Algorithms **2015**, *8*, 562-572; doi:10.3390/a8030562

OPEN ACCESS

algorithms

ISSN 1999-4893

www.mdpi.com/journal/algorithms

Article

Modeling Documents with Event Model

Longhui Wang [1,2] Guoguang Zhao [1,*] and Donghong Sun [1]

[1] Tsinghua University, Beijing 100000, China; E-Mails: henryalink@126.com (L.W.); sundh@cernet.edu.cn (D.S.)

[2] School of Mechanical Engineering, Shandong University, Jinan 250061, China

* Author to whom correspondence should be addressed; E-Mail: zhaoinchina@gmail.com or zhaogg@bioinfo.ac.cn; Tel.: +86-138-1116-7867.

Academic Editor: Jun-Bao Li

Received: 3 May 2015 / Accepted: 23 July 2015 / Published: 4 August 2015

Abstract: Currently deep learning has made great breakthroughs in visual and speech processing, mainly because it draws lessons from the hierarchical mode that brain deals with images and speech. In the field of NLP, a topic model is one of the important ways for modeling documents. Topic models are built on a generative model that clearly does not match the way humans write. In this paper, we propose Event Model, which is unsupervised and based on the language processing mechanism of neurolinguistics, to model documents. In Event Model, documents are descriptions of concrete or abstract events seen, heard, or sensed by people and words are objects in the events. Event Model has two stages: word learning and dimensionality reduction. Word learning is to learn semantics of words based on deep learning. Dimensionality reduction is the process that representing a document as a low dimensional vector by a linear mode that is completely different from topic models. Event Model achieves state-of-the-art results on document retrieval tasks.

Keywords: deep learning; neurolinguistics; topic model; Event Model; document retrieval

1. Introduction

At present, deep learning has become a popular trend in the field of machine learning. Deep learning references the mode that the brain deals with images and speech and has achieved great success in the field of vision and speech processing [1–3]. For natural language processing, people also proposed some topic models based on the deep learning to model documents, such as Replicated Softmax Model

(RSM) [4] and Deep Boltzmann Machine (DBM) [5]. The basic assumption of topic model, followed by RSM and DBM, is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words [6]. In RSM and DBM, words and latent topics are respectively visible units and hidden units of Restricted Boltzmann Machine. However, compared with Latent Dirichlet Allocation (LDA), the document retrieval accuracy of deep topic models merely improves about 10% on a training dataset of 20-newsgroups. Although deep learning is suitable for visual and speech processing, it may not be suitable for the topic model. A recent study [7] demonstrated some limiting factors of the topic model, such as number of documents, average document length and number of topics, without thinking whether the basic assumption of the topic model is valid or not. The assumption of topic model is not on the basis of biology [8]. We explore a way that the brain processes languages according to neurolinguistics and use the way to model documents, which is different from topic models.

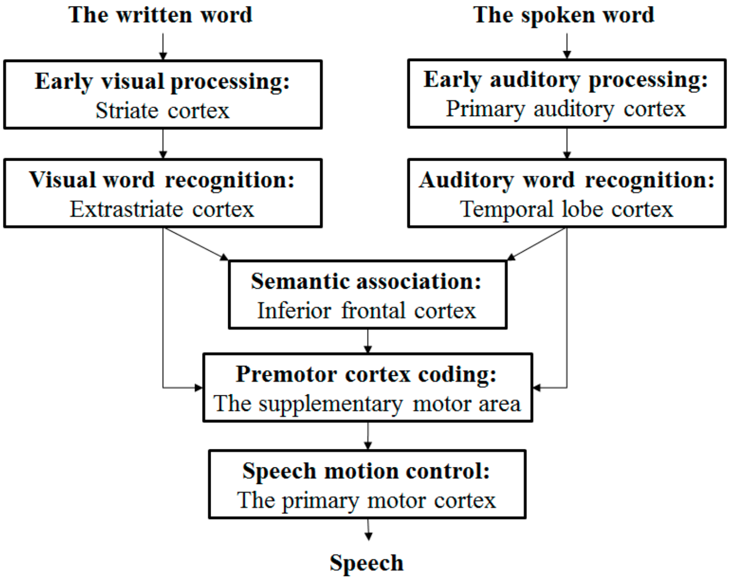

The written word
The spoken word
Early visual processing: Striate cortex
Early auditory processing: Primary auditory cortex
Visual word recognition: Extrastriate cortex
Auditory word recognition: Temporal lobe cortex
Semantic association: Inferior frontal cortex
Premotor cortex coding: The supplementary motor area
Speech motion control: The primary motor cortex
Speech

