Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 122 (2017) 229–236

Procedia Computer Science

www.elsevier.com/locate/procedia

Information Technology and Quantitative Management (ITQM 2017)

TIDM: Topic-Specific Information Detection Model

Wen Xu[a], Jing He[a,b,*], Bo Mao[a], Youtao Li[c], Peiqun Liu[d], Zhiwang Zhang[e], Jie Cao[a]

[a]NanJing University of Finace & Economics, 3 WenYuan Road, NanJing and 210023, China
[b]*Victoria University, Footscray Park Campus, Melbourne and 14428, Australia*
[c]*JingQi Network Technology, INC. GaoXin District, AnHui and 230088, China*
[d]*NingBo FLO Optical CO .. LTD, GaoXin District, NingBo and 315020, China*
[e]*LuDong University, 186 Middle HongQi Road, YanTai and 264025, China*

Abstract

Nowadays information control and detection on the social network have become a problem that we should solve as soon as possible. Unfortunately, due to the informal expressions, detecting the massive data on the internet is a big challenge based on the traditional text mining methods such as Topic Model. In our paper, we propose a simple 4-Tuple Structure instead of the raw text event which usually contains many meaningless words. Using the word embedding technique, we propose the Topic-Specific Information Detection Model (TIDM) for detecting the specific information. For training the words and idiomatic phrases, we adopt the supervise learning technique: manually constructing a specific Semantic Dataset for training our model. Our experiments based on the Amazon Reviews demonstrate that the TIDM can effectively detect and recognize the information.

Peer-review under responsibility of the scientific committee of the 5th International Conference on Information Technology and Quantitative Management, ITQM 2017.

Keywords: Information control and detection; 4-Tuple Structure; Word Embedding; Topic-Specific Information Detection Model; Semantic Dataset

1. Introduction

During the Information Age, millions of people are surfing the E-Commerce websites for shopping, posting their reviews regarding the products which they have bought. An interesting phenomenon is that even though most reviews of the product are good, only few bad reviews will seriously affect the customers buying intentions. We may wonder if these bad reviews are real or just malicious information. Another rapid growth is the social

* Corresponding author. Tel.: +61 3 9919 4676; fax: +61 3 9919 4908.
E-mail address: jing.he@vu.edu.au.

1877-0509
Peer-review under responsibility of the scientific committee of the 5th International Conference on Information Technology and Quantitative Management, ITQM 2017.
10.1016/j.procs.2017.11.365

network services (eg. Facebook, Twitter etc.), we use these public services for reading news, sharing our opinions, expressing our thoughts. Unfortunately, many unhealthy, illegal, and malicious information is seriously affecting the healthy development of our social network. Thus, it's very important and urgent to deploy a method to help us detect and control the massive social data automatically.

Many popular techniques regarding natural language processing (NLP) have been proposed such as Topic Model based on Latent Dirichlet Allocation [1,2], which usually treat each event in the corpus as the multinomial mixture of topics. However, to find the topics of the corpus is meaningless for us to detect and control information online. Recently, for speech recognition and object classification, using Deep Learning to process these high-dimensional datasets (eg. audios, images) has achieved good results. Thus, the Vector Space Model which has a rich history in NLP has been developing. Generally, we summarize this principle as two categories: Count Model and Predict Model [3].

The Count Model such as Latent Semantic Analysis usually use the count-statistics [4], and the Predict Model such as [5] usually try to predict a word based on its neighbors. In this paper, we adopt the Predict Model technique to train the specific semantic dataset. Two popular ways regarding the Predict Model are: Continuous Bag-of-Words model (CBOW) [6] and Skip-Gram model [11]. For a small dataset (millions of words), our experiments show that the CBOW gets a better result than the Skip-Gram model, reversely, for a large dataset (billions of words), the Skip-Gram model usually gets a better result. The reason behind this shift is obvious, for example, "I have bought several of the Vitality canned dog food products", using the CBOW, it tries to predict the "products" from "I have … dog food", while the Skip-Gram does the reverse. Thus, for the small dataset, the CBOW has eliminated much distributional information and get a better result than the Skip-Gram model.

