Spherical Paragraph Model

Ruqing Zhang, Jiafeng Guo, Yanyan Lan, Jun Xu, and Xueqi Cheng

University of Chinese Academy of Sciences, Beijing, China
CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology,
`zhangruqing@software.ict.ac.cn`,
`{guojiafeng,lanyanyan,junxu,cxq}@ict.ac.cn`

Abstract Representing texts as fixed-length vectors is central to many language processing tasks. Most traditional methods build text representations based on the simple Bag-of-Words (BoW) representation, which loses the rich semantic relations between words. Recent advances in natural language processing have shown that semantically meaningful representations of words can be efficiently acquired by distributed models, making it possible to build text representations based on a better foundation called the Bag-of-Word-Embedding (BoWE) representation. However, existing text representation methods using BoWE often lack sound probabilistic foundations or cannot well capture the semantic relatedness encoded in word vectors. To address these problems, we introduce the Spherical Paragraph Model (SPM), a probabilistic generative model based on BoWE, for text representation. SPM has good probabilistic interpretability and can fully leverage the rich semantics of words, the word co-occurrence information as well as the corpus-wide information to help the representation learning of texts. Experimental results on topical classification and sentiment analysis demonstrate that SPM can achieve new state-of-the-art performances on several benchmark datasets.

Introduction

A central question to many language understanding problems is how to capture the essential meaning of a text in a machine-understandable format (*e.g.*, fixed-length vector representation). Most traditional methods either directly use the Bag-of-Words (BoW) representation [1], or built upon BoW using matrix factorization [2,3] or probabilistic topical models [4,5]. However, by using BoW as the foundation, rich semantic relatedness between words is lost. The text representation thus is learned purely based on the word-by-text co-occurrence information. However, humans understand a piece of text not solely based on its content (*i.e.*, the word occurrences), but also her background knowledge (*e.g.*, semantics of the words). Recent advances in the Natural Language Processing (NLP) community have shown that semantics of the words or more formally the distances between the words can be effectively revealed by distributed word representations [6], also referred to as "word embeddings" or "word vectors". Therefore, a natural idea is that one can build text representations based on a better foundation, namely the Bag-of-Word-Embeddings (BoWE) representation, by replacing distinct words with word vectors learned a priori with rich semantic relatedness encoded.

There have been some recent attempts to use BoWE for text representations. Perhaps the simplest way is to represent the text as a weighted average of all its word vectors [7]. Besides, [8] aggregated the word vectors into a text-level representation under the Fisher Kernel framework. Another well-known approach is the Paragraph Vector (PV) [9], which jointly learns the word and text representations as a direct optimization problem. There are several clear drawbacks with existing methods: (1) Existing methods often lack sound probabilistic foundations, making them heuristic or weak in interpretability; (2) All the methods assume the independency between texts, limiting their ability to leverage the corpus-wide information to help the representation learning of each piece of text. This limitation is analogous to that of Probabilistic Latent Semantic Indexing (PLSI) [4] in topic modeling, which has been addressed by Latent Dirichlet Allocation (LDA) [5]; (3) Simple weighted sum or aggregation using fisher kernel cannot well capture the semantic relatedness encoded in word vectors, which is typically revealed by the distance (or similarity) between word vectors.

To address these problems, we introduce a novel Spherical Paragraph Model (SPM), which learns text representations through modeling the generation of the corpus based on BoWE representations. Specifically, each piece of text is first represented as a bag of ℓ_2-normalized word vectors. Note that by normalization, the cosine similarity between word vectors are equal to the dot product between them, and all the word vectors lie on a unit hypersphere. We then assume the following generation process of the whole corpus. A text vector is first sampled from a corpus-wide prior distribution, and a word vector is then sampled from a text-level distribution given the text vector. The von Mises-Fisher (vMF) distribution [10] is employed for both corpus-wide and text-level distributions, which arises naturally for data distributed on the unit hypersphere and model the directional relation (*i.e.*, dot product) between vectors. The text representations can then be inferred by maximizing the likelihood of the generation of the whole corpus. We develop a variational EM algorithm to learn the SPM efficiently.