Figure 1. A language processing in neurolinguistics. This picture marks various processing stages of written and spoken word retelling tasks. The cortical regions associated with above tasks and observed by PET imaging are shown under every stage. When people recognize words by sight or hearing, next there are two processing approaches. One is directly outputting. In addition to the written word and the spoken word, the brain can transform the input signals from vision, audition, *et al.* to words directly. This is the “WYSIWYG” way. Another approach is to use semantic association to find semantically similar words in the brain first and output.

Figure 1 demonstrates that how the brain processes languages [9]. In neurolinguistics, the language function of the brain is closely related to vision, audition, and other sensations. Recently researchers proposed many multimodal neural language models [10–13] for transforming images to texts. These models first recognize objects in the images and their corresponding words, then use a language model to generate texts. Brain and multimodal neural language models can process language using the “WYSIWYG” way, while it is not necessary to consider topics. We treat language as the description of events, such as an image, voice, behavior, or psychological activity rather than topics. A document, regardless of its length, can be viewed as an event. The following sentence is an example of an event.

A man is embracing a woman

It is difficult for us to find the relevance between the words (man, woman, embrace) and some specific topics. However, our brains can clearly imagine the scene described by this sentence, because man, woman, embrace, are very clear in the semantics and related with entity concepts.

In topic model, words are the input features and independent with each other. While semantic association between words is one of the language processing stages shown in Figure 1, which should not be ignored. Furthermore, the premise of the semantic association is that the brain knows the semantics of words. That means before reading words, you must learn them first. For example, the condition of associating "blue" with "green" is that brain knows the two words represent colors.

For document modeling, we propose Event Model (EM). Event Model has two stages. The first stage is word learning and the second stage is dimensionality reduction. Word learning is to learn semantics of words by using context and distributed representations [14,15] of words. Dimensionality reduction is the process that representing documents as low dimensional vectors based on a linear document generating mechanism. We also proposed topic model based on word learning to further compare the document generating mechanisms of Event Model and topic model. Finally we report experimental results on document retrieval tasks.

2. Word Learning

The main task of word learning is finding the semantics of words. Humans' ability with image and sound perception is innate, while words have to be learnt one by one. Words are high-level cognitive abstract entities of objects in nature and semantics of words are the attributes of objects.

In neurolinguistics, there are correlations between words and Figure 2 illustrates these relations. The connections between words are generated from their context and distances between words represent their semantic similarities.

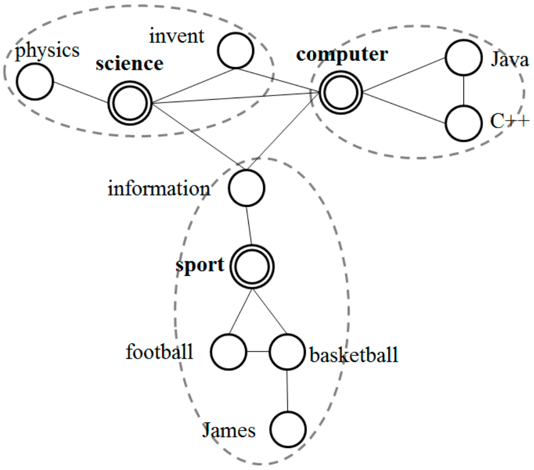


Figure 2. Vector space. For Event Model, we do not deny the existence of topics, but they are only some specific words, such as science, computer and sport in the figure. People perceive topics by semantic association. We use a same way to deal with topics and other words.