To the best of our knowledge, we are the first to try to construct specific datasets for information control and detection based on the word embedding. The advantages of our proposed model include: (I)We use the Stanford CoreNLP to construct a 4-Tuple Structure for each coming event of the social media, which is simpler and easier for information control and detection. (II) The training time for a small dataset is quite short (only a few minutes). (III) We can train billions of words with the Skip-Gram model. (IV) Our model can detect any kind of specific semantic corpus.

The rest of this paper is organized as follows. Section 2 shows the related work. Section 3 proposes our entire framework and the TIDM in detail, and introduce the Skip-Gram model with Negative Sampling. Section 4 gives the experimental results. Finally, we conclude our paper in Section 5.

2. Related work

2.1. Topic model

For a long time in NLP, the Topic Model [1,12,13,14] based on the Latent Dirichlet Allocation [1], which assumes that each word in the corpus belongs to a mixture of topics and each one is a distribution over the vocabulary, is popular for solving kinds of applications. Algorithmically, the computations based on the probability distribution is heavy. Thus, it's really difficulty to train a large corpus based on these methods.

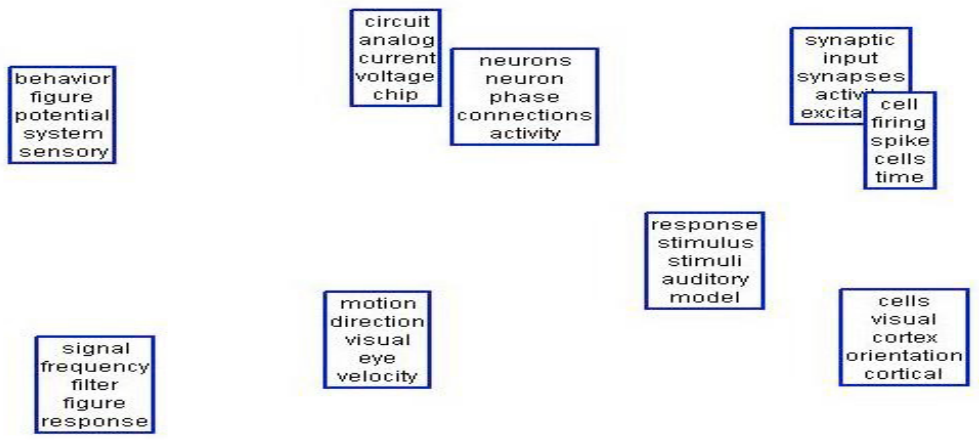

Fig. 1. Topic model visualization

In figure 1, we can see the words regrading one topic end up clustering nearly each other, but the words are not similar enough and it is really hard for us to use the Topic Model for detecting the specific information on social media.

2.2. Word Embedding

With the revival of deep learning in NLP, such as language modeling [5,15], it has been proving effective in many NLP tasks to learn the vector representations of words (which is called word embedding), meanwhile, the distributional semantic models (DSMs) use vectors to keep track of the co-occurring words, which means that the system usually try to learn the similar words with assigning similar vectors. Recently, the Skip-Gram model [6] has proposed an efficient method for training a large corpus (billions of words) with only one day, which is much faster than the previous neural network architectures.

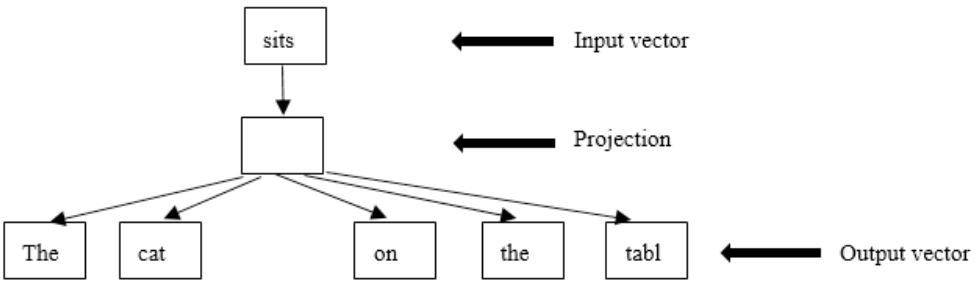

Fig. 2. Skip-Gram model architecture

3. Topic-specific information detection model

In section 3.1, we present the overview of our framework. In section 3.2, we explain why we use 4-Tuple

Structure Event for detecting the topic-specific information, such as abnormal/malicious information, and then develop the skip-gram model with negative sampling [7] for training the Topic-Specific Information Detection Model (TIDM), which aims to learn high-quality vector representations of words and phrases. In section 3.3, we propose the technique details of TIDM and show how to improve the model after measuring the quality.

