Collaboratively Improving Topic Discovery and Word Embeddings by Coordinating Global and Local Contexts

Guangxu Xun[1], Yaliang Li[1,2], Jing Gao[1], Aidong Zhang[1]
[1]Department of Computer Science and Engineering, SUNY at Buffalo, New York, USA
[2]Baidu Research Big Data Lab, Sunnyvale, CA USA
[1]{guangxux, yaliangl, jing, azhang}@buffalo.edu, [2]yaliangli@baidu.com

ABSTRACT

A text corpus typically contains two types of context information – global context and local context. Global context carries topical information which can be utilized by topic models to discover topic structures from the text corpus, while local context can train word embeddings to capture semantic regularities reflected in the text corpus. This encourages us to exploit the useful information in both the global and the local context information. In this paper, we propose a unified language model based on matrix factorization techniques which 1) takes the complementary global and local context information into consideration simultaneously, and 2) models topics and learns word embeddings collaboratively. We empirically show that by incorporating both global and local context, this collaborative model can not only significantly improve the performance of topic discovery over the baseline topic models, but also learn better word embeddings than the baseline word embedding models. We also provide qualitative analysis that explains how the cooperation of global and local context information can result in better topic structures and word embeddings.

CCS CONCEPTS

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

KEYWORDS

Topic modeling, word embeddings, global context, local context, unified language model

ACM Reference format:
Guangxu Xun, Yaliang Li, Jing Gao, and Aidong Zhang. 2017. Collaboratively Improving Topic Discovery and Word Embeddings by Coordinating Global and Local Contexts. In *Proceedings of KDD '17, August 13-17, 2017, Halifax, NS, Canada,* 9 pages.
DOI: 10.1145/3097983.3098009

1 INTRODUCTION

Topic models [2, 6, 11, 29] and word embedding models [1, 4, 20] are two of the most successful and prevalent language models nowadays. They model languages from two different but complementary points of view — the global viewpoint and the local viewpoint. Topic models, such as Probabilistic Latent Semantic Analysis (PLSA) [11] and Latent Dirichlet Allocation (LDA) [2], are usually built upon the document-level global context information in a text corpus. Topic models follow the bag-of-word assumption that a document is represented as a bag of its words (disregarding grammar and even word order, but keeping multiplicity). Documents are modeled as mixtures of latent topics, where latent topics are formulated as multinomial distributions over words. Bayesian methods are then employed to infer the topic structures based on the global document-word frequency matrix of the corpus. In contrast to topic models utilizing the document-level global context information, most of the word embedding models, such as Neural Probabilistic Language Model (NPLM) [1] and Skip-Gram [20], are based on the local context information. Word embedding models follow the distributional hypothesis [9] that words occurring in similar local contexts tend to have similar syntactic and semantic properties. Semantically related words ought to be projected close to each other in the word embedding space. Word embeddings can then be constructed using internal representations from neural network architectures of local word sequences.

While local context can help disambiguate word meanings, global context can also provide useful topical information. Therefore, it is natural to expect more sufficient input information and better performance if a language model is able to utilize these two complementary context information collaboratively. In addition, both topic structures and word embeddings can be discovered from the corpus. However, it is difficult to develop a unified language model in depth which can absorb both the idea of globality from topic models and the idea of locality from word embedding models, because topic models are usually statistical generative models while word embedding models are mostly based on artificial neural networks.

Instead of developing a unified language model, researchers tend to combine global and local context by using the pre-trained result based on one type of context information to assist in modeling language on the other type of context information [5, 18, 27, 28]. For example, Gaussian-LDA [5] uses pre-trained word embeddings learned from large external corpora such as Wikipedia and then models topics with Gaussian distributions in the word embedding space; in contrast, Topical Word Embedding (TWE) [18] uses pre-trained topic structures to learn topic embeddings and improve word embeddings.

However, there are several limitations in combining topic models and word embedding models in this separate and heuristic manner. The first limitation stems from the difference of semantics in different datasets. One popular way to model topics upon word embeddings is to replace discrete word types in the target dataset (e.g. ESPN sports news) with continuous word embeddings learned from

KDD '17, August 13-17, 2017, Halifax, NS, Canada

978-1-4503-4887-4/17/08…$15.00
DOI: 10.1145/3097983.3098009

an external corpus (e.g. Wikipedia). But the difference of semantics in ESPN sports news and Wikipedia would probably result in poor performance. The second limitation is that external corpus is not always available. In some research domains such as biomedicine, there are no such knowledge bases as comprehensive as Wikipedia. However, if we want to directly train word embeddings on the target dataset and the target dataset is relatively small, we have to face the third limitation, i.e., the lack of local context information, as training word embeddings typically requires a large amount of local contexts.

The aforementioned limitations inspire us to develop a unified language model which is able to make use of both the global and the local context information collaboratively. We propose to unify the process of modeling topics and learning word embeddings via matrix factorization, and take advantage of both the idea of globality from topic models and the idea of locality from word embedding models. The new model is named Collaborative Language Model (CLM). In CLM, the global context information is encoded in the document-word matrix and the local context information is encoded in the word co-occurrence matrix. In addition to topic structures and word embeddings, we also introduce topic embeddings for topics and assume that the importance of a word in a topic is proportional to the inner product value of the corresponding word embedding and topic embedding. By fully exploiting the context information in a text corpus, CLM has the following advantages:

- CLM is able to discover topic structures and learn word embeddings collaboratively.
- CLM does not rely on pre-trained topic structures or pre-trained word embeddings learned from external corpora.
- When the text corpus is not large enough, with the help of global context information, CLM can overcome the lack of local context information and learn good word embeddings.
- With the help of local context information, CLM can discover more coherent latent topics.

To evaluate how well CLM discovers topics and learns word embeddings, we perform four quantitative evaluation tasks including two topic structure evaluation tasks and two word embedding evaluation tasks. We show that by taking both global and local context information into consideration, CLM outperforms the baselines on both the topic structure evaluation tasks and the word embedding evaluation tasks. We also provide qualitative assessment and case studies to explain how topic structures and word embeddings can collaboratively enhance the quality of each other.