A word in the vector space can be expressed as a vector. Then its semantics can be obtained through its context using a statistical language model. CBOW is a statistical language model using distributed representations of words. The objective function is a log-likelihood function as

$$L = \sum_{w \in C} log\, p(w|Context(w)) \tag{1}$$

where *w* is *word*, *C* is corpus and *Context* is the context of *w*. Every input word is mapped to a vector. We can get both language model and word vector by training a neural network. The network consists of three layers: input layer, projection layer, and output layer. The input layer contains 2c word vectors of *Context*(*w*) and *v*(*Context*(*w*)$_1$), *v*(*Context*(*w*)$_2$), ... *v*(*Context*(*w*)$_{2c}$) ∈ $\mathbb{R}^m$. *m* is the length of the word vector. The projection layer sums the 2c word vectors, as

$$X_w = \sum_{i=1}^{2c} v(Context(w)_i) \in \mathbb{R}^m \tag{2}$$

The output layer is a Huffman tree. Its leaf nodes are the words in dictionary and weights are frequencies of words in corpus. For the word *w* in dictionary, there must be a path p^w from the root node to the corresponding leaf node of *w* and l^w is equal the number of nodes in p^w. p^w consists l^w-1 branches. Each branch is a binary classification and presents a probability. Multiply all probabilities together to get *p*(*w*|*Context*(*w*)) and

$$P\big(w\big|Context(w)\big) = \prod_{j=2}^{l^w} p(d_j^w|x_w, \theta_{j-1}^w) \tag{3}$$

where d_j^w denotes the Huffman code of w and $\theta_{j-1}^{\mathrm{w}}$ denotes the corresponding vectors of *j*-1 non leaf nodes. Then use a stochastic gradient ascent algorithm to obtain the word vectors. The word vector generating process is word learning.

3. Dimensionality Reduction

We have already assumed that a document is the description of an event. Next we split an event (a document) into following layers:

1. A *scene* is a fragment or picture of an event, whether concrete or abstract. Scenes correspond to the sentences in the document. Figure 3 shows two specific scenes. The left one can be described by using the sentence “A grey cat is sitting in a house”. The right one is “An orange cat is walking in a house”.

2. An *object* is a basic element which constitutes the scene, such as character, time, place, behavior, emotion, and so on. Objects are commensurate with words in the document.

The event (document) generating process is object-scene-event (word-sentence-document), in which event (document) is the linear superposition of the scenes (sentences). For example, we can merge the two scenes in Figure 3 to generate an event. After filtering out stop words, the dictionary is (grey, orange, cat, sit, lie, house). Let S_1 denote the left scene and S_2 denote the right scene. S_1 = (1, 0, 1, 1, 0, 1) and

$S_2 = (0, 1, 1, 0, 1, 1)$. Let E be the event and $E = S_1 + S_2 = (1, 1, 2, 1, 1, 2)$. Furthermore, scene (sentence) is the linear combination of objects (words).

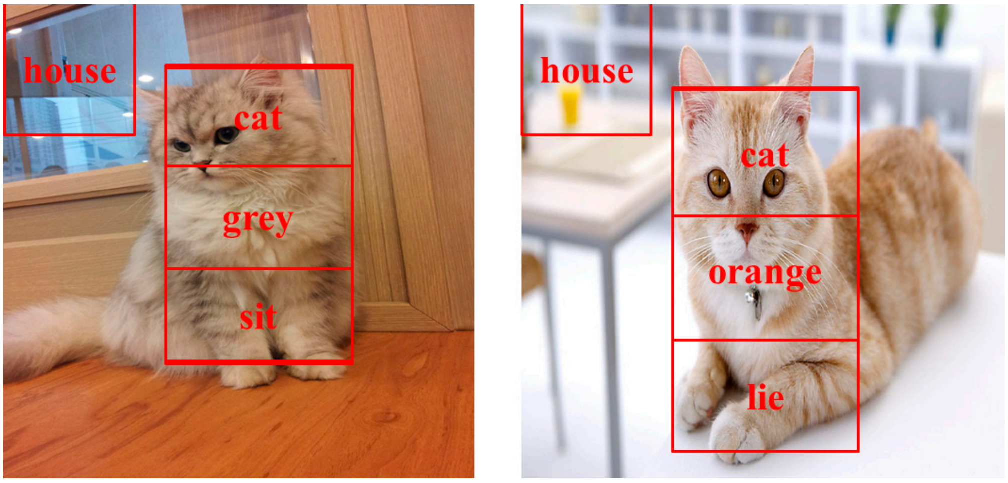
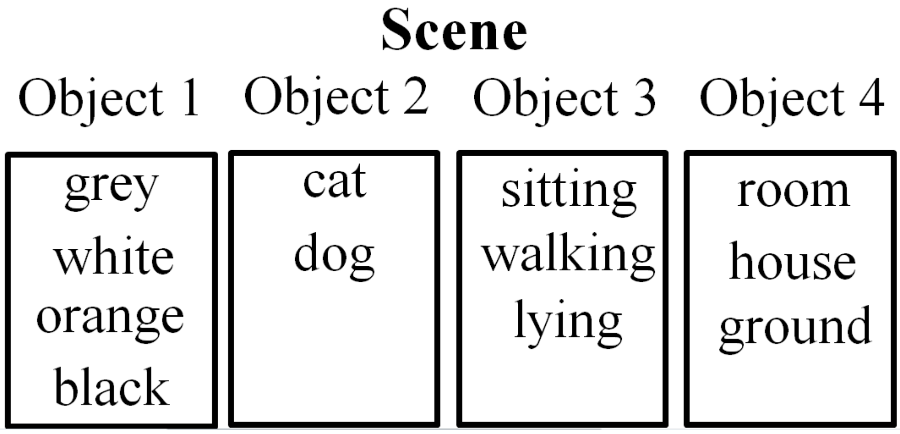