3.1. Overview of our framework

The goal to construct a semantic dataset is to get labeled information corpus for training vector representations of words and phrases. After our corpus samples become more and more robust, we then apply them to the training work for getting high-quality vectors of words and phrases. Each keyword in the new event/document will be detected by the generated vectors space. The main modules and procedures of our framework are plotted in Figure 3.

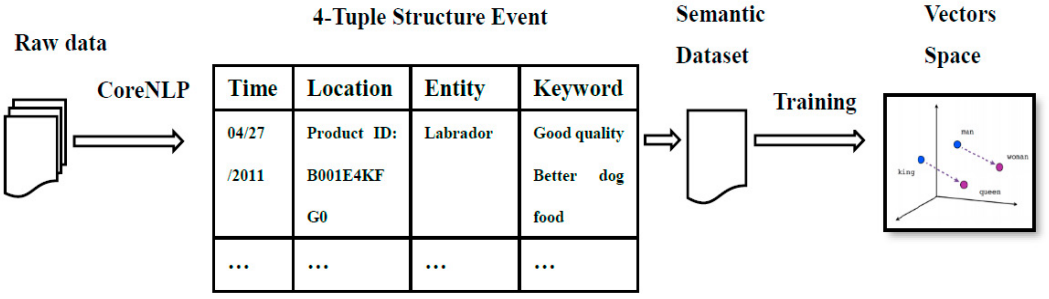

Fig. 3. Overview of the framework

We firstly use Stanford CornNLP to preprocess the given raw stream of the event (e.g. public twitters, online reviews, etc. We regard each of them as one event in this paper). With the help of CorNLP (which provides a bunch of natural language analysis tools), we can easily get the named-entities or hashtags, such as times, locations, people, organizations, etc. If there is not any Named Entity Tagger detected by the CornNLP, we treat this information of the event as pointless babbles and ignore it. After preprocessing, a 4-Tuple Structure is filtered from the raw data and the meaningless words (e.g. a, an, the, is etc.) have been removed.

How could we get the semantic dataset for training our model at first? Here we use the technique of supervised learning, we can manually label some events as malicious information and construct a malicious information dataset if we want to detect malicious information after training or we can use some public datasets which are related to the topic-specific information. Inspired by another hot research area: reinforcement learning, we can use the pre-trained model (entity vectors space) and then use it to train our own new topic-specific model. Regarding this part, we will give more details in section 3.3.

Finally, we use the trained vector spaces of words and phrases to detect the topic-specific information e.g. malicious information in the social network, malicious reviews online, fine food reviews on Amazon, etc. For each keyword and entity in the 4-tuple structure event, we will find the closest words/phrases in the vocabulary, e.g. the input word “dog” will get the most similar words to it, which are “dogs”, “puppy”, “pup”, “pet”, etc. in our system.

3.2. Skip-gram model with negative sampling

Our aim is to train a large corpus by observing the co-occurrences (e.g. “good” and “better”, “great purchase”

and "really like", etc.). Notice that the most frequent words are meaningless for training the vector representations of words, such as "a", "an", "the" which usually occurs a lot in one sentence. In [7], they proposed the subsampling for discarding these frequent words which accelerated learning and improved accuracy. But for us, we want to construct a semantic dataset for topic-specific information detection such as malicious information in the social network, we proposed the 4-Tuple Structure Event in which these meaningless words for training have been discarded. After preprocessing, each event only contains the keywords extracted from the raw data and is much more accurate than using the subsampling [7].

