Aggregating Neural Word Embeddings for Document Representation

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 Recent advances in natural language processing (NLP) have shown that semantically meaningful representations of words can be efficiently acquired by distributed models. In such a case, a text document can be viewed as a bag-of-word-embeddings (BoWE), and the remaining question is how to obtain a fixed-length vector representation of the document for efficient document process. Beyond those heuristic aggregation methods, recent work has shown that one can leverage the Fisher kernel (FK) framework to generate document representations based on BoWE in a principled way. In this work, words are embedded into a Euclidean space by latent semantic indexing (LSI), and a Gaussian Mixture Model (GMM) is employed as the generative model for nonlinear FK-based aggregation. In this work, we propose an alternate FK-based aggregation method for document representation based on neural word embeddings. As we know, neural embedding models have been proven significantly better performance in word representations than LSI, where semantic relations between neural word embeddings are typically measured by cosine similarity rather than Euclidean distance. Therefore, we introduce a mixture of Von Mises-Fisher distributions (moVMF) as the generative model of neural word embeddings, and derive a new FK-based aggregation method for document representation based on BoWE. We report document classification, clustering and retrieval experiments and demonstrate that our model can produce state-of-the-art performance as compared with existing baseline methods.

1 Introduction

Representing text documents as fixed-length vectors is central to many language processing tasks. Perhaps the most popular fixed-length vector representation for documents is the bag-of-words (BoW) representation [1], where each word is viewed as a distinct feature dimension based on strong independent assumption. Most traditional methods either directly use the BoW representation (e.g., tf-idf vector), or are built upon BoW (e.g., matrix factorization [2,3] and probabilistic topical models [4,5]). Apparently, by using BoW as the foundation, rich semantic relatedness between words is lost. The document representation thus is obtained purely based on the word-by-document co-occurrence information.

Recent developments in distributed word representations [6,7] have succeeded in revealing rich linguistic regularities between words. Specifically, by mapping each word

into a continuous vector space, both syntactic and semantic relatedness between words can be captured using simple algebra over word vectors. Therefore, a natural idea is that one can build document 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. The follow-up question is how to obtain a fixed-length vector representation of document based on BoWE for efficient document processing.

There have been several heuristic ways to obtain the document vector based on word embeddings, e.g., by using the average or weighted sum of all the word vectors contained in a document [8]. Another well-known approach is the Paragraph Vector (PV) [9] method, which jointly learns the word and document vectors through some prediction task. A common problem of all these methods is that they assume that the document vector lies in the same semantic space as words vectors. However, this may not be a necessary condition in practice since documents usually convey much richer semantics than individual words.

Recent work [10] has shown that one can use the Fisher kernel (FK) framework [11] as a flexible and principled way to generate document representations based on BoWE. It consists in non-linearly mapping the word embeddings into a higher-dimensional space and in aggregating them into a document representation. Specifically, in the FK-based aggregation, words are embedded into a Euclidean space by latent semantic indexing (LSI), and a Gaussian Mixture Model (GMM) is employed as the generative model of the word embeddings. The gradients of the GMM parameters are then used to generate the document representation. This FK-based aggregation method is highly efficient (i.e., simple adding operation to generate a new document representation), and has shown its superiority in several document clustering and retrieval tasks.

However, recent advances have shown that neural word embedding models (e.g., word2vec [6]) can produce significantly better performance in word representations than LSI. Such neural word embeddings can be efficiently acquired from large text corpus. Therefore, a natural question is whether we could leverage neural word embeddings for better document representation under the FK framework. Unfortunately, directly using the existing FK-based aggregation method [10] over neural word embeddings may not be appropriate. The major reason is that the generative model (i.e., GMM) in [10] is employed to capture the Euclidean distances between word embeddings from LSI, while semantic relations between neural word embeddings (e.g., Glove and word2vec) are typically measured by cosine similarity. Therefore, we propose an alternate FK-based aggregation method for document representation based on neural word embeddings. As we known, the von Mises-Fisher (vMF) distribution is well-suited to model directional data distributed on the unit hypersphere and capture the directional relations (i.e., cosine similarity) between vectors. Therefore, we introduce a Mixture of von Mises-Fisher distributions (moVMF) [12] as the generative model of neural word embeddings, and derive a new aggregation algorithm based moVMF model under the FK framework. We evaluated the effectiveness of our model by comparing with existing document representation methods. The empirical results demonstrate that our model can achieve new state-of-the-art performances on several document classification, clustering and retrieval tasks.