2 RELATED WORK

Topic models are a powerful unsupervised tool to reveal the latent semantic structure from a text corpus based on its global document-word context information. Latent Semantic Analysis (LSA) [6] is proposed as a dimensionality reduction technique by projecting the document-word matrix to a linear subspace with Singular Value Decomposition (SVD). PLSA [11] introduces latent variables which can be viewed as 'topics' between documents and words, where each document is a multinomial distribution over topics and each topic is also a multinomial distribution over words. LDA [2] further extends PLSA to a complete probabilistic model by adding Dirichlet priors at the document level. Non-negative Matrix Factorization (NMF) [14] is a useful matrix decomposition technique for multivariate data and its non-negativity makes the resulting matrices easy to explain in many application domains. Ding et al. [7] have proven the equivalence between PLSA and NMF as they optimize the same objective function of the global document-word matrix.

Word embeddings, also known as word vectors and distributed representations of words, have proven to be able to capture semantic regularities in language by learning the local word co-occurrence context information. Specifically, NPLM [1] first introduces word embeddings into natural language processing. Many variants have been proposed since then to improve the efficiency of NPLM [4, 20]. In particular, the popular Continuous Bag Of Words (CBOW) and Skip-Gram models proposed by Mikolov et al. [20] are efficient to train and obtain state-of-the-art results on various linguistic tasks. The training methods of CBOW and Skip-Gram are highly popular, but not well understood until Levy et al. [15] proved that Skip-Gram with negative sampling training method is implicitly factorizing the pointwise mutual information (PMI) matrix of the local word co-occurrence patterns.

In order to make use of both the global context and the local context information, many composite models have been proposed to combine topic models and word embedding models. One common way is to use pre-trained word embeddings and replace the multinomial distribution over words with a probability function defined in the word embedding space to generate a focus word given its topic and neighboring words. Among them, Latent Feature Topic Modeling (LFTM) [23] defines the probability function as a mixture of the conventional multinomial distribution and a link function between the embeddings of the focus word and topics. TopicVec [17] adds context word embeddings to the link function in addition to the focus word embeddings and topic embeddings. Gaussian-LDA [5] models topics as Gaussian distributions over the continuous word embeddings. The other common way to combine topic models and word embedding models is to use pre-trained LDA topic structures to learn topic embeddings and assist in training word embeddings. For example, Topic2Vec [24] treats the pre-trained topic labels as special words and learns embeddings for topics by including the topic labels in the neural network architecture. Topical Word Embedding (TWE) [18] further concatenates the topic embedding with the word embedding to form the topical word embedding for each word.

However, all of these composite models combine topic models and word embedding models in a separate and heuristic manner – they either utilize pre-trained word embeddings or pre-trained topic structures. In contrast, our CLM model proposes to make the topic model and the word embedding model work collaboratively, and fully exploit the complementary global and local context information in a text corpus. Huang et al. [12] and Le et al. [13] propose to incorporate global information to help the learning of word embeddings by assigning an embedding to each document. Their ideas can be viewed as special cases of our model, with the number of topics set to the number of documents.

3 NOTATIONS AND DEFINITIONS

Table 1 shows the notations used in this paper. We use bold uppercase letters such as $\boldsymbol{D}$ to represent matrices, bold lowercase

Table 1: Table of Notations

Notation	Meaning
V	Vocabulary size
N	Number of documents
K	Number of topics
M	Dimensionality of the embedding space
$\boldsymbol{D} \in \mathbb{R}^{V \times N}$	Document-word matrix $[\boldsymbol{d}_1, ..., \boldsymbol{d}_N]$
$\boldsymbol{W} \in \mathbb{R}^{V \times V}$	Word co-occurrence matrix
$\boldsymbol{T} \in \mathbb{R}^{K \times V}$	Topic-word matrix $[\boldsymbol{t}_1, ..., \boldsymbol{t}_V]$
$\boldsymbol{\Theta} \in \mathbb{R}^{K \times N}$	Document-topic matrix $[\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_N]$
$\boldsymbol{A} \in \mathbb{R}^{M \times K}$	Topic embedding matrix $[\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_K]$
$\boldsymbol{B} \in \mathbb{R}^{M \times V}$	Word embedding matrix $[\boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_V]$
$\boldsymbol{C} \in \mathbb{R}^{M \times V}$	Context word embedding matrix $[\boldsymbol{c}_1, ..., \boldsymbol{c}_V]$
$\boldsymbol{d_n}$	The n^{th} document
$\boldsymbol{\theta_n}$	Topic representation for the n^{th} document
$\boldsymbol{t_v}$	Topic distribution for the v^{th} word
$\boldsymbol{\alpha}_k$	Topic embedding for the k^{th} topic
$\boldsymbol{\beta_v}$	Word embedding for the v^{th} word
$\boldsymbol{c_v}$	Context embedding for the v^{th} word
d_{ij}, w_{ij}, t_{ij}	The ij^{th} entry in matrix $\boldsymbol{D}, \boldsymbol{W}, \boldsymbol{T}$ respectively

letters such as $\boldsymbol{d_n}$ to represent vectors or embeddings, regular uppercase letters such as V to represent scalar constants, and regular lowercase letters such as d_{ij} to represent scalar variables.

Given a text corpus, its document-level global context information is encoded in the document-word matrix $\boldsymbol{D}$ and its local context information is encoded in the word co-occurrence matrix $\boldsymbol{W}$. The word co-occurrence matrix $\boldsymbol{W}$ is constructed from small fixed-sized text intervals in the documents. Each text interval is composed of a focus word and its neighboring context words falling in a fix-sized window centered at the focus word. The value of entry w_{ij} is the number of times that a context word w_j appears in word w_i's contexts.

Given the global context matrix $\boldsymbol{D}$ and the local context matrix $\boldsymbol{W}$, our goal is to discover topic structures and learn word embeddings collaboratively based on both context information.

4 METHODOLOGY

Our CLM model follows three basic assumptions: (1) each document focuses on only a small amount of topics and each topic assigns high probability to only a small number of words; (2) words appearing in similar local context tend to have similar syntactic and semantic properties and should be mapped to nearby areas in the embedding space; and (3) words close to each other in the embedding space tend to have similar topic distributions and vice versa.

We will introduce our CLM model according to how CLM is formulated based on these assumptions as well as how CLM utilizes the global and the local context information.

4.1 Our Proposed Model

The three aforementioned assumptions correspond to three building blocks of CLM. We first introduce the three building blocks. Then we describe our proposed model CLM and its relationship with other existing composite models.