Compared with previous methods, SPM enjoys the following merits: (1) By modeling the generation process of the whole corpus based on BoWE, SPM can fully leverage the rich semantics of words, the word-by-text co-occurrences information as well as the corpus-wide information to help the representation learning of texts; (2) By employing the vMF distribution, SPM can well capture the semantic relatedness encoded in words vectors (*i.e.*, cosine similarity between word vectors); (3) SPM has good probabilistic interpretability as traditional topic models (*e.g.*, LDA), while allows unlimited hidden topics (*i.e.*, word clusters) as neural embedding models (*e.g.*, PV) by eliminating the topic layer.

We evaluated the effectiveness of our SPM by comparing with existing text presentation methods based on several benchmark datasets. The empirical results demonstrate that our model can achieve new state-of-the-art performances on several topical classification and sentiment analysis tasks.

Related Work

In this section, we briefly review researches related to our work, including existing text representation methods and text models using the vMF distribution.

Existing models for Texts

The most common fixed-length representation is Bag-of-Words (BoW) [1]. For example, in the popular TF-IDF scheme [11], each text is represented by *tfidf* values of selected feature-words. However, the BoW representation often suffers from data sparsity and high dimension. Meanwhile, by viewing each word as a distinct feature dimension, the BoW representation has very little sense about the semantics of the words.

To address this shortcoming, several dimensionality reduction methods have been proposed based on BoW, including matrix factorization methods such as LSI [2] and NMF [3], and probabilistic topical models such as PLSI [4] and LDA [5]. The key idea of LSI is to map texts to a vector space of reduced dimensionality (the latent semantic space), based on a Singular Value Decomposition over the term-document co-occurrence matrix. NMF is distinguished from the other methods by its non-negativity constraints, which leads to a parts-based representation because they allow only additive, not subtractive combinations. In PLSI, each word is generated from a single topic, and different words in a document may be generated from different topics. LDA is proposed by introducing a complete generative process over the documents, and demonstrated as a state-of-the-art document representation method. However, as built upon the BoW, all these methods do not leverage the rich semantics of the words, and learn text representations purely based on the word-by-text co-occurrence information.

Recent developments in distributed word representations have succeeded in capturing semantic regularities in language. Specifically, neural embedding models, *e.g.*, Word2Vec model [6] and Glove model [12], learn word vectors (also called word embeddings) efficiently from very large text corpus. The learned word vectors can reveal the semantic relatedness between words and perform word analogy tasks successfully.

With rich semantics encoded in word vectors, a natural question is how to obtain the text representation based on word vectors. A simple approach is to use a weighted average [8] or sum of all the word vectors. Besides, Fisher Vector (FV) [8] transforms the variable-cardinality word vectors into a fixed-length text representation based on the Fisher kernel framework [13]. However, these methods often lack sound probabilistic foundations. Meanwhile, simple weighted sum or aggregation using fisher kernel cannot well capture the semantic relatedness encoded in word vectors, which is typically revealed by the distance (or similarity) between word vectors. Later, Paragraph Vector (PV) which has two different model architectures (*i.e.*, PV-DM and PV-DBOW) [9] is introduced to jointly learn the word and text representations. Although these models seem to work well in practice, there is a strong independence assumption between texts in these methods, limiting their ability to leverage the corpus-wide information to help the representation learning of each piece of text.

Besides these unsupervised representation learning methods, there have been many supervised deep models which directly learn text representations for the prediction tasks. Recursive Neural Network [14] has been proven to be efficient in terms of constructing sentence representations. Recurrent Neural Network [15] can be viewed as an extremely deep neural network with weight sharing across time. Convolution Neural Network [16] can fairly determine discriminative phrases in a text with a max-pooling layer. However, these deep models are usually task dependent and time-consuming in training due to the complex model structures.

vMF in Text Models

The von Mises-Fisher distribution is known in the literature on directional statistics [17], and suitable for data distributed on the unit hypersphere. Here we first review the vMF distribution.