2 Related Work

We provide a short review of the works on those topics which are most related to our work: Bag-of-Words, Bag-of-Word-Embeddings, vMF and Fisher Vector.

- **Bag-of-Words** The most common fixed-length representation is Bag-of-Words (BoW) [1]. For example, in the popular TF-IDF scheme, each document is represented by *tfidf* values of a set of selected feature-words. Besides, 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 PLSA [4] and LDA [5]. LDA, the generative counterpart of PLSA, has played a major role in the development of probabilistic models for textual data. As a result, it has been extended or refined in a countless studies [13,14]. Besides, several studies reported that LDA does not generally outperform LSI in IR or sentiment analysis tasks [15,16]. To further tackle the prediction task, Supervised LDA [17] is developed by jointly modeling the documents and the labels.
- **Bag-of-Word-Embeddings** Recent advances in the natural language processing (NLP) community have shown that semantics of words or more formally the distances between words can be effectively revealed by distributed word representations. Specifically, neural embedding models, *e.g.*, Word2Vec [6] and Glove [7], learn word vectors efficiently from very large text corpus. Word embeddings are useful because they encode both syntactic and semantic information of words into continuous vectors and similar words are close in vector space. With rich semantics encoded in word vectors, there have been many methods [8,9,18,19,20] built upon Bag-of-Word-Embedding (BoWE) for document representations.
- **vMF in topic models** The vMF distribution has been used to model directional data by placing points on a unit sphere and is known in the literature on directional statistics [21]. [12] proposed an admixture model (moVMF) that uses vMF to model the document corpus based on normalized word frequency vectors. [22] used vMF as the observational distribution of each word and used a Hierarchical Dirichlet Process (HDP) [23], a Bayesian nonparametric variant of Latent Dirichlet Allocation (LDA), to automatically infer the number of topics.
- **Fisher Kernel** Fisher kernel is a generic framework introduced in [11] for classification purposes to combine the strengths of the generative and discriminative worlds. The idea is to characterize a signal with a gradient vector derived from a probability density function (pdf) which models the generation process of the signal. This representation can then be used as input to a discriminative classifier. This framework has been successfully applied to computer vision [24,25] and text analysis[10]. The gradient representation of the Fisher kernel has a major advantage over the histogram of occurrences of the BoW: for the same vocabulary size, it is much larger. Hence, there is no need to use costly kernels to (implicitly) project these very high-dimensional gradient vectors into a still higher dimensional space.

3 Model

In this section, we describe our proposed FK framework in detail, including the generation process of words with continuous mixture models and the FK-based aggregation. The proposed procedure is as follows:
Learning phase: Given an unlabeled training set of documents:

– Learn the neural word embedding in a low-dimensional space, e.g., by word2vec. After this operation, each word w is then represented by a vector E_w of size d.
– Fit a probabilistic model, i.e., a mixture of Von Mises-Fisher model (moVMF), on these neural word embeddings. The detailed description of moVMF is shown in the following Probabilistic modeling Section.

Document representation: Given a document whose BoW representation is $\{w_1, \ldots, w_T\}$:

– Transform the BoW representation into the BoWE representation:

$$\{w_1, \ldots, w_T\} \rightarrow \{E_{w_1}, \ldots, E_{w_T}\}$$

– Aggregate the neural word embeddings E_{w_t} using the Fisher Kernel framework. We detail the framework in the following Fisher kernel aggregation Section.

3.1 Probabilistic modeling

We use the mixture of Von Mises-Fisher distributions (moVMF) as the generative model of neural word embeddings. Here we describe the vMF distribution and moVMF model in detail.

The von Mises-Fisher distribution is known in the literature on directional statistics, and suitable for data distributed on the unit hypersphere. A d-dimensional unit random vector x (*i.e.*, $x \in \mathbb{R}^d$ and $||x|| = 1$) is said to have d-variate von Mises-Fisher distribution if its probability density function is given by,

$$f(x|\mu, \kappa) = c_d(\kappa) e^{\kappa \mu^{\mathrm{T}} x}, \tag{1}$$

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

$$c_d(\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)}, \tag{2}$$

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. Larger values of κ imply stronger concentration about the mean direction.