4.1.1 Exploiting global context information. Given the global document-word matrix $\boldsymbol{D}$, NMF decomposes it into the product of document-topic matrix $\boldsymbol{\Theta}$ and topic-word matrix $\boldsymbol{T}$. The non-negativity of NMF ensures the explainability of document-topic distribution and topic-word distribution. The objective function to factorize $\boldsymbol{D}$ with regularization is:

$$\begin{aligned} L_{glo} &= ||\boldsymbol{D} - \boldsymbol{T}^T\boldsymbol{\Theta}||_2^2 + \lambda_s||\boldsymbol{\Theta}||_2^2 + \lambda_s||\boldsymbol{T}||_2^2, \\ &\textit{subject to}: \\ &\boldsymbol{\Theta} \geq 0 \textit{ and } \boldsymbol{T} \geq 0, \end{aligned} \tag{1}$$

where $||\boldsymbol{\Theta}||_2^2$ denotes the l_2 norm regularization we use on document-topic matrix $\boldsymbol{\Theta}$, $||\boldsymbol{T}||_2^2$ denotes the l_2 norm regularization we use on topic-word matrix $\boldsymbol{T}$, and λ_s is the parameter to prevent overfitting: the larger value of λ_s, the larger amount of shrinkage on $\boldsymbol{\Theta}$ and $\boldsymbol{T}$. In our model, instead of the raw frequency matrix, we use the document-word matrix with TF-IDF weights as $\boldsymbol{D}$.

4.1.2 Exploiting local context information. Based on the local context information, word embedding models can learn a low-dimensional representation for each word. In the Skip-Gram model [20], the objective of each training step is to predict neighboring words within a fixed window given a focus word. Stochastic gradient descent with negative sampling is a regular way to train Skip-Gram. Levy et al. [15] have proven an equivalence between Skip-Gram trained with negative sampling value of k and the factorization of the positive PMI word co-occurrence matrix shifted by $\log k$, i.e., the shifted positive pointwise mutual information matrix (SPPMI). Therefore, in our model, instead of the raw frequency matrix, we use the SPPMI word co-occurrence matrix as $\boldsymbol{W}$.

The PMI value between a pair of discrete outcomes x and y is defined as:

$$PMI(x, y) = \log \frac{P(x, y)}{P(x)P(y)}.$$

Empirically, the PMI value between a word w and its context word c can be estimated by considering the actual number of their co-occurrence times in the corpus:

$$PMI(w, c) = \log \frac{\#(w, c) \cdot E}{\#(w) \cdot \#(c)},$$

where $\#(w, c)$ is the number of times words w and c co-occur, $\#(w) = \sum_c \#(w, c)$, $\#(c) = \sum_w \#(w, c)$, and E is the total number of word-context pairs. The SPPMI matrix is then constructed as:

$$SPPMI_k(w, c) = \max(PMI(w, c) - \log k, 0).$$

Following the similar idea, CLM exploits local context information and learns word embeddings by factorizing matrix $\boldsymbol{W}$:

$$L_{loc} = ||\boldsymbol{W} - \boldsymbol{B}^T\boldsymbol{C}||_2^2 + \lambda_s||\boldsymbol{B}||_2^2 + \lambda_s||\boldsymbol{C}||_2^2. \tag{2}$$

4.1.3 Collaboration. By exploiting the global context information of a text corpus, we can discover the topic structures; and by exploiting the local context information, we can learn the word embeddings. However, these two parts should not be isolated from each other: semantically related words usually belong to similar topics and they are also close to each other in the embedding space. Hence we assume that the distances between word embeddings correlate with their topical similarities. As with exploiting the local context information, we realize this assumption by factorizing

topic-word matrix $\boldsymbol{T}$ into the product of topic embedding matrix $\boldsymbol{A}$ and context word embedding matrix $\boldsymbol{C}$:

$$L_{com} = ||\boldsymbol{T} - \boldsymbol{A}^T\boldsymbol{C}||_2^2 + \lambda_s||\boldsymbol{A}||_2^2 + \lambda_s||\boldsymbol{C}||_2^2. \tag{3}$$

Hence the probability of word w being grouped into topic z can be measured by the inner product of the corresponding topic embedding and word embedding: $p(z|w) \propto t_{zw} = \boldsymbol{\alpha}_z^T\boldsymbol{c}_w$. Therefore, besides achieving the topic embeddings, Eq. 3 also regulates words with similar topic distributions to be close in the embedding space and nearby words to have similar topic distributions.

4.1.4 Unifying the three assumptions. Both the global context information and the local context information contain useful patterns in a text corpus. We propose to utilize both types of context information jointly, and to discover topic structures and learn word embeddings collaboratively:

$$\begin{aligned} L = &\underbrace{\lambda_d||\boldsymbol{D} - \boldsymbol{T}^T\boldsymbol{\Theta}||_2^2}_{global} + \underbrace{\lambda_w||\boldsymbol{W} - \boldsymbol{B}^T\boldsymbol{C}||_2^2}_{local} + \underbrace{||\boldsymbol{T} - \boldsymbol{A}^T\boldsymbol{C}||_2^2}_{joint} \\ &+ \underbrace{\lambda_s||\boldsymbol{\Theta}||_2^2 + \lambda_s||\boldsymbol{T}||_2^2 + \lambda_s||\boldsymbol{A}||_2^2 + \lambda_s||\boldsymbol{B}||_2^2 + \lambda_s||\boldsymbol{C}||_2^2}_{regularization}, \\ &subject\ to: \\ &\boldsymbol{\Theta} \geq 0\ and\ \boldsymbol{T} \geq 0, \end{aligned} \tag{4}$$

where λ_d and λ_w are the parameters controlling the weights of the global and local modeling parts. Eq. 4 is the objective function of our model. Topic-word distribution matrix $\boldsymbol{T}$ is shared by both the global modeling and the joint modeling parts of the objective function. Context word embedding matrix $\boldsymbol{C}$ is shared by both the local modeling and the joint modeling parts of the objective function. Therefore, the topic structures and word embeddings we obtained must account for both the global and the local context information of a text corpus.

4.1.5 Relationship with other composite models. As we discussed in Section 2, two popular ways to combine topic models and word embedding models are 1) modeling topics based on pre-trained word embeddings and 2) learning word embeddings based on pre-trained topic structures. These models can be viewed as special cases of CLM that keep either topic structures or word embeddings fixed. More specifically, if we use pre-trained word embeddings and keep them fixed, then CLM considers only the combination of Eq. 1 and Eq. 3. This results in CLM becoming functionally equivalent to the composite models that discover topic structures with the help of pre-trained word embeddings, such as Gaussian-LDA [5] and TopicVec [17]. In contrast, if we use pre-trained topic structures and keep them fixed, then CLM considers only the combination of Eq. 2 and Eq. 3. Again CLM becomes functionally equivalent to the composite models that learn word embeddings and topic embeddings with the help of pre-trained topic structures, such as TWE [18] and Topic2Vec [24].