A d-dimensional unit random vector x (*i.e.*, $x \in \mathbb{R}^K$ and $||x|| = 1$) is said to have K-variate von Mises-Fisher distribution if its probability density function is given by,

$$f(x|\mu, \kappa) = c_K(\kappa) e^{\kappa \mu^{\mathrm{T}} x},$$

where $||\mu|| = 1$, $\kappa \geq 0$ and $K \geq 2$. The normalizing constant $c_K(\kappa)$ is given by,

$$c_K(\kappa) = \frac{\kappa^{K/2-1}}{(2\pi)^{K/2} I_{K/2-1}(\kappa)},$$

where $I_r(\cdot)$ represents the modified Bessel function of the first kind and order r. The density $f(x|\mu, \kappa)$ is parameterized by the mean direction μ, and the concentration parameter κ. The concentration parameter κ characterizes how strongly the unit vectors drawn from the distribution are concentrated on the mean direction μ. The vMF distribution has properties analogous to those of the multi-variate Gaussian distribution for data in $\mathbb{R}^K$, parameterized by cosine similarity rather than Euclidean distance. Evidence suggests that this type of directional measure (*i.e.*, cosine similarity) is often superior to Euclidean distance in high dimensions [18,19].

The vMF distribution has been applied in text representations based on BoW in literature. For example, [10] introduced the mixture of von Mises-Fisher distributions (movMF) that serves as a generative model for directional text data. The movMF model treats each normalized text vector (*i.e.*, normalized *tf* or *tf-idf* vector) as drawn from one of the M vMF distributions centered on one cluster mean, selected by a mixing distribution. The cluster assignment variable for instance x_i is denoted by $z_i \in \{1, 2, \ldots, M\}$. The probabilistic generative process is given by,

$$z_i \sim \text{Categorical}(.|\pi), \quad x_i \sim \text{vMF}(.|\mu_{z_i}, \kappa),$$

where parameters $\Theta = \{\pi, \boldsymbol{\mu}, \boldsymbol{\kappa}\}$ are treated as fixed unknown constants and $\boldsymbol{Z} = \{z_i\}_{i=1}^{M}$ are treated as a latent variables. Later, [20] introduced the Spherical Admixture Model (SAM), a Bayesian admixture model of normalized vectors on $\mathbb{S}^{K-1}$.

All these vMF-based methods treat the text as a single object (*i.e.*, a normalized feature vector), and successfully integrate a directional measure of similarity into a probabilistic setting for text modeling. However, the foundations of these methods are still BoW, which means that they cannot leverage the rich semantic relatedness between the words for text representation. Unlike these methods, we use vMF to capture the semantic relatedness encoded in word vectors revealed by cosine similarity, and build text representations based on a better BoWE foundation.

Spherical Paragraph Model

In this section, we describe our proposed SPM in detail, including the notations, the model definition, the inference and parameter estimation algorithms. Besides, we also provide some discussions on SPM as compared with existing advanced text representation methods.
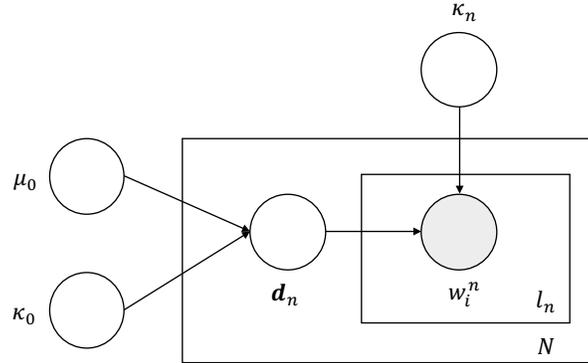
Figure 1. A graphical model representation of Spherical Paragraph Model (SPM). (The boxes are "plates" representing replicates; a shaded node is an observed variable; an unshaded node is a hidden variable.)

Notation

Before presenting our model, we first introduce the notations used in this paper. Let $\boldsymbol{D} = \{d_1, \ldots, d_N\}$ denote a corpus of N texts, where each text $d_n = (w_1^n, w_2^n, \ldots, w_{l_n}^n)$ and $n \in 1, 2, \ldots, N$ is an l_n-length word sequence over the word vocabulary $\boldsymbol{V}$ of size M. Let c_n denote all the words in text d_n. Each text $d \in \boldsymbol{D}$ and each word $w \in \boldsymbol{V}$ is associated with a vector $\boldsymbol{d} \in \mathbb{R}^K$ and $\boldsymbol{w} \in \mathbb{R}^K$, respectively, where K denotes the embedding dimensionality.