Later, [12] introduce the mixture of von Mises-Fisher distributions (moVMF) that serves as a generative admixture model for directional data. Let $f_i(x|\theta_i)$ denote a vMF

distribution with parameter $\theta_i = (\mu_i, \kappa_i)$ for $1 \leq i \leq N$. Then a mixture of these N vMF distributions has a density given by

$$f(x|\Theta) = \sum_{i=1}^{N} \alpha_i f_i(x|\theta_i), \tag{3}$$

where parameters $\Theta = \{\alpha_1, \ldots, \alpha_N, \theta_1, \ldots, \theta_N\}$ and the α_i are non-negative and sum to one. To sample a point from this mixture density we choose the i-th vMF randomly with probability α_i, and then sample a point on $\mathbb{S}^{d-1}$ ($\mathbb{S}^{d-1}$ denotes the $(d-1)$-dimensional sphere embedded in $\mathbb{R}^d$) following $f_i(x|\theta_i)$. To train the model, we can use the familiar EM algorithm, to efficiently iterate between estimating the most likely conditional distribution of $\{\alpha_1, \ldots, \alpha_N\}$ in the E-step and optimizing $\{\theta_1, \ldots, \theta_N\}$ to maximize the likelihood in the M-step. The moVMF generalizes clustering methods parameterized by cosine distance and it successfully integrates a directional measure of similarity into a probabilistic setting.

3.2 Fisher kernel aggregation

In this work, we describe a given document, $X = \{x_t, t = 1 \ldots T\}$, as a set of d-dimensional neural word embeddings whose generation process can be modeled by the probability density function (pdf) of moVMF. Evidence suggests that this type of directional measure (i.e., cosine similarity) is often superior to Euclidean distance in high dimensions [26]. In this moVMF, each vMF distribution p_i can be viewed as a visual word and N is the vocabulary size. We denote $\lambda = \{w_i, \mu_i, \kappa_i, i = 1 \ldots N\}$, where $\{w_i, \mu_i, \kappa_i\}$ are respectively the mixture weight, mean vector and concentration of i-th vMF.

In practice, the moVMF is estimated offline with a set of neural word embeddings learned *a prior* from a large training set of documents. The parameters Θ are estimated through the optimization of a Maximum Likelihood (ML) criterion using the Expectation-Maximization (EM) algorithm.

Since the partial derivatives with respect to mixture weights α_Θ and concentration parameters κ_Θ carry little additional information, we only focus on the partial derivatives with respect to the mean parameters μ_Θ. Given μ_Θ, X can be described by the gradient vector:

$$G_\Theta^X = \nabla_{\mu_\Theta}^T \log f(X|\Theta). \tag{4}$$

Intuitively, it describes in which direction the parameters Θ of the model should be modified so that the model μ_Θ better fits the data. Assuming that the word embeddings x_t in X are iid, we have:

$$G_\Theta^X = \sum_{t=1}^{T} \nabla_{\mu_\Theta} \log f(x_t|\Theta). \tag{5}$$

In the following, $\gamma_t(i)$ denotes the occupancy probabiltity, i.e. the probability for observation x_t to be generated by the i-th vMF. Bayes formula gives:

$$\gamma_t(i) = p(i|x_t, \Theta) = \frac{\alpha_i f_i(x|\theta_i)}{\sum_{j=1}^{N} \alpha_j f_j(x|\theta_j)}. \tag{6}$$

Simple mathematical derivation with respect to μ_i has:

$$G_{\mu_i}^X = \sum_{t=1}^{T} \gamma_t(i) \kappa_i x_t. \tag{7}$$

To normalize the dynamic range of different dimensions of gradient vectors, it is important to normalize the vectors. As in [11], the Fisher information matrix (FIM) F_Θ of μ_Θ is suggested for this purpose:

$$F_\Theta = E_{x \sim \mu_\Theta}[\nabla_\Theta \log f(x|\Theta) \nabla_\Theta \log f(x|\Theta)'']. \tag{8}$$

As F_Θ is symmetric and positive definite, it has a Cholesky decomposition. Then, [11] proposed to measure the similarity between two samples X and Y:

$$K(X, Y) = G_\Theta^{X'} F_\Theta^{-1} G_\Theta^Y. \tag{9}$$

Then $K(X, Y)$ can be rewritten as a dot-product between normalized vectors $\mathcal{G}_\Theta$ with:

$$\mathcal{G}_\Theta^X = F_\Theta^{-1/2} G_\Theta^X, \tag{10}$$

where $\mathcal{G}_\Theta^X$ is referred to as the *Fisher Vector* (FV) of X [27].