4.2 Parameter Inference

In this subsection, we will introduce how to do parameter estimation and inference for our proposed CLM model via collective matrix factorization. First the objective function of CLM in Eq. 4 is expanded as:

$$\begin{aligned} L =&\lambda_d \sum_{v=1,n=1}^{V,N}(d_{vn} - \boldsymbol{t}_v^T\boldsymbol{\theta}_n)^2 + \lambda_w \sum_{v=1,v'=1}^{V,V}(w_{vv'} - \boldsymbol{\beta}_v^T\boldsymbol{c}_{v'})^2 \\ &+ \sum_{k=1,v=1}^{K,V}(t_{kv} - \boldsymbol{\alpha}_k^T\boldsymbol{c}_v)^2 + \lambda_s\sum_{n=1}^{N}(\boldsymbol{\theta}_n^T\boldsymbol{\theta}_n) + \lambda_s\sum_{v=1}^{V}(\boldsymbol{t}_v^T\boldsymbol{t}_v) \\ &+ \lambda_s\sum_{k=1}^{K}(\boldsymbol{\alpha}_k^T\boldsymbol{\alpha}_k) + \lambda_s\sum_{v=1}^{V}(\boldsymbol{\beta}_v^T\boldsymbol{\beta}_v) + \lambda_s\sum_{v=1}^{V}(\boldsymbol{c}_v^T\boldsymbol{c}_v), \\ &subject\ to: \\ &\boldsymbol{\Theta} \geq 0\ and\ \boldsymbol{T} \geq 0. \end{aligned} \tag{5}$$

Then we compute the gradient of our objective function Eq. 5 with respect to each vector $\{\boldsymbol{\theta}_{1:N}, \boldsymbol{t}_{1:V}, \boldsymbol{\alpha}_{1:K}, \boldsymbol{\beta}_{1:V}, \boldsymbol{c}_{1:V}\}$:

$$\begin{aligned} \frac{\partial L}{\partial \boldsymbol{\theta}_n} =&2\lambda_d\sum_{v=1}^{V}(d_{vn} - \boldsymbol{t}_v^T\boldsymbol{\theta}_n)(-\boldsymbol{t}_v) + 2\lambda_s\boldsymbol{\theta}_n \\ \frac{\partial L}{\partial \boldsymbol{t}_v} =&2\lambda_d\sum_{n=1}^{N}(d_{vn} - \boldsymbol{t}_v^T\boldsymbol{\theta}_n)(-\boldsymbol{\theta}_n) + 2(\boldsymbol{t}_v - \boldsymbol{A}^T\boldsymbol{c}_v) + 2\lambda_s\boldsymbol{t}_v \\ \frac{\partial L}{\partial \boldsymbol{\alpha}_k} =&2\sum_{v=1}^{V}(t_{kv} - \boldsymbol{\alpha}_k^T\boldsymbol{c}_v)(-\boldsymbol{c}_v) + 2\lambda_s\boldsymbol{\alpha}_k \\ \frac{\partial L}{\partial \boldsymbol{\beta}_v} =&2\lambda_w\sum_{v'=1}^{V}(w_{vv'} - \boldsymbol{\beta}_v^T\boldsymbol{c}_{v'})(-\boldsymbol{c}_{v'}) + 2\lambda_s\boldsymbol{\beta}_v \\ \frac{\partial L}{\partial \boldsymbol{c}_{v'}} =&2\lambda_w\sum_{v=1}^{V}(w_{vv'} - \boldsymbol{\beta}_v^T\boldsymbol{c}_{v'})(-\boldsymbol{\beta}_v) + 2\lambda_s\boldsymbol{c}_{v'} \\ &+ 2\sum_{k=1}^{K}(t_{kv'} - \boldsymbol{\alpha}_k^T\boldsymbol{c}_{v'})(-\boldsymbol{\alpha}_k) \end{aligned}$$

Similar to the Alternating Least Squares (ALS) matrix factorization method, we obtain the following closed-form updates by iteratively setting the gradient to zero:

$$\begin{aligned} \boldsymbol{\theta}_n =&(\lambda_d\sum_{v=1}^{V}\boldsymbol{t}_v\boldsymbol{t}_v^T + \lambda_s\boldsymbol{I})^{-1}(\lambda_d\sum_{v=1}^{V}d_{vn}\boldsymbol{t}_v) \\ \boldsymbol{t}_v =&(\lambda_d\sum_{n=1}^{N}\boldsymbol{\theta}_n\boldsymbol{\theta}_n^T + (1+\lambda_s)\boldsymbol{I})^{-1}(\lambda_d\sum_{n=1}^{N}d_{vn}\boldsymbol{\theta}_n + \boldsymbol{A}^T\boldsymbol{c}_v) \\ \boldsymbol{\alpha}_k =&(\sum_{v=1}^{V}\boldsymbol{c}_v\boldsymbol{c}_v^T + \lambda_s\boldsymbol{I})^{-1}(\sum_{v=1}^{V}t_{kv}\boldsymbol{c}_v) \\ \boldsymbol{\beta}_v =&(\sum_{v'=1}^{V}\boldsymbol{c}_{v'}\boldsymbol{c}_{v'}^T + \lambda_s\boldsymbol{I})^{-1}(\sum_{v'=1}^{V}w_{vv'}\boldsymbol{c}_{v'}) \\ \boldsymbol{c}_{v'} =&(\lambda_w\sum_{v=1}^{V}\boldsymbol{\beta}_v\boldsymbol{\beta}_v^T + \sum_{k=1}^{K}\boldsymbol{\alpha}_k\boldsymbol{\alpha}_k^T + \lambda_s\boldsymbol{I})^{-1} \\ &*(\lambda_w\sum_{v=1}^{V}w_{vv'}\boldsymbol{\beta}_v + \sum_{k=1}^{K}t_{kv'}\boldsymbol{\alpha}_k) \end{aligned} \tag{6}$$

Note that this update does not guarantee the non-negativity of $\boldsymbol{\theta}_n$ and $\boldsymbol{t}_v$. Since our objective function is continuous, the minimum should be either at the points where the gradient is zero or on

the boundary. Hence, if Eq. 6 assigns $\boldsymbol{\theta}_n$ and $\boldsymbol{t}_v$ negative entries, we can just set the negative entries to zeros. The main difference between our updates and ALS is that many variables are associated with more than one matrix factorization term. For example, context word embeddings $\boldsymbol{c}_v$ is associated with both local context matrix $\boldsymbol{W}$ and topic-word matrix $\boldsymbol{T}$. Iteratively performing these updates achieves a stationary point of our model's objective function L.