Figure 3. Scenes and objects.

In topic model, words are simply regarded as input features and isolated from each other. However, for Event Model, words denote natural objects and there are associations between the basic properties of objects. It means that one word is linked with others in semantics. When we replace the words in a sentence with other semantically similar words to generate a new sentence, the scenes described by these two sentences are similar, as shown in Figure 4.



Scene

Object 1	Object 2	Object 3	Object 4
grey white orange black	cat dog	sitting walking lying	room house ground

Figure 4. Similar scenes. In the word learning stage, we have already got the semantics of words based on deep learning. The dimensionality reducing mode of topic model is representing a document as an approximate posterior over the latent topics. Event Model uses the semantic similarities to reduce dimensionality.

A word class consists of several semantic similar words. For example, in Figure 4, (grey, white, orange, black) compose the word class “color” and (cat, dog) compose the word class “pet”. Obviously, different word classes can compose sentences, so we let a word class become the intermediate layer between word and sentence. Without considering the order of words, the sentence layer can be removed. As the number of word classes is far less than the number of words, Event Model generates a low dimensional vector representing the document by the process shown in Figure 5.

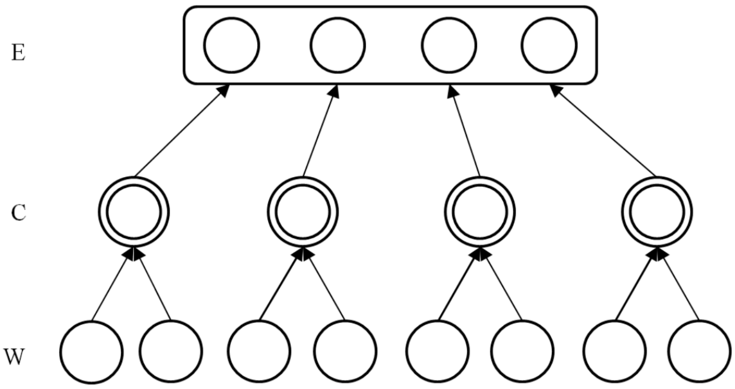


Figure 5. The dimensionality reducing process of Event Model.

W is the word layer, *C* is the word class layer and *E* is the document (event) layer in Figure 5. Let *K* denote the dictionary size and *w* be word. r_i are the distributed representations of w_i, then the network $R = \{r_1,....,r_K\}$. Using K-means algorithm to cluster vectors in *R*, the number of clustering is *N* and record every word's category. *c* is a word class, and $c_i = \{w_m,...,w_n\}$, *where* $i = 1$ to *N* and $m, n \in (1,...,K)$. Let *C* be an *N* dimensional vector, representing the word class that a word belongs to. If the number of words of a document is *D*, then the number of *C* is also *D*. The value of *C* is

$$C_i(j) = \begin{cases} 1 \; if \; w_i \in c_j \\ 0 \; else \end{cases} \tag{4}$$

where $i = 1$ to *D* and j ∈ (1,*N*). It is the process that words are mapped to word classes. Only one dimension in *C* is 1 and others are 0.