Let f_{μ_i} denote the diagonal approximation of FM which corresponds respectively to μ_i. According to Equation 8, we can get

$$f_{\mu_i} = \int_X f(X|\Theta)[\sum_{t=1}^{T} \gamma_t(i) \kappa_i x_t]^2 dX. \tag{11}$$

Using the diagonal approximation of the FIM, we finally obtain the following formula for the gradient with respect to μ_i:

$$\mathcal{G}_i^X = f_{\mu_i}^{-1/2} G_{\mu_i}^X = \sum_{t=1}^{T} \frac{\gamma_t(i) x_t d}{w_i \kappa_i ||\mu_i||}. \tag{12}$$

The FV $\mathcal{G}_\Theta^X$ is the concatenation of the $\mathcal{G}_i^X, \forall i$, and is therefore N×d dimensional, where d is the dimensionality of the continuous word embeddings and N is the number of vMFs.

4 Experiments

In this section, we conduct experiments to verify the effectiveness of our model over docuemnt classification, clustering and retrieval tasks.

4.1 Baselines

- **Bag-of-word**. The Bag-of-Words model (BoW) [1] represents each document as a bag of words using *tf-idf* [28] as the weighting scheme. We select top $5,000$ words according to *tf-idf* scores and use the vanilla TFIDF in the gensim library[1].
- **LSI** LSI [2] maps both documents and words to lower-dimensional representations in a so-called latent semantic space using singular value decomposition (SVD) decomposition. We use the vanilla LSI in the gensim library with topic number set as 50.
- **LDA**. In LDA [5], each word within a document is modeled as a finite mixture over an set of topics. We use the vanilla LDA in the gensim library with topic number set as 50.
- **cBow**. Continuous Bag-of-Words model [6]. We use average pooling to compose a document vector from a set of word vectors.
- **PV**. Paragraph Vector [9] is an unsupervised model to learn distributed representations of words and documents. We implement PV-DBOW and PV-DM model by ourselves since no original code is available.
- **FV-GMM**. Fisher Kernel based on Gaussian mixture model (GMM) [10] is used for document representation from word embeddings. It treats documents as bags-of-embedded-words (BoEW) and to learn probabilistic mixture models once words were embedded in a Euclidean space.

We refer to our FK-based aggregation method as **FV-moVMF**.

4.2 Setup

We used two datasets for classificaiton, one for clustering and one for information retrieval. Preprocessing steps were applied to all the datasets: words were lowercased, non-English characters and stop words were removed. All the neural word embeddings used in the above methods were trained on the corresponding document collections in each task under 50-dimension by word2vec[2]. For FK-based aggregation methods, the number of mixture components were set as 15 since we observed ignorable performance differences with larger value. In previous work, FV-GMM [10] obtained the word embeddings by LSI. For comparison, we also tried FV-GMM based on neural word embeddings.

We refer to these two types of aggregation methods as FV-GMM$_{LSI}$ and FV-GMM$_{Neu}$, respectively. Similarly, we also have two versions of FV-moVMF, namely FV-moVMF$_{LSI}$ and FV-moVMF$_{Neu}$.

4.3 Classification

We run the classification experiments on two publicly available datasets:

[1] `http://radimrehurek.com/gensim/`
[2] `https://code.google.com/p/word2vec/`

Table 1. Classification accuracies (%) of different models. Best scores are bold. Two-tailed t-tests demonstrate the improvements of our model to all the baseline models are statistically significant (‡ indicates p-value < 0.05).

Model	Subj	MR
BoW	89.5	74.3
LSI	85.4	64.2
LDA	72.7	58.2
cBow	90.9	74.8
PV-DBOW	90.1	73.9
PV-DM	90.4	74.4
FV-GMM$_{LSI}$	87.8	68.5
FV-GMM$_{Neu}$	90.3	72.6
FV-moVMF$_{LSI}$	88.6	71.5
FV-moVMF$_{Neu}$	**91.8**‡	**75.7**‡

– **Subj**, Subjectivity dataset [29][3] which contains $5,000$ subjective instances (snippets) and $5,000$ objective instances (snippets). The task is to classify a sentence as being subjective or objective;
– **MR**, Movie reviews [30] with one sentence per review. There are $5,331$ positive sentences and $5,331$ negative sentences. Classification involves detecting positive/negative reviews.