5 EXPERIMENTS

We carry out experiments on two real-world text corpora to demonstrate the efficacy of our CLM model in two aspects: modeling topics and learning word embeddings. To investigate the quality of the topic structures discovered by CLM, we compare its performance with existing topic modeling methods on two topic evaluation tasks, the topic coherence evaluation task and the document classification task. To investigate the quality of the word embeddings learned by CLM, we compare its performance with existing word embedding models on two word embedding evaluation tasks, the word similarity task and the word analogy task. Moreover, we provide case studies to show the advantages of exploiting both types of context information.

The 20 Newsgroups dataset[1] and the Reuters-21578 dataset[2] are used in our experiments. 20News contains approximately 20,000 newsgroup documents evenly partitioned into 20 different categories. Reuters contains about 10,000 documents, but the numbers of documents in each category are highly imbalanced. We select only the largest 8 categories in Reuters, leaving us with 7,674 documents in total. In the preprocessing step, stop words and words with total frequency lower than 10 get removed, and all words are converted to lowercase. When constructing the local context matrix $\boldsymbol{W}$, we set the context window size to 10, i.e., 5 preceding words and 5 following words are considered as local context words for a focus word. For the parameters controlling the weights and regularization, we set $\lambda_d = 1e-2$, $\lambda_w = 2e-2$, and $\lambda_s = 1e-7$.

The source code of our implementation is available at https://github.com/XunGuangxu/2in1.

5.1 Evaluation on Topic Coherence

5.1.1 Baselines and experimental settings. The topic modeling methods we use as our baselines are the vanilla LDA [2], NMF [14], PLSI [11], Gaussian-LDA [5] and LFTM [23], among which Gaussian-LDA and LFTM are composite topic models that are built upon pre-trained word embeddings. For 20News and Reuters, we set the number of topics K to 20 and 8, respectively, as there are 20 newsgroups and 8 categories. We set the number of iterations to 100 for all the methods. For LDA, we set the hyperparameters alpha to $50/K$ and beta to 0.01. For the sake of fairness, the word embeddings used in Gaussian-LDA and LFTM are trained on the same dataset using word2vec toolkit [20]. And we set the dimensionality of word embeddings to 50 for Gaussian-LDA, LFTM and CLM.

5.1.2 Evaluation metrics. In order to quantitatively assess the topic coherence, we adopt an automated metric, called coherence score of topics proposed by [22], which is able to automatically evaluate the coherence of each topic. Given a topic z and its top U words $V^z = \{v_1^z, v_2^z, ..., v_U^z\}$, the coherence score of this topic with respect to its top U words is defined as:

$$C(z; V^z) = \sum_{u=2}^{U} \sum_{l=1}^{u} \log \frac{D(v_u^z, v_l^z) + 1}{D(v_l^z)},$$

where $D(v_l^z)$ is the document frequency of word v_l^z and $D(v_u^z, v_l^z)$ is the number of documents in which words v_u^z and v_l^z co-occurred. The coherence score follows the intuition that top words in the same topic tend to frequently co-occur in documents. A topic coherence score closer to zero means a higher co-occurrence rate of the topic words, indicating a more coherent topic. This topic coherence score shows high consistency with human judgements on topic qualities [22]. In order to investigate the overall quality of the discovered topic set, we use the average topic coherence score: $\bar{C} = \frac{1}{K} \sum_z C(z; V^z)$.

Table 2: Topic Coherence Scores on 20News

Top U words	5	10	20	50
NMF	-18.051	-85.538	-417.199	-2796.776
PLSI	-15.151	-78.597	-365.693	-2684.952
LDA	-15.308	-80.482	-368.820	-2694.437
Gaussian-LDA	-19.450	-94.523	-435.903	-3407.968
LFTM	-16.589	-78.541	-385.734	-2807.011
CLM	**-11.624**	**-60.303**	**-282.799**	**-2275.523**

Table 3: Topic Coherence Scores on Reuters

Top U words	5	10	20	50
NMF	**-11.281**	-66.412	-335.619	-2705.525
PLSI	-13.226	-70.078	-333.570	-2767.808
LDA	-12.093	-69.806	-352.296	-2840.746
Gaussian-LDA	-24.223	-108.453	-478.433	-3688.172
LFTM	-13.268	-71.352	-369.009	-2982.395
CLM	-11.483	**-63.083**	**-313.459**	**-2683.163**

5.1.3 Experimental results. The topic coherence results of each method on 20News and Reuters are reported in Table 2 and Table 3, respectively. To make this evaluation more comprehensive, we vary the number of topic words $U = \{5, 10, 20, 50\}$. The best scores are highlighted in boldface. As generative models, PLSI and LDA achieve similar topic coherence scores. Gaussian-LDA does not perform well and this is probably because this topic coherence metric is more appropriate for measuring frequent words in a topic while Gaussian-LDA ranks words according to their Gaussian probabilities in each topic. LFTM outperforms Gaussian-LDA because LFTM takes advantage of both the conventional Dirichlet multinomial and the link function in the embedding space. As with CLM, NMF also factorizes the document-word matrix to learn topic structures, but our CLM model is able to utilize the additional semantic information in word embeddings learned from local context and this semantic information helps CLM discover more coherent topics. CLM ranks words in each topic according to their values in the topic-word matrix $\boldsymbol{T}$. We can see that CLM achieves significantly higher coherence scores than the baselines.