We have already introduced that an event is a linear combination of the words, thus the relationship between document and word classes is also linear. Let *E* be an *N* dimensional vector, then we have

$$E = \sum_{i=1}^{D} C_i \tag{5}$$

Normalizing *E*, then each document can be represented with an *N* dimensional vector E_n, computed as

$$E_{max} = max\{E(0), ... E(N)\} \tag{6}$$

$$E_{min} = min\{E(0), ... E(N)\} \tag{7}$$

$$E_n(\mathrm{i}) = \mathrm{a} + \frac{E(i) - E_{min}}{E_{max} - E_{min}} \times (1 - a) \; i = 1 \; to \; N \; 0 \le a < 1 \tag{8}$$

All along, language model and topic model play important roles in different NLP areas. Now we apply the language model to document modeling. The language model is very different with document modeling of course, so we split Event Model into two stages. It is a daily experience of learning first then reading. The main difference between word learning and semantic perception is whether to consider the word order. Order is necessary for word learning. If there are two sentences written as following

room is running happily in the cat
dog is running happily in the house

Then the grammar and semantics of room and dog are same, which means that room and dog belong to a same word class, so do cat and house. That is obviously unreasonable. For semantic perception (used for document retrieval and classification), the word order is not important, such as

room is running happily in the cat
house is running happily in the dog

They convey the same meaning and we do not need to consider their true meanings.

4. Topic Models Based on Word Learning

The document generating mechanisms of topic model and Event Model are completely different. In order to further compare the two mechanisms, we propose a topic model based on word learning.

After obtaining word vectors and clustering them, words can be mapped to word class vector space. Word classes usually have specific concepts as shown in Table 1, so we can explore the relationship between word classes and continue to reduce dimensionality in linear or nonlinear ways.

Table 1. Word classes and their corresponding words.

Word Class	Word	Word Class	Word	Word Class	Word	Word Class	Word
Football	Ronaldo	Basketball	James	Brand	Google	Language	C
	Messi		Kobe		Facebook		C++
	Neuer		Howard		Microsoft		Java
	Dempsey		Duncan		Alibaba		Python

We use the following three ways to further reduce dimensionality: Event Model, WC-LDA, WC-RBM, as shown in Figure 6. WC-LDA and WC-RBM are topic models based on word learning. The dimensionality reduction process of Event Model is same as word-word class and we merge word classes with similar semantics into a bigger word class. For this process, we do not need to analyze the semantics of word classes and just let *N* be a value which is less than the number of word classes. Then generate the document according to the linear mode. The generative model of WC-LDA and WC-RBM is word class-topic-document, which replaces word in bag-of-words model with word class.

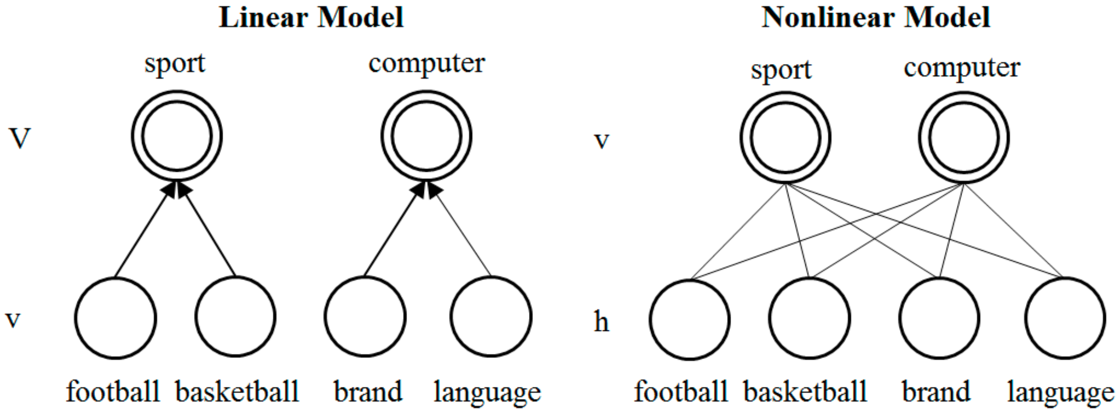


Figure 6. Left: The Relationship of V and v is linear. $V(1) = v(1) + v(2)$, *etc.* **Right:** For WC-LDA, The Relationship of v and h is nonlinear, as shown in Equations 9–12.