We use 10-fold cross-validation and Logistic Regression as the the classifier.

Table 1 shows the evaluation results on the two datasets. The results show that learning text representations over BoWE (e.g., cBow, PV-DBOW, PV-DM) can in general achieve better performances than that over BoW (e.g., BoW, LSI and LDA) by involving richer semantics between words. For the FV models, the consistent improvements of neural embedding based methods over LSI based methods (i.e., FV-moVMF$_{Neu}$ and FV-GMM$_{Neu}$ vs FV-moVMF$_{LSI}$ and FV-GMM$_{LSI}$) verify the effectiveness of neural embeddings in capturing word semantics. Furthermore, each version of FV-moVMFs works better than FV-GMMs (e.g., FV-moVMF$_{Neu}$ vs FV-GMM$_{Neu}$), indicating that moVMF is a better statistical model for neural word embeddings than GMMs. Finally, FV-moVMF$_{Neu}$ can outperform all the baselines on the two datasets, demonstrating the effectiveness of our approach.

4.4 Clustering

We used one well-known and publicly available dataset: the 20 Newsgroups[4], for clustering. The 20Newsgroups contains about $20,000$ newsgroup documents harvested from

[3] http://www.cs.cornell.edu/people/pabo/movie-review-data/
[4] http://qwone.com/~jason/20Newsgroups/

Table 2. Clustering experiments of different models (in %). Best scores are bold. Two-tailed t-tests demonstrate the improvements of our model to all the baseline models are statistically significant (‡ indicates p-value < 0.05).

	20News	
Model	ARI	NMI
BoW	9.2	29.7
LSI	33.2	43.1
LDA	30.8	47.2
cBow	38.6	53.8
PV-DBOW	42.6	56.6
PV-DM	42.9	56.8
FV-GMM$_{LSI}$	38.5	46.7
FV-GMM$_{Neu}$	42.3	54.6
FV-moVMF$_{LSI}$	37.9	51.9
FV-moVMF$_{Neu}$	**44.1**‡	**57.8**‡

20 different Usenet newsgroups, with about $1,000$ documents from each newsgroup. We compared k-means over all the methods and use two standard evaluation metrics[5] to assess the quality of the clusters, namely the Adjusted Rand Index (ARI)[31] and Normalized Mutual Information (NMI) [32]. These measures compare the clusters with respect to the partition induced by the category information. For all the clustering methods, the number of clusters is set to the true number of classes of the collections.

Overall, we can observe similar performance trending of different methods as that on the classification tasks. Moreover, the PV methods show better performances than FV-GMM$_{Neu}$. It indicates that dot product employed by PV works better than Euclidean distance used in FV-GMM$_{Neu}$. Finally, our FV-moVMF$_{Neu}$ outperforms all the other baseline models, showing the power of FK framework for document representation with the appropriate generative distribution.

4.5 Document Retrieval

We use one TREC collection: Robust04[6], for the document retrieval task. The topics of Robust04 are collected from TREC Robust Track 2004. It has approximately $500,000$ documents and the vocabulary size is about $600,000$. The retrieval experiments described in this section are implemented using the Galago Search Engine[7]. We use the standard cosine similarity to produce the relevance scores between documents and the query based on different models. For evaluation, the top-ranked $1,000$ documents are

[5] http://scikit-learn.org/stable/

[6] `http://trec.nist.gov/`

[7] `http://www.lemurproject.org/galago.php`

Table 3. Retrieval experiments of different models (in %). Best scores are bold.

	Robust04	
Model	MAP	P@20
BM25	24.1	33.7
LSI	3.4	3.9
LDA	4.7	5.6
cBow	7.2	11.1
FV-GMM$_{Neu}$	9.8	12.4
FV-moVMF$_{Neu}$	11.2	13.9
BM25+LSI	25.3	36.6
BM25+LDA	25.4	36.3
BM25+cBow	25.3	36.5
BM25+FV-GMM$_{Neu}$	25.4	36.3
BM25+FV-moVMF$_{Neu}$	**25.6**	**36.7**

compared using the mean average precision (MAP) and precision at rank 20 (P@20). We also compare with the traditional retrieval model, namely BM25 [33], and linearly combine the normalized scores of BM25 and the other models :

$$score(d, Q) = \lambda score_{BM25}(d, Q) + (1 - \lambda) score_{model}(d, Q), \tag{13}$$

where (d, Q) is the document-query pair and λ is the interpolation parameter. In our experiments, we select λ as 0.8 based on the development set.