Model Definition

SPM is a probabilistic generative model over a text corpus based on BoWE. Specifically, each piece of text is first represented as a bag of ℓ_2-normalized word vectors. Note that by normalization, the cosine similarity between word vectors is equal to the dot product between them, and all the word vectors lie on a unit hypersphere. SPM then assumes the following generative process of the corpus:

For each text $d_n \in \boldsymbol{D}, n = 1, 2, \ldots, N$:
(a) Draw paragraph vector $\boldsymbol{d}_n \sim \text{vMF}(\mu_0, \kappa_0)$
(b) For each word $w_i^n \in d_n, i = 1, 2, \ldots, l_n$:
    Draw word vector $\boldsymbol{w}_i^n \sim \text{vMF}(\boldsymbol{d}_n, \kappa_n)$

where μ_0 is the corpus mean direction, κ_0 controls the concentration of text vectors around μ_0, and κ_n controls the concentration of word vectors around the text vector $\boldsymbol{d}_n$. Figure 1 provides the graphical model of the SPM.

As we can see from the above generative process, in SPM the text vectors in a corpus are determined by the corpus-wide prior distribution over the unit hypersphere, as well as the word vectors contained in the text. By using the vMF distribution, all the relations between these vectors are modeled by the dot product, which is equal to the cosine similarity measure between them (due to the ℓ_2-normalization). As we known, cosine similarity is widely adopted in revealing semantic relatedness in previous neural word embedding methods [6,21].

Based on the above generative process, we can obtain the joint probability of the whole corpus as follows,

$$p(\boldsymbol{D}) = \prod_{n=1}^{N} \int p(d_n|\mu_0, \kappa_0) \prod_{w_i^n \in d_n} p(w_i^n|d_n, \kappa_n) d\boldsymbol{d}_n,$$

where:

$$P(w_i^n; d_n; \kappa_n) = e^{\kappa_n \boldsymbol{d}_n{}^{\mathrm{T}} \boldsymbol{w}_i^n} c_K(\kappa_n).$$

Variational Inference

The key inferential problem that we need to solve in order to use SPM is that of computing the posterior distribution of the hidden text vector given its word vectors and the corpus prior:

$$p(d_n|c_n, \mu_0, \kappa_0, \kappa_n) = \frac{p(d_n, c_n|\mu_0, \kappa_0, \kappa_n)}{p(c_n|\mu_0, \kappa_0, \kappa_n)}.$$

Unfortunately, this distribution is intractable to compute in general. Thus we develop an efficient variational inference algorithm to perform approximate inference in SPM.

The basic idea of convexity-based variational inference is to make use of Jensen's inequality [22] to obtain an adjustable lower bound on the log likelihood. We approximate the posterior by introducing an distinct vMF distribution for each document,

$$q(d_n) \sim \text{vMF}(.|\mu_n', \kappa_n').$$

Here, μ_n', κ_n' are the free variational parameters. To approximate the posterior distribution of the latent variables, the mean-field approach finds the optimal parameters of the fully factorizable q (*i.e.*, $q(d_n)$) by maximizing the Evidence Lower Bound (ELBO),

$$\begin{aligned}
\mathcal{L} &= E_q[\log P(\mathcal{D})] - \mathcal{H}(q) \\
&= E_q[\log P(\boldsymbol{D}, \boldsymbol{V}|\mu_0, \kappa_0, \kappa_n)] - E_q[\log q(\boldsymbol{D})] \\
&= E_q[\log P(\boldsymbol{D}|\mu_0, \kappa_0)] + E_q[\log P(\boldsymbol{V}|\boldsymbol{D}, \kappa_n)] - E_q[\log q(\boldsymbol{D})].
\end{aligned}$$

Note that the expectations in this expression are taken over the variational distribution q. The posterior expectation of text vector $\boldsymbol{d}_n$ is given by,

$$E_q[\boldsymbol{d}_n] = \mu_n' \left(\frac{I_{d/2}(\kappa_n')}{I_{d/2-1}(\kappa_n')} \right),$$

where $\frac{I_{d/2}(\kappa_n')}{I_{d/2-1}(\kappa_n')}$ is a ratio of Bessel functions [23] that differ in their order by just one.