For WC-LDA, calculate the model parameters by

$$P(wc|d) = \sum_{t} P(wc|t) \times P(t|d) \tag{9}$$

where *wc* is word class, t is topic and *d* is document. For WC-RBM, firstly change the dimensions which are not equal to 0 in the vector to 1. The energy function of word classes (v) and latent topics (h) is defined as

$$E(v,h) = -\sum_{i} a_i v_i - \sum_{j} b_j h_j - \sum_{i}\sum_{j} v_i w_{i,j} h_j \tag{10}$$

where a_i are bias weights for v and b_j for h. The joint probability distribution over v and h is

$$p(v,h) = \frac{1}{Z} e^{-E(v,h)} \tag{11}$$

The partition function is

$$Z = \sum_{v,h} e^{-E(v,h)} \tag{12}$$

5. Experiments

In this section, we present experimental results on two text datasets: 20-newsgroups and Fudan corpus, and report performance of Event Model and other models on the document retrieval tasks.

5.1. Description of Datasets

20-newsgroups corpus contains 18,845 postings taken from the Usenet newsgroup collection. Each post belongs to exactly one newsgroup. The data was split by date into 11,314 training and 7531 test articles. We further processed the data by removing common stop words, stemming, and deleting the documents which are blank and with less than five words in the training set.

Fudan corpus is a high-quality Chinese corpus provided by Computer Department of Fudan University, including news and professional papers. The original corpus consists of 19,637 documents and is split into training set and test set. It has 20 categories and the difference of documents between different categories is very obvious. This corpus requires preprocessing by merging the training set and test set, and then removing documents which are duplicated and inaccurately classified. Retaining eight main categories and removing others, then dividing the corpus into training set and test set according to the ratio of 2:1. Particular description of two datasets is shown in Table 2. The average length of the documents and the dictionary size of Fudan corpus are significantly greater than those of 20-newsgroups. For LDA, the vocabulary is made up by the 2000 most frequent words in the training set.

Table 2. Particular description of two datasets.

Corpus	**Number of Docs**		**Avg.Length**	**Dictionary**
	Train	**Test**	**of Docs**	**Size**
20-newsgroups	11,277	7513	123	17,279
Fudan Corpus	9824	4948	1751	82,384

5.2. Document Retrieval

For each query, documents in the database were ranked using the cosine distance as the similarity metric. To decide whether a retrieved document is relevant to the query document, we simply check if they have the same class label.

First we compare Event Model with LDA. For Event Model, *N* is 600 for 20-newsgroups and 1000 for Fudan corpus. *a* = 0.1.Then test the effect of N on retrieval accuracy. Finally compare Event Model with WC-LDA and WC-RBM. The results of above three experiments are respectively shown in Figures 7–9.