From Table 3 we can see that, simple cosine similarity between documents and query based on different representation models cannot work well in the retrieval task since many exact matching singles are lost in this way. When combined with BM25 method, improved performance can be obtained as semantic relatedness between document and query is captured. Moreover, our proposed FV-moVMF$_{Neu}$ can bring the largest improvement among all the combinations, indicating that our model offers a better similarity with latent representations.

5 Conclusion

In this paper we introduced an alternate FK framework for document representations based on BoWE. Our new FK-based aggregation method builds upon neural word embeddings by employing a moVMF distribution as the generative model. The experimental results demonstrate that our model can achieve new state-of-the-art performances on several document processing tasks.

Nevertheless, there is still room to improve our model in the future. For example, we could like to learn the parameters of moVMF together with the FV framework,

instead of estimating offline. Moreover, it is interesting to validate the effectiveness of using other word embedding techniques like Glove [7] and other statistical models for Bag-of-Word-Embeddings.

6 Acknowlegements

This work was funded by the 973 Program of China under Grant No. 2014CB340401, the National Natural Science Foundation of China (NSFC) under Grants No. 61232010, 61433014, 61425016, 61472401, 61203298 and 61722211, the Youth Innovation Promotion Association CAS under Grants No. 20144310 and 2016102, and the National Key R&D Program of China under Grants No. 2016QY02D0405.

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: Proceedings of the 22nd annual international ACM 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. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Volume 14. (2014) 1532–1543
8. Vulic, I., Moens, M.F.: Cross-lingual semantic similarity of words as the similarity of their semantic word responses. In: NAACL-HLT 2013, ACL (2013) 106–116
9. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML. Volume 14. (2014) 1188–1196
10. 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
11. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: NIPS. (1999) 487–493
12. 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
13. Eisenstein, J., Ahmed, A., Xing, E.P.: Sparse additive generative models of text. (2011)
14. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the third ACM international conference on Web search and data mining, ACM (2010) 261–270
15. Wang, Q., Xu, J., Li, H., Craswell, N.: Regularized latent semantic indexing. In: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, ACM (2011) 685–694

16. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, Association for Computational Linguistics (2011) 142–150
17. David M.Blei, J.D.: Supervised topic models. In: Proceedings of Advances in Neural Information Processing Systems. (2007)
18. 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
19. Sutskever, I., Martens, J., Hinton, G.E.: Generating text with recurrent neural networks. In: ICML-11. (2011) 1017–1024
20. Zhao, H., Lu, Z., Poupart, P.: Self-adaptive hierarchical sentence model. In: IJCAI. (2015) 4069–4076
21. Fisher, R.: Dispersion on a sphere. In: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences. Volume 217., The Royal Society (1953) 295–305
22. Batmanghelich, K., Saeedi, A., Narasimhan, K., Gershman, S.: Nonparametric spherical topic modeling with word embeddings. arXiv preprint arXiv:1604.00126 (2016)
23. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Sharing clusters among related groups: Hierarchical dirichlet processes. In: NIPS. (2005) 1385–1392
24. Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: CVPR, IEEE (2007) 1–8
25. Bressan, M., Cifarelli, C., Perronnin, F.: An analysis of the relationship between painters based on their work. In: ICIP, IEEE (2008) 113–116
26. 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
27. Perronnin, F., Liu, Y., Sánchez, J., Poirier, H.: Large-scale image retrieval with compressed fisher vectors. In: CVPR, IEEE (2010) 3384–3391
28. Salton, G., McGill, M. In: Introduction to Modern Information Retrieval. McGraw-Hill (1983)
29. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on Association for Computational Linguistics, Association for Computational Linguistics (2004) 271
30. Pang, B., Lee, L.: Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In: Proceedings of the 43rd annual meeting on association for computational linguistics, Association for Computational Linguistics (2005) 115–124
31. Hubert, L., Arabie, P.: Comparing partitions. Journal of classification **2**(1) (1985) 193–218
32. Estévez, P.A., Tesmer, M., Perez, C.A., Zurada, J.M.: Normalized mutual information feature selection. IEEE Transactions on Neural Networks **20**(2) (2009) 189–201
33. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: SIGIR, Springer-Verlag New York, Inc. (1994) 232–241