[1] http://qwone.com/ jason/20Newsgroups/
[2] http://www.daviddlewis.com/resources/testcollections/reuters21578/

Table 4: Document Classification on 20News

	Precision	Recall	F1
NMF	0.704	0.701	0.697
PLSI	0.722	0.712	0.709
LDA	0.727	0.722	0.719
Gaussian-LDA	0.309	0.265	0.227
LFTM	0.716	0.714	0.709
TWE	0.525	0.466	0.437
PV-DBOW	0.510	0.491	0.459
PV-DM	0.428	0.386	0.361
CLM	**0.825**	**0.818**	**0.816**

Table 5: Document Classification on Reuters

	Precision	Recall	F1
NMF	0.911	0.877	0.891
PLSI	0.919	0.896	0.906
LDA	0.888	0.870	0.879
Gaussian-LDA	0.462	0.315	0.353
LFTM	0.893	0.591	0.661
TWE	0.794	0.512	0.626
PV-DBOW	0.755	0.505	0.549
PV-DM	0.681	0.434	0.507
CLM	**0.944**	**0.916**	**0.929**

5.2 Evaluation on Document Classification

5.2.1 Baselines and experimental settings. In addition to the baselines used in Subsection 5.1, we also include TWE [18] and two Doc2Vec models [13]: PV-DBOW and PV-DM, as they can provide document-level representations for this document classification task. TWE is a composite model built upon pre-trained topic structures, so we feed the output of the vanilla LDA to TWE as the pre-trained topic structures. We keep the same experimental settings as in Subsection 5.1, except that, for 20News we set the number of topics to 280 for the topic models and the dimensionality of document embeddings to 280 for TWE, PV-DBOW and PV-DM, and for Reuters we set them to 110.

5.2.2 Evaluation metrics. In the document classification task on 20News, each newsgroup document is represented as a 280 dimensional vector. Hence, 20,000 newsgroup documents are classified into 20 classes according to their document-topic distributions or document embeddings. The reason why we change the number of topics from 20 to 280 is that the number of classes is already 20 and the number of features (topics) should be larger than that. Similarly, we set the number of features (topics) to 110 for Reuters. In order to evaluate the overall performance across all the document classes, we adopt the macro-averaged precision, recall and F1 measures as the evaluation metrics, as macro-averaging gives equal weight to each class.

5.2.3 Experimental results. Table 4 and Table 5 present the classification performance of the different methods on 20News and Reuters, respectively. The highest scores are highlighted in boldface. The document-topic representation used here corresponds to the document-topic matrix $\boldsymbol{\Theta}$ in our model. We can see that CLM outperforms the baselines significantly. On this task, PLSI, LDA and LFTM still obtain similar and better scores than the other baselines. As with CLM, NMF is also based on matrix factorization techniques, but NMF does not achieve as good performance as CLM due to its inability to utilize both context information. Gaussian-LDA performs considerably inferior to all other methods. By checking its output variables manually, we find that the Gaussian distributions for different topics are highly similar and hence its document-topic representations are not discriminative enough. TWE, PV-DBOW and PV-DM assign a low-dimensional embedding to each document based on the word embeddings in it, but the classification results on these document embeddings are inferior to the results on document topic proportions.

5.3 Evaluation on Word Similarity

Having shown the superiority of CLM in topic discovery, we now evaluate the quality of word embeddings learned from the 20 Newsgroups dataset by CLM in the following two tasks. As we know, training word embeddings requires a large amount of local context information to capture language regularities. Hence, Wikipedia, the largest online encyclopedia, is the most frequently used training dataset for word embeddings due to its sheer range of topics and ample local context information. However, for experiment domains involving smaller corpus size such as the 20 Newsgroups dataset, gathering the local context information is quite a challenge. We will show that our CLM model is able to overcome the challenge of lacking local context information by taking the complementary global context information into consideration.

5.3.1 Baselines and experimental settings. The word embedding methods we include as our baselines are the SPPMI matrix without dimensionality reduction [15], SVD of the SPPMI matrix [15], GloVe [25], CBOW [20], Skip-Gram [20], PV-DBOW [13], PV-DM [13] and TWE [18], among which TWE is a composite model that is built upon pre-trained LDA topic structures, and PV-DBOW and PV-DM take the influence of documents on word embeddings into consideration. Different from the others, GloVe constructs word co-occurrence matrix and learns word embeddings purely based on document-level global context information. For uniformity, we set the number of context window size to 10, the number of negative samples to 5, and the dimensionality of the embedding space to 50 for all the methods. And we set the number of topics to 20 for TWE and CLM. We then perform 100 iterations of training for all the methods.

5.3.2 Evaluation metrics. We use several test datasets to evaluate the word pair similarities calculated by word embeddings: WordSim353 (WS353) [8] (including WordSim Relatedness (WS Sim) and WordSim Similarity (WS Rel)), MEN [3], Turk [26], SimLex-999 [10], and Rare [19]. These datasets contain word pairs associated with human-assigned similarity scores. After ranking the word pairs according to their cosine similarities in the embedding space and human-assigned similarity scores respectively, the word embeddings are evaluated by measuring the Spearman's rank correlation with the human ratings. We exclude word pairs that contain out-of-vocabulary words from the test datasets. A higher correlation value indicates it is more consistent with human judgements on word similarities.

Table 6: Comparison of Word Similarity Results

	WS353	WS Rel	WS Sim	Men	Turk	SimLex-999	Rare
SPPMI	0.461	0.444	0.465	0.444	**0.551**	0.131	0.245
SPPMI + SVD	0.451	0.435	0.449	0.426	0.489	0.166	0.349
GloVe	0.300	0.279	0.320	0.192	0.268	0.049	0.230
Skip-Gram	0.492	0.479	0.473	0.456	0.512	0.155	0.407
CBOW	0.488	0.451	0.494	0.432	0.529	0.151	0.407
PV-DBOW	0.477	0.442	0.486	0.449	0.488	0.139	0.285
PV-DM	0.297	0.304	0.310	0.236	0.339	0.013	0.157
TWE	0.317	0.231	0.407	0.190	0.260	0.084	0.184
CLM	**0.526**	**0.486**	**0.550**	**0.477**	0.525	**0.189**	**0.411**

5.3.3 Experimental results. The results are summarized in Table 6. The highest correlation scores are highlighted in boldface. Similar performances are achieved by the SPPMI matrix and SVD of SPPMI; however, the dimensionality of SPPMI word representations is the vocabulary size – 20,678 – much higher than 50 dimensions for the other methods. As with CLM, SVD also learns word embeddings by factorizing the local SPPMI matrix, but its inability to utilize the additional global topical information results in inferior performance to ours. Skip-Gram and CBOW yield better results than SPPMI and SVD of SPPMI. PV-DBOW performs on a par with Word2Vec models. GloVe performs inferior to the other methods. This may be due to the fact that GloVe utilizes only global context information but there is inadequate global context information to train word embeddings in the 20 Newsgroups dataset. As one can see, the SPPMI matrix obtains the best correlation score on the Turk test dataset, and CLM outperforms the baselines on all the other test datasets.