Thus the optimizing values of the variational parameters μ_n' and κ_n' are found by minimizing the KL divergence between the variational distribution q and the true posterior $p(d_n|c_n, \mu_0, \kappa_0, \kappa_n)$. Optimizing the ELBO with respect to μ_n' and κ_n', we have

$$\kappa_n' = ||\kappa_0 \mu_0 + \sum_{i=1}^{l_n} \kappa_n \boldsymbol{w}_i^n||,$$

$$\mu_n' = \frac{\kappa_0 \mu_0 + \sum_{i=1}^{l_n} \kappa_n \boldsymbol{w}_i^n}{||\kappa_0 \mu_0 + \sum_{i=1}^{l_n} \kappa_n \boldsymbol{w}_i^n||} = \frac{\kappa_0 \mu_0 + \sum_{i=1}^{l_n} \kappa_n \boldsymbol{w}_i^n}{\kappa_n'}.$$

Parameter Estimation

We use an empirical Bayes method for parameter estimation in our SPM model. As described above, variational inference provides us with a tractable lower bound on the log likelihood. We can thus find approximate empirical Bayes estimates via an alternating variational EM procedure that maximizes the lower bound with respect to the variational parameters μ_n' and κ_n'. Then, for fixed values of the variational parameters, we maximize the lower bound with respect to the model parameters μ_0, κ_0 and κ_n. The variational EM algorithm is as follows:

– (E-step) For each text, find the optimizing values of the variational parameters μ_n', κ_n', as described in the previous section 1.
– (M-step) Maximize the lower bound with respect to the parameters μ_0, κ_0 and κ_n.

These two steps are repeated until the lower bound on the log likelihood converges.

The M-step update for μ_0, κ_0 are given by,

$$\mu_0 = \frac{\sum_{n=1}^N E_q[\boldsymbol{d}_n]}{|| \sum_{n=1}^N E_q[\boldsymbol{d}_n]||},$$

$$\kappa_0 = \frac{\bar{r}K - \bar{r}^3}{1 - \bar{r}^2} \text{ where } \bar{r} = \frac{|| \sum_{n=1}^N E_q[\boldsymbol{d}_n]||}{N}.$$

The M-step update for κ_n is given by,

$$\kappa_n = \frac{\bar{r}K - \bar{r}^3}{1 - \bar{r}^2} \text{ , where } \bar{r} = \frac{E_q[\boldsymbol{d}_n] \sum_{i=1}^{l_n} \boldsymbol{w}_i^{n\mathrm{T}}}{l_n}.$$

Model Discussion

SPM is a probabilistic generative model based on BoWE for text representation. As it bridges two well-known branches in text representation methods, namely the probabilistic generative models and neural embedding models, here we compare SPM with these two types of methods to show its benefits.

Probabilistic generative models, also called probabilistic topic models (*e.g.*, PLSI and LDA), are advanced text modeling approaches. By assuming a generative process of the texts under a probabilistic framework, these methods usually have sound theoretical foundation and good model interpretability. However, there are two major problems in traditional topic models: (1) As built upon the BoW representation, traditional topic methods do not leverage the rich semantic relatedness of the words, and learn the text representations purely based on the word-by-text co-occurrence information; (2) There is an explicit topic layer in these models to guide the word clustering. The topic number is usually heuristically defined *a prior* which may lead to non-optimal word clustering. As we can see, SPM enjoys the merits of good interpretability as a probabilistic generative model. Meanwhile, SPM can avoid the arbitrary definition of topic numbers by eliminating the topic layer, while allows unlimited hidden topics (*i.e.*, word clusters) learned by any prior neural word embedding models based on very large corpus.

As compared with neural embedding models, here we take the state-of-the-art PV model as an example. The PV model can also be viewed as a probabilistic model based on its prediction definition. However, from the probabilistic view, PV is not a

full Bayesian model and suffers a similar problem as PLSI that it provides no model on text vectors. Therefore, texts from the same corpus are assumed to be independent from each other and no corpus-wide constraint is employed in text modeling. Moreover, it is unclear how to infer the representations for texts outside of the training set with the learned model. Although PV makes itself as an optimization problem so that one can learn representations for new texts anyway, it loses the sound probabilistic foundation in that way. In contrary, SPM solves this problem by defining a complete Bayesian model. In this way, it can not only leverage corpus-wide information to help constrain the text vectors, but also infer the representations of unseen texts based on the learned model, at the expense of the usage of an approximate variational method.

