[14] R. Das, M. Zaheer, and C. Dyer, "Gaussian lda for topic models with word embeddings," in *Proceedings of the 53nd Annual Meeting of the Association for Computational Linguistics*, 2015.

[15] D. Menestrina, S. E. Whang, and H. Garcia-Molina, "Evaluating entity resolution results," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 208–219, 2010.

[16] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, p. 391, 1990.

[17] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 2004, pp. 487–494.

[18] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 261–270.

[19] A. Gruber, Y. Weiss, and M. Rosen-Zvi, "Hidden topic markov models," in *International conference on artificial intelligence and statistics*, 2007, pp. 163–170.

[20] N. Djuric, H. Wu, V. Radosavljevic, M. Grbovic, and N. Bhamidipati, "Hierarchical neural language models for joint representation of streaming documents and their content," in *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2015, pp. 248–255.