5.4 Evaluation on Word Analogy

5.4.1 Baselines and experimental settings. For the second evaluation task on the quality of word embeddings, we use the same baselines and keep the same experimental settings as in the previous word similarity task in Subsection 5.3.

5.4.2 Evaluation metrics. The word analogy task refers to questions of the form "a is to a* as b is to b*", where b* is hidden and needs to be inferred from the vocabulary. We use two test datasets for the word analogy task: MSR [21], which contains 8000 morphosyntactic analogy questions, such as "good is to better as rich is to richer", and Google [20], which contains 19544 questions, about half of the same syntactic type as in MSR, and the other half of a semantic nature, such as "king is to queen as man is to woman". We filter out questions involving out-of-vocabulary words. The hidden words b* can be inferred by optimizing 3CosAdd [16]:

$$\arg\max_{b* \in V} (cos(b*, b - a + a*)).$$

The evaluation metric for the word analogy task is the percentage of questions for which the 3CosAdd result is the correct answer b*.

5.4.3 Experimental results. Table 7 shows the results on the word analogy task. As can be seen, the lack of local context information in the 20 Newsgroups dataset heavily limits the performance of the different methods on such a difficult task as word analogies. Linguistically speaking, the word analogy task relies more on contextual information from common words and auxiliary verbs to correctly infer b*. Word embeddings learned from larger dataset which provides sufficient local context such as Wikipedia can achieve better performance on this task. For word embeddings learned from the 20 Newsgroups dataset, SVD of the SPPMI matrix performs on a par with the raw SPPMI matrix. CBOW, Skip-Gram and PV-DBOW yield better results than SPPMI and SVD of SPPMI because their training procedures give more influence to frequent pairs. GloVe does not perform well since it only explores global context information. TWE does not achieve good results because it is heavily limited by the sparsity issue and influenced by the pre-trained topic structures. We can still see that, by exploiting both the global and the local context information, CLM overcomes the lack of local context information and outperforms the baselines significantly.

Table 7: Comparison of Word Analogy Results

	Google	MSR
SPPMI	6.60%	5.40%
SPPMI + SVD	4.93%	7.32%
GloVe	2.67%	3.51%
Skip-Gram	6.62%	10.70%
CBOW	5.61%	12.00%
PV-DBOW	7.12%	11.77%
PV-DM	2.84%	7.55%
TWE	3.76%	5.38%
CLM	**8.28%**	**14.20%**

5.5 Qualitative Assessment of Topic Embeddings

Besides topic structures and word embeddings, CLM can also learn topic embeddings for each topic, i.e., the topic embedding matrix $\boldsymbol{A}$. Those topic embeddings are of the same dimensionality as word embeddings. The relationships between topic embeddings and word embeddings are modeled in Eq. 3: the larger inner product value a word embedding and a topic embedding get, the more important that word is in the topic. After convergence, the similarities and correlations among topics are also captured in the embedding space. Figure 2 shows the two-dimensional PCA projection of the topic embeddings related to religions and Mideast. Each topic embedding is annotated with its topic name and top 5 words. We can observe that the semantic similarities between topics correlate with the

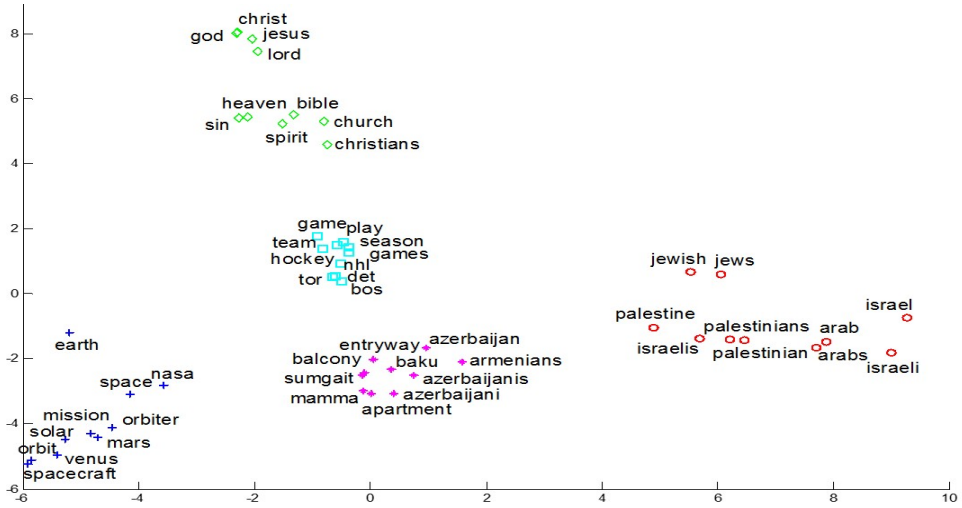

Figure 1: Two-dimensional PCA projection of word embedding clusters.

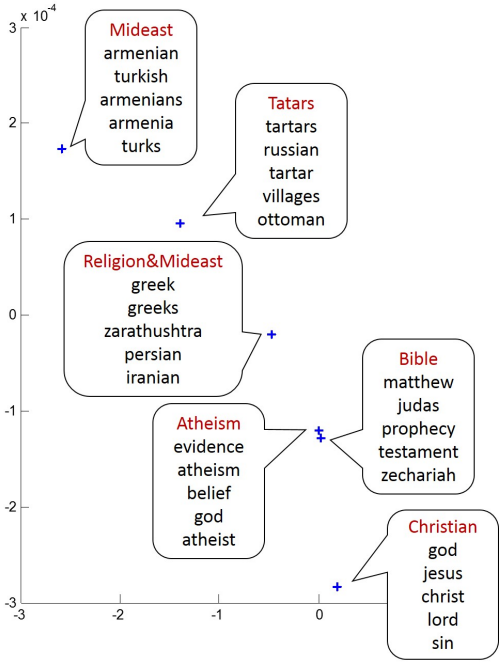

Figure 2: Two-dimensional PCA projection of the topic embeddings related to religions and mideast.

Euclidean distances between the corresponding topic embeddings. The correlations among topics can also be captured in this embedding space. For example, Figure 2 illustrates how the topic of Christian transitions to the topic of Mideast through the topics of Bible and religions.