Experiments

In this section, we conduct experiments to verify the effectiveness of SPM based on two text classification tasks.

Baselines

- **Bag-of-Words**. The Bag-of-Words model (BoW) [1] represents each text as a bag of words using *tf* as the weighting scheme. We select top $5,000$ words according to *tf* scores as discriminative features.
- **LSI** and **LDA**. LSI [2] maps both texts and words to lower-dimensional representations using SVD decomposition. In LDA [5], each word within a text is modeled as a finite mixture over an set of topics. We use the vanilla LSI and LDA in the gensim library[1] with topic number set as 50.
- **movMF** and **SAM**. movMF[2] [10] is the mixture of von-Mises Fisher clustering with soft assignments. SAM[3] [20] is a class of topic models that represent data using directional distributions on the unit hypersphere. The topic numbers are both 50.
- **cBow**. We use average pooling to compose a text vector from a set of word vectors [6], where the dimension of text vectors is set as 50.
- **PV**. Paragraph Vector [9] is an unsupervised model to learn distributed representations of words and texts. We implement PV-DBOW and PV-DM model initialized with 50-dimension word embeddings due to the original code has not been released.
- **skip-thought** and **FastSent**. skip-thought[4] [24] encodes a sentence to predict sentences around it using 2400-dimension vector representation. FastSent[5] [25] is a simple additive sentence model designed to exploit the same signal, but at much lower computational expense under 100 dimension.

---

[1] http://radimrehurek.com/gensim/
[2] https://github.com/mrouvier/movMF
[3] https://github.com/austinwaters/py-sam
[4] https://github.com/ryankiros/skip-thoughts
[5] https://github.com/fh295/SentenceRepresentation

Model	topical classification				sentiment analysis	
	different	similar	same	Reuters	Subj	MR
BoW	91.4	81.8	75.6	**95.4**	89.5	74.3
LSI	85.2	80.1	68.2	93.1	85.4	64.2
LDA	73.3	67.5	56.7	89.6	72.7	58.2
movMF	71.4	64.5	59.4	87.1	67.6	53.4
SAM	88.6	81.2	70.5	88.2	74.2	61.8
cBow	91.6	81.6	75.9	91.8	90.8	74.4
PV-DBOW	91.4	80.2	**76.2**	89.6	90.1	73.9
PV-DM	91.5	80.8	76.1	90.4	90.4	74.4
FastSent	89.6	80.1	61.5	89.4	88.7	70.8
uni-skip	86.4	77.8	59.2	77.4	92.1	**75.5**
SPM	**91.8**	**82.0**	70.0	93.2	**92.5**	75.0

Table 1. Classification accuracies (%) of different models on topical classification and sentiment analysis. Best scores are shown in bold.

Setup

We perform experiments on two text classification tasks: topical classification and sentiment analysis. We utilize 50-dimension word embeddings trained on Wikipedia with word2vec [6]. The corpus in total has $3,035,070$ articles and about 1 billion tokens. The vocabulary size is about $400,000$. The vectors are post-processed to have unit ℓ_2-norm. In our model, text vectors are randomly initialized with values uniformly distributed in the range of [-0.5, +0.5] with 50-dimension and then ℓ_2-normalized, κ_0 is intialized as 1500 and κ_n are randomly initialized with values uniformly distributed in the range of [1000, 1500]. Through our experiments, we use support vector machines (SVM)[6] as the classifier. Preprocessing steps were applied to all datasets: words were lowercased, non-English characters and stop words occurrence in the training set are removed. If explicit split of train/test is not provided, we use 10-fold cross-validation instead.
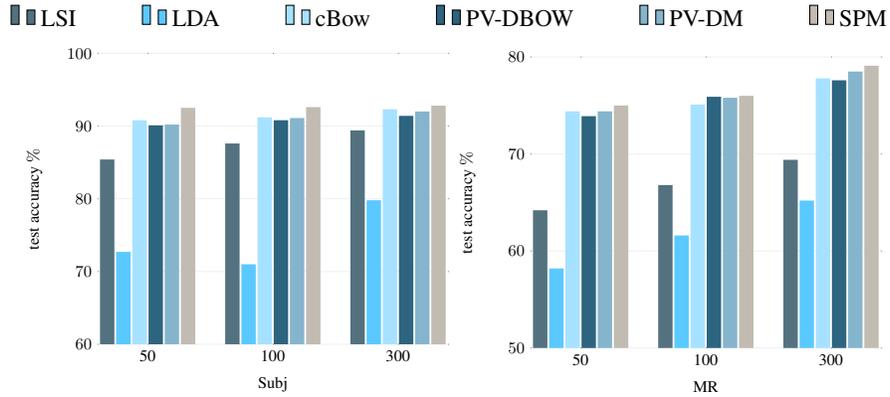
Topical Classification