What is the Skip-Gram model [11]? The Skip-Gram model belongs to the predictive models and it try to predict the source context-words from the input/target words, while the Continuous Bag-of-Words (which belongs to the predictive models) model does the inverse. The reason we choose Skip-Gram model is that it treats words/context-target pairs as new observations, thus we can train a large corpus dataset. The goal of the Skip-Gram model is to get Maximization Average:

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{-C\le j\le C, j\neq 0}\log P(W_{i+j}\mid W_i) \tag{1}$$

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{-C\le j\le C, j\neq 0}\log L(V'^{T}_{W_{i+j}} V_{W_i}) + \sum_{k=1}^{K} E_{W_k \sim P_n(W)}[\log L(-V'^{T}_{W_k} V_{W_i})] \tag{2}$$

Where N is the total number of training words, C is the training context size. L is the Logistic Function, K is the negative samples for each sample. V_W is the input word vector and V'_w is the output word vector, and $P_n(w)$ is the Noise Distribution [8, 9]. Given an input word W_i, we want to maximize the probability of output word W_{i+j}, Using the Negative sampling [7], we get the $\log P(W_{i+j}|W_i)$ term in equation (2).

3.3. Topic-Specific Information Detection Model

We proposed a new Topic-Specific Information Detection Model (TIDM) to classify each event, for example, to indicate whether one review on the Amazon website is bad or good or whether the event belongs to the malicious information or normal information. After training and learning the labeled topic-specific corpus (a vocabulary of these words/phrases will be constructed at first), we will get a vectors space of these use-specified words and phrases. For example, after training the review words of find food on Amazon, the input word "error" will get the output words: "errors", "mistake", "issue" etc. based on the cosine distance of the words vectors. We will give more details in next experimental section. To control and detect the specific information, the phrases usually have more accurate and exact meaning than the individual words. Thus, to improve the accuracy of our model, we also train the phrases vectors, for example, the closest tokens to "good quality" are "excellent quality", "good value", "good service", etc. in our system, which means that the review/event with the phrase "good quality" represents the product is well-received by the customer.

How to improve the accuracy of TIDM? The amount and quality of the constructed semantic dataset are the key influence factors. We proposed two methods to improve our model. (I) Testing the quality of the training vectors space with designed test sets. (II) Using the detected results which belong to the topic-specific information as our new training data.

4. Experimental results

In this part, we demonstrate the effectiveness of the novel TIDM for detecting the specific information with public dataset. In our experimentation, we firstly construct a good-reviews dataset which comes from the Amazon Fine Foods Reviews2, because each review in this dataset has a review score, we can easily select these reviews with high scores (score is 5.0). Then we will use this dataset for training in case that we can get a vectors space regarding these keywords of good reviews. Finally, the words and phrases vectors learned by the Skip-Gram model perform precise analogical reasoning which can help us to detect the new event information.

4.1. Dataset and Preprocessing

Now the public datasets of Social Medial such as Twitter are unable to be downloaded dur to modified authorizations, or some have been deleted. Thus, we use 10 years (from October,1999 to October, 2012) data of reviews of fine foods from Amazon (published by Stanford) to evaluate our model. The total number of reviews is 568,454, included product and user information, ratings and plaintext review such as "I have bought several of the Vitality canned dog food products and have

found them all to be of good quality." Our aim is to construct a semantic dataset regarding good/bad reviews, after training the dataset, a vectors space will be generated for detecting each keyword of new event. If most of the keywords can find their synonyms in the trained vocabularies (which means most of the keywords in the event are close to the vocabularies), this illustrates the test event belongs to the dataset, otherwise not.

After the preprocessing with the Stanford CoreNLP, we can easily get the simple and straightforward 4-Tuple Structure. Table 1 shows examples of words preprocessed by our framework. Each word in the event will be detected and the meaningless words will be ignored, and finally the left words will be our keywords (buy, vitality, good, Labrador are the keywords of this example) in the 4-Tuple Structure. The 4-Tuple Structure can describe any text event such as twitter, news and so on, in the 4-Tuple Structure of the Fine Food Reviews, the Product ID is treated as the Location.