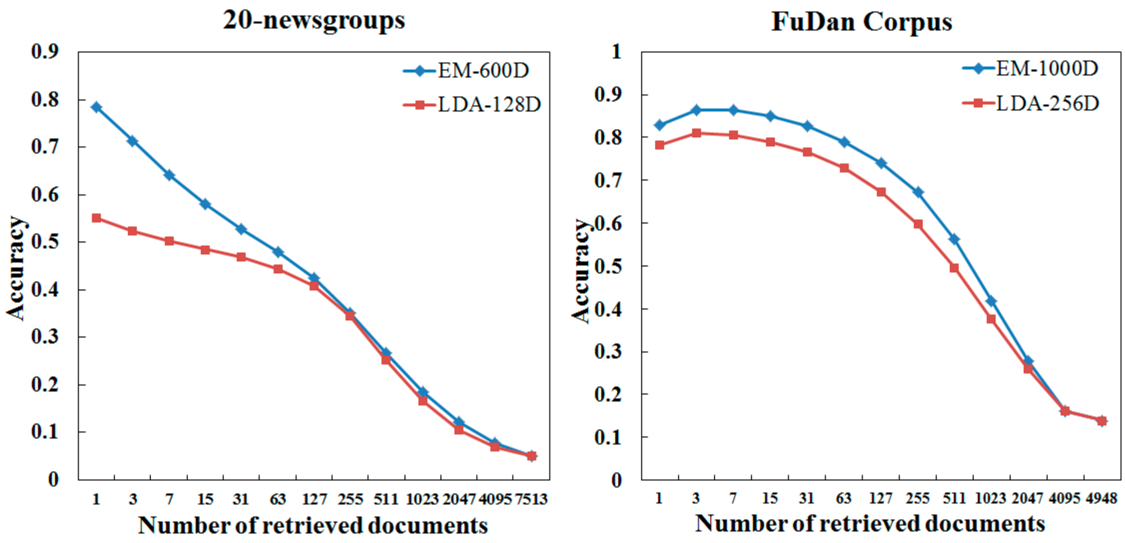


Figure 7. The fraction of retrieved documents in the same class as the query when a query document from the test set is used to retrieve other test set documents. Results are averaged over all 7531 (for 20-newsgroups) and 4948(for Fudan corpus) possible queries. The Event Model performs significantly better than LDA no matter on short documents or long documents, particularly when retrieving the top few documents on 20-newsgroups, up to 25%.

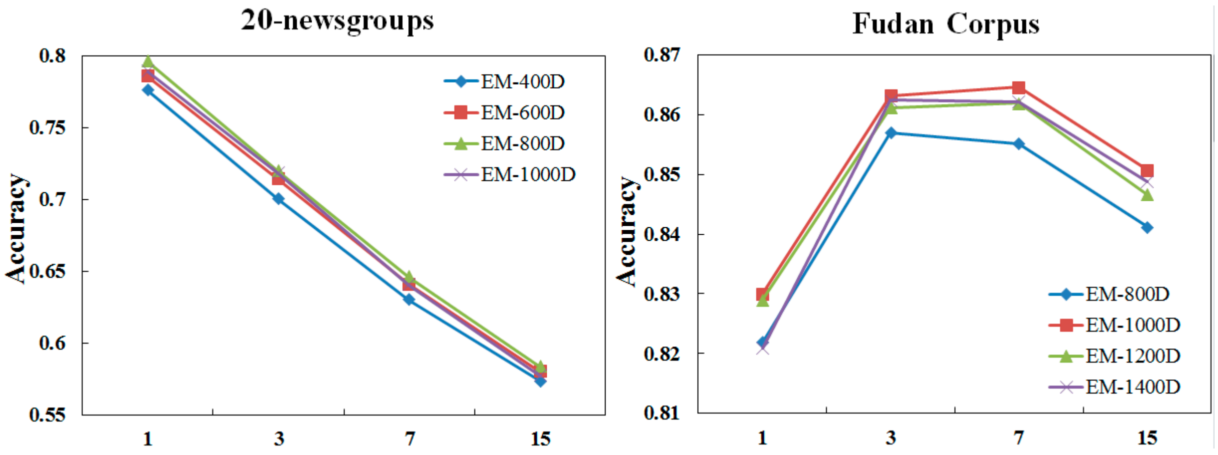


Figure 8. The effect of parameter *N* on retrieval performance for Event Model. *N* can be changed in very large ranges (400 to 1000 for 20-newsgroups and 800 to 1400 for Fudan Corpus), which means that the degree of semantic relatedness between words in the same word class has small influence on retrieval results.