We used two standard topical classification corpora: the 20Newsgroups[7] and the Reuters corpus [8]. The 20Newsgroups contains about $20,000$ newsgroup documents harvested from 20 different Usenet newsgroups, with about $1,000$ documents from each newsgroup. Following [26], three subsets of 20News are used for evaluation: (1) **news-20-different** consists of three newsgroups that cover different topics (*rec.sport.baseball*, *sci.space* and *alt.atheism*); (2) **news-20-similar** consists of three newsgroups on the more similar topics (*rec.sport.baseball*, *talk.politics.guns* and *talk.politics.misc*); (3) **news-20-same** consists of three newsgroups on the highly related topics (*comp.os.ms-windows.misc*, *comp.windows.x* and *comp.graphics*). The Reuters contains $10,788$ documents, where each document is assigned to one or more categories. Documents appearing in two or more categories were removed and we selected the largest 10 categories, leaving $8,025$ documents in total.

---

[6] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[7] http://qwone.com/~jason/20Newsgroups/

[8] http://www.nltk.org/book/ch02.html

Figure 2. Sentiment Classification accuracies on two datasets. Results are grouped by dimension.

Results The left of Table 1 shows the evaluation results on topical classification. We have the following observations: (1) The BoW representation, although simple, can achieve surprising accuracy using much larger dimensionality (*i.e.*, $5,000$ dimension). Meanwhile, our SPM, using only 50-dimension text vector, can achieve slightly better or comparable performance as BoW. (2) As compared with the text representation methods built upon BoW (*i.e.*, LSI, LDA, movMF and SAM), SPM can outperform these methods almost. The results indicate that learning text representations over BoWE can in general achieve better performances than that over BoW by involving rich semantics between words. (3) Comparing with the three BoWE based representation methods, namely cBow, PV-DBOW and PV-DM, we find our SPM can outperform them on three out of four datasets. Recall that in cBow, PV-DBOW and PV-DM, texts in a corpus are actually assumed to be independent from each other. These results indicate that by modeling texts under a sound probabilistic generative framework, SPM can well leverage the corpus-wide information to help improve the text representation. (4) Compared with FastSent and uni-skip, SPM can outperform both of them over the four datasets. It seems that FastSent and uni-skip, which were proposed for short texts (*i.e.*, sentences) modeling originally, cannot work well on long texts.

Sentiment Analysis

We run the sentiment classification experiments on two publicly available datasets: (1) **Subj**, Subjectivity dataset [27][9] which contains $5,000$ subjective instances and $5,000$ objective instances. The task is to classify a sentence as being subjective or objective; (2) **MR**, Movie reviews [28] with one sentence per review. There are $5,331$ positive and $5,331$ negative sentences. Classification involves detecting positive/negative reviews.

Results The right of Table 1 shows the evaluation results on sentiment analysis. We have the following observations: (1) SPM can outperform all the baseline methods on the Subj dataset. This indicates that SPM can capture better semantic representations of texts using a probabilistic generative model over BoWE. (2) SPM can also outperform

---

[9] http://www.cs.cornell.edu/people/pabo/movie-review-data/

all the baseline methods except uni-skip on the MR dataset. Note that skip-thought uses 2400-dimension sentence representation while SPM only uses 50-dimension vector. However, SPM can still achieve similar performance as uni-skip on the MR dataset even with much less model parameters.