Table 1. Preprocessing results in our system

Word	PoS	Lemma	NE
I	PRP	I	0
have	VBP	have	0
bought	VBN	buy	0
Vitality	NN	vitality	0
good	JJ	good	0
Labrador	NNP	Labrador	LOCATION

Table 2. The 4-tuple structure after preprocessing with Stanford CornNLP

Time	Location	Entity	Keywords
04/27/2011	ProductID	Labrador	Buy Vitality Good

4.2. Training topic-specific information dataset

Starting with the constructed good-reviews dataset, we first training the words vectors space, using the Skip-Gram Model with Negative Sampling. We can set different hyperparameters for training the TIDM, for our

training, the learning rate is 0.025, the word vector dimensionality is 300 and K = 10 in equation (2), and window = 8 (max skip length between words). Table 3 gives some examples of the training result, the left is the words from an event, the right is the closest words based on the cosine distance of the two words vectors. The result show that, if the input keywords are in the vocabulary, our system can effectively find their similarities, otherwise not (such as unsavory).

Table 3 Examples of the similar words after training the good-reviews dataset

Word	Similarities in the words space		
Good	Great	Nice	Decent
More	Than	Better	Much
Finicky	Picky	Perfect	Eater
Yummy	Delicious	Tasty	Yummy
Unsavory	Null	Null	Null

As mentioned earlier, the phrases can express more accurately, and the meaning of many phrases is not just the composition of individual words, for example “China Airlines”. Because the dataset is small, the learning rate is 0.025, here the dimensionality is also 300 and we set K = 15 based on the Negative Sampling, and the hyperparameter window = 10.

Table 4. Examples of the similar phrases after training the good-reviews dataset

Phrase	Similarities in the phrases space		
Good quality	Good service	Good value	Excellent value
Great price	Healthy cat food	Great selection	Great variety
Hot sauce	Hot sauces	Marie sharps	Pepper sauce
Good for	Great healthy	Lot of	Long enough
Bad product	Null	Null	Null

5. Conclusions

In this paper, we propose a novel method - Topic-Specific Information Detection Model for detecting the specific information such as malicious or praise information in the social network. We design the 4-Tuple Structure for extracting the keywords for our training. Our experiments demonstrate that using the TIDM can detect and control the specific information effectively based on the words and phrases vectors. Moreover, the designed framework can detect not only the reviews online but also any other event such as the twitter events, news etc.

Acknowledgements

This work is partially supported by JiangSu Science and Technology Program (BE2016178).

References

[1] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan): 993-1022.

[2] Griffiths T L, Steyvers M. Finding scientific topics[J]. Proceedings of the National academy of Sciences, 2004, 101(suppl 1): 5228-5235.
[3] Baroni M, Dinu G, Kruszewski G. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors[C]//ACL (1). 2014: 238-247.
[4] Dumais S T. Latent semantic analysis[J]. Annual review of information science and technology, 2004, 38(1): 188-230.
[5] Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. journal of machine learning research, 2003, 3(Feb): 1137-1155.
[6] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.
[7] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[C]//Advances in neural information processing systems. 2013: 3111-3119.
[8] Gutmann M U, Hyvärinen A. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics[J]. Journal of Machine Learning Research, 2012, 13(Feb): 307-361.
[9] Mnih A, Teh Y W. A fast and simple algorithm for training neural probabilistic language models[J]. arXiv preprint arXiv:1206.6426, 2012.
[11] Guthrie D, Allison B, Liu W, et al. A closer look at skip-gram modelling[C]//Proceedings of the 5th international Conference on Language Resources and Evaluation (LREC-2006). 2006: 1-4.
[12] Rosen-Zvi M, Chemudugunta C, Griffiths T, et al. Learning author-topic models from text corpora[J]. ACM Transactions on Information Systems (TOIS), 2010, 28(1): 4.
[13] Lin C, He Y. Joint sentiment/topic model for sentiment analysis[C]//Proceedings of the 18th ACM conference on Information and knowledge management. ACM, 2009: 375-384.
[14] Rosen-Zvi M, Griffiths T, Steyvers M, et al. The author-topic model for authors and documents[C]//Proceedings of the 20th conference on Uncertainty in artificial intelligence. AUAI Press, 2004: 487-494.
[15] Mnih A, Hinton G E. A scalable hierarchical distributed language model[C]//Advances in neural information processing systems. 2009: 1081-1088.