5.6 Case Studies

5.6.1 How local context information assists global context information in discovering topic structures. Having shown the superiority of CLM in discovering topic structures in Subsections 5.1 and 5.2, we now take the topic of astronomy as an example to illustrate how word embeddings can help discover more coherent topics. Word embeddings learned from local context information are able to capture semantic regularities in language: words with similar semantic properties are found to be close to each other in the embedding space. And we are encouraged to group semantically related words (words that are geographically close in the embedding space) into same topics. This intuition is illustrated in Figure 1: words belonging to same topics tend to locate in nearby areas.

Table 8: Case Study 1

	Coherence score	Avg cosine distance
10 random words	-171.641	0.026
NMF	-102.422	0.570
CLM	**-89.731**	**0.728**

To verify our assumption, we quantitatively show that the average cosine distance of words in a topic is consistent with the topic coherence score. As our closest competitor in topic discovery, NMF is equivalent to our CLM model without considering word embeddings. The top 10 words in the topic of astronomy discovered by CLM are *{'space', 'orbit', 'solar', 'spacecraft', 'mission', 'mars', 'earth', 'venus', 'nasa', 'orbiter'}* as shown on the bottom left corner in Figure 1. The top 10 words in the topic of astronomy discovered by NMF are *{'space', 'earth', 'planet', 'system', 'spacecraft', 'solar', 'venus', 'surface', 'moon', 'kilometers'}*. We then calculate the average cosine distance of words and the topic coherence score for CLM and NMF respectively. Table 8 justifies the consistency between the topic coherence score and the average cosine distance. Therefore, by considering the spatial information of word embeddings, more coherent topics can be discovered by CLM.

5.6.2 How global context information assists local context information in learning word embeddings. In Subsections 5.3 and 5.4, we have shown that the word embeddings learned by CLM are closer to human judgements in terms of word similarities. We now take

Table 9: Case Study 2

Word pairs	Ground truth ranking	SPPMI +SVD ranking	CLM ranking	Cosine similarity of $p(z\|w)$
king – queen	13	135	118	0.733
money – currency	7	79	41	0.851
planet – space	42.5	201	143	0.918
mile – kilometer	12	94	62	0.879
man – woman	24	70	44	0.731

several word pairs in the WS353 test dataset to illustrate how global topical information can help us learn better word embeddings. We compare our rankings for these example words with the rankings of our closest competitor SPPMI+SVD which is equivalent to our CLM model without considering global topical information. As we can see in Table 9, due to the lack of sufficient local context information, these word pairs are not ranked properly by SPPMI+SVD. With the help of global topical information, CLM can improve the similarity ranking as words' topic distribution regulates. If two words have similar topic distributions (measured by the cosine similarity between their $p(z|w)$), such as planet and space, CLM would adjust the two corresponding word embeddings closer to each other accordingly and assign them a higher position in the similarity ranking.

6 CONCLUSION

We present a unified language model CLM based on matrix factorization techniques which is able to collaboratively discover topic structures and learn word embeddings. Moreover, building our model on both the global and the local context enables it to make use of more sufficient information. The proposed CLM model formulates documents as admixtures of topics, where each topic is a multinomial distribution over words and is influenced by word embeddings in the way that words close to each other in the embedding space should be grouped into same topics. At the same time, CLM also assumes that words appearing in similar local contexts and having similar topic distributions tend to get mapped to nearby areas in the embedding space. Topics and words are jointly trained and embedded in the vector space that preserves semantic regularities, while sparse and interpretable document-topic distributions are achieved simultaneously. The experiments on the real-world datasets validate the effectiveness of CLM.

ACKNOWLEDGMENTS

We thank Wayne Xin Zhao and Sheng Li for their help and useful discussions. This work was supported in part by the US National Science Foundation under grants NSF IIS-1218393, IIS-1514204 and IIS 1319973. 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] Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain. 2006. Neural probabilistic language models. In *Innovations in Machine Learning*. Springer, 137–186.

[2] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research* 3 (2003), 993–1022.

[3] Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 136–145.

[4] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research* 12 (2011), 2493–2537.

[5] Rajarshi Das, Manzil Zaheer, and Chris Dyer. 2015. Gaussian LDA for Topic Models with Word Embeddings.. In *ACL (1)*. 795–804.

[6] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41, 6 (1990), 391.

[7] Chris Ding, Tao Li, and Wei Peng. 2006. Nonnegative matrix factorization and probabilistic latent semantic indexing: Equivalence chi-square statistic, and a hybrid method. In *AAAI*, Vol. 6. 137–143.

[8] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*. ACM, 406–414.

[9] Zellig S Harris. 1954. Distributional structure. *Word* 10, 2-3 (1954), 146–162.

[10] Felix Hill, Roi Reichart, and Anna Korhonen. 2016. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics* (2016).

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

[12] Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. 2012. Improving word representations via global context and multiple word prototypes. In *ACL (1)*. 873–882.

[13] Quoc V Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents.. In *ICML*, Vol. 14. 1188–1196.

[14] Daniel D Lee and H Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*. 556–562.

[15] Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*. 2177–2185.

[16] Omer Levy, Yoav Goldberg, and Israel Ramat-Gan. 2014. Linguistic Regularities in Sparse and Explicit Word Representations.. In *CoNLL*. 171–180.

[17] Shaohua Li, Tat-Seng Chua, Jun Zhu, and Chunyan Miao. 2016. Generative topic embedding: a continuous representation of documents. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics (ACL)*.

[18] Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2015. Topical Word Embeddings.. In *AAAI*. 2418–2424.

[19] Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology.. In *CoNLL*. 104–113.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[21] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations.. In *Hlt-naacl*, Vol. 13. 746–751.

[22] David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 262–272.

[23] Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. 2015. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics* 3 (2015), 299–313.

[24] Liqiang Niu, Xinyu Dai, Jianbing Zhang, and Jiajun Chen. 2015. Topic2Vec: learning distributed representations of topics. In *Asian Language Processing (IALP), 2015 International Conference on*. IEEE, 193–196.

[25] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation.. In *EMNLP*, Vol. 14. 1532–1543.

[26] Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on World wide web*. ACM, 337–346.

[27] Guangxu Xun, Vishrawas Gopalakrishnan, Fenglong Ma, Yaliang Li, Jing Gao, and Aidong Zhang. 2016. Topic Discovery for Short Texts Using Word Embeddings. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 1299–1304.

[28] Guangxu Xun, Yaliang Li, Wayne Xin Zhao, Jing Gao, and Aidong Zhang. 2017. A Correlated Topic Model Using Word Embeddings. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*.

[29] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. 2011. Comparing twitter and traditional media using topic models. In *Advances in Information Retrieval*. Springer, 338–349.