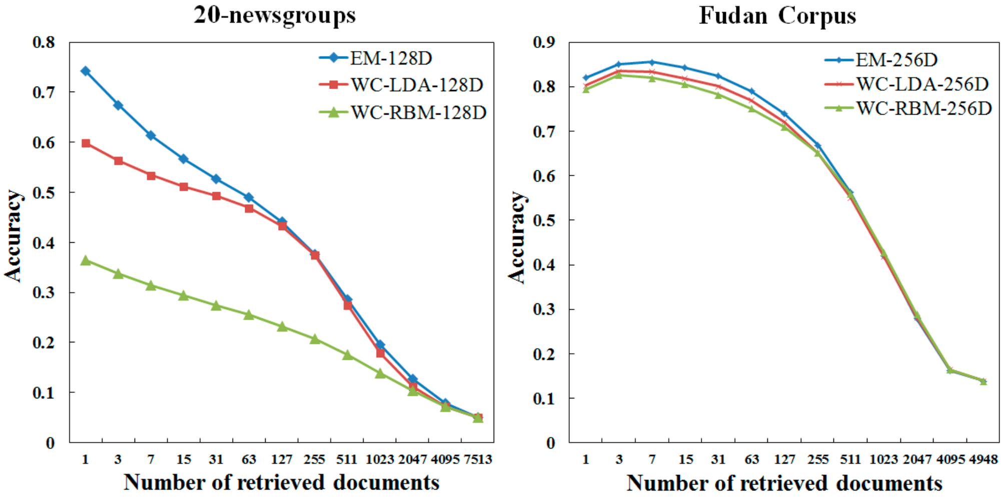


Figure 9. The document retrieving results of EM, WC-LDA and WC-RBM. Dimensionalities of 20-newsgroups and Fudan Corpus are reduced from 600D and 1000D to 128D and 256D. The results of EM remain stable. Although the results of WC-LDA are higher than LDA (up to 4.8% for 20-newsgroups and 2% for Fudan Corpus), they are still far away from EM. For WC-RBM, results on 20-newsgroups are not satisfactory, which may be due to the features of short documents are sparse, while on Fudan Corpus, results are a little worse than WC-LDA.

6. Conclusions

We propose the Event Model to model documents according to neurolinguistics and some language phenomena. The Event Model is based on statistical language model and makes full use of the association between words. Explore the biological language perception process and replace topic with semantics to be the document organization form.

The documents retrieval experimental results on 20-newsgroups and Fudan corpus show that the Event Model is more precise and efficient than the topic model. In addition, we can be clear from the event model that words have abundant information themselves, so it is not suitable for simply using words as input feature. For example, the category labels of documents and images are words and topics perceived by human are also words.

Acknowledgments

This research was supported by National Natural Science Foundation of China (Grant No. 61272427).

Author Contributions

The idea for this research work is proposed by Donghong Sun, the Scala code is achieved by Longhui Wang, and the paper writing is completed by Longhui Wang.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Hinton, G.E.; Salakhutdinov, R.R. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507.
2. Hinton, G. Training products of experts by minimizing contrastive divergence. *Neural Comput.* **2002**, *14*, 1711–1800.
3. Sun, Z.J.; Xue, L.; Yang-Ming, X.U.; Wang, Z. Overview of deep learning. *Appl. Res. Comput.* **2012**, *29*, 2806–2810.
4. Salakhutdinov, R.; Hinton, G. Replicated softmax an undirected topic model. In Proceedings of the NIPS2009, Vancouver, BC, Canada, 6–12 December 2009.
5. Srivastava, N.; Salakhutdinov, R.; Hinton, G. *Modeling Documents with a Deep Boltzmann Machine*; Cornell University Library: Ithaca, NY, USA, 2013.
6. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
7. Tang, J.; Meng, Z.; Nguyen, X.; Mei, O.; Zhang, M. Understanding the Limiting Factors of Topic Modeling via Posterior Contraction Analysis. In Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 21–26 June 2014; pp. 190–198.
8. Boyd-Graber, J.; Mimno, D.; Newman, D. Care and Feeding of Topic Models: Problems, Diagnostics, and Improvements. In *Handbook of Mixed Membership Models and Their Applications*; CRC Press: Boca Raton, FL, USA, 2014.
9. Bear, M.F.; Connors, B.W.; Paradiso, M.A. Chapter 20. In *Neuroscience: Exploring the Brain*, 2nd ed.; Lippincott Williams & Wilkins: Philadelphia, PA, USA, 2004; Volume 3, p. 620.
10. Frome, A.; Corrado, G.S.; Shlens, J.; Bengio, S.; Dean, J.; Ranzato, M.A.; Mikolov, T. DeViSE: A deep visual-semantic embedding model. In Proceedings of the Neural Information Processing Systems (NIPS2013), Lake Tahoe, NV, USA, 5–10 December 2013.
11. Chen, X.; Lawrence Zitnick, C. *Learning a Recurrent Visual Representation for Image Caption Generation*; Cornell University Library: Ithaca, NY, USA, 2014.
12. Fang, H.; Gupta, S.; Iandola, F.; Srivastava, R.; Deng, L.; Dollár, P.; Gao, J.; He, X.; Mitchell, M.; Platt, J.C.; *et al. From Captions to Visual Concepts and Back*; Cornell University Library: Ithaca, NY, USA, 2014.
13. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. *Show and Tell: A Neural Image Caption Generator*; Cornell University Library: Ithaca, NY, USA, 2014.
14. Bengio, Y.; Ducharme, R.; Vincent, P. A neural probabilistic language model. *J. Mach. Learn. Res.* **2003**, *3*, 1137–1155.
15. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. *Efficient Estimation of Word Representations in Vector Space*; Cornell University Library: Ithaca, NY, USA, 2013.