We conduct evaluations over different dimensions (*i.e.*, 50, 100, 300) to see the impact of the dimensionality on different models. For cBow, PV and SPM, we utilize 50, 100 and 300 dimensional word embeddings trained on Wikipedia using word2vec. For LSI and LDA, we set the topic numbers as 50, 100 and 300 for comparison. Figure 2 shows the results on the two different datasets. As we can see, with the increase of the dimensionality, all the models can improve their performance while SPM can consistently outperform all the other baselines. Moreover, we can find that the SPM model under dimensionality 100 can already beat the uni-skip under dimensionality 2400 (76.0% vs 75.5%)

Conclusion

In this paper, we propose the SPM, a novel generative model based on BoWE for text modeling. The SPM is a full Bayesian framework which models the generation of both the text vectors and word vectors, where the vMF distribution is employed to capture the directional relations between these vectors. SPM has good probabilistic interpretability and can fully leverage the rich semantics of words, the word co-occurrence information as well as the corpus-wide information to help the representation learning. The experimental results demonstrate that SPM can achieve new state-of-the-art performances on several topical classification and sentiment analysis tasks.

For the future work, we would like to explore the possibility to jointly learn word and text vectors in SPM. One idea is to leverage the word vectors learned from other large corpus as the initialization, and fine-tune them on the training data. Moreover, word order information is often critical in capturing the meaning of texts. We would also try to accommodate n-grams in the generative process to enhance the model ability. We may also test SPM on other text processing tasks to verify its generalization ability.

Acknowlegements

References

1. Harris, Z.S.: Distributional structure. Word **10**(2-3) (1954) 146–162
2. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. Journal of the American society for information science **41**(6) (1990) 391

3. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature **401**(6755) (1999) 788–791
4. Hofmann, T.: Probabilistic latent semantic indexing. In: SIGIR, ACM (1999) 50–57
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of machine Learning research **3**(Jan) (2003) 993–1022
6. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
7. Vulic, I., Moens, M.F.: Cross-lingual semantic similarity of words as the similarity of their semantic word responses. In: NAACL-HLT, ACL (2013) 106–116
8. Clinchant, S., Perronnin, F.: Aggregating continuous word embeddings for information retrieval. In: Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality. (2013) 100–109
9. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML. Volume 14. (2014) 1188–1196
10. Banerjee, A., Dhillon, I.S., Ghosh, J., Sra, S.: Clustering on the unit hypersphere using von mises-fisher distributions. Journal of Machine Learning Research **6**(Sep) (2005) 1345–1382
11. Salton, G., McGill, M.J.: Introduction to modern information retrieval. (1986)
12. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP. Volume 14. (2014) 1532–1543
13. Jaakkola, T.S., Haussler, D., et al.: Exploiting generative models in discriminative classifiers. NIPS (1999) 487–493
14. Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: EMNLP. Volume 1631., Citeseer (2013) 1642
15. Sutskever, I., Martens, J., Hinton, G.E.: Generating text with recurrent neural networks. In: ICML-11. (2011) 1017–1024
16. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNLP. (2014) 1746–1751
17. Mardia, K.V., Jupp, P.E.: Directional statistics. Volume 494. John Wiley & Sons (2009)
18. Manning, C.D., Schütze, H., et al.: Foundations of statistical natural language processing. Volume 999. MIT Press (1999)
19. Zhong, S., Ghosh, J.: Generative model-based document clustering: a comparative study. Knowledge and Information Systems **8**(3) (2005) 374–384
20. Reisinger, J., Waters, A., Silverthorn, B., Mooney, R.J.: Spherical topic models. In: ICML-10. (2010) 903–910
21. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS. (2013) 3111–3119
22. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. Machine learning **37**(2) (1999) 183–233
23. Watson, G.N.: A treatise on the theory of Bessel functions. Cambridge university press (1995)
24. Kiros, R., Zhu, Y., Salakhutdinov, R.R., Zemel, R., Urtasun, R., Torralba, A., Fidler, S.: Skip-thought vectors. In: NIPS. (2015) 3294–3302
25. Hill, F., Cho, K., Korhonen, A.: Learning distributed representations of sentences from unlabelled data. In: NAACL-HLT. (2016)
26. Banerjee, A., Basu, S.: Topic models over text streams: A study of batch and online unsupervised learning. In: SDM. Volume 7., SIAM (2007) 437–442
27. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: ACL, ACL (2004) 271
28. Pang, B., Lee, L.: Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In: ACL, ACL (2005) 115–124