Topic2Vec: Learning Distributed Representations of Topics

Liqiang Niu, Xinyu Dai, Jianbing Zhang and Jiajun Chen
Natural Language Processing Research Group
Department of Computer Science and Technology
National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210023, China
Email: niulq@nlp.nju.edu.cn, {daixinyu,zhangjb,chenjj}@nju.edu.cn

***Abstract*—Latent Dirichlet Allocation (LDA) mining thematic structure of documents plays an important role in nature language processing and machine learning areas. However, the probability distribution from LDA only describes the statistical relationship of occurrences in the corpus and usually in practice, probability is not the best choice for feature representations. Recently, embedding methods have been proposed to represent words and documents by learning essential concepts and representations, such as Word2Vec and Doc2Vec. The embedded representations have shown more effectiveness than LDA-style representations in many tasks. In this paper, we propose the Topic2Vec approach which can learn topic representations in the same semantic vector space with words, as an alternative to probability distribution. The experimental results show that Topic2Vec achieves interesting and meaningful results.**

***Keywords*-learning; topic; embedding;**

I. Introduction

Modeling text (words, topics and documents) is a key problem in nature language processing (NLP) and information retrieval (IR). The goal is to find short and essential descriptions which enable efficient processing of large systems and benefit basic tasks such as classification, clustering, summarization and estimation of similarity or relevance.

During the past decades, various models and solutions are proposed, such as Bag-of-Words (BOW) [12], *TF-IDF* [33], Latent Semantic Analysis (LSA) [3] and Probabilistic Latent Semantic Analysis (PLSA) [34]. But the best-known model is Latent Dirichlet Allocation (LDA) [4] which describes the hierarchical relationships between words, topics and documents. In LDA, documents are represented as probability distributions over latent topics where each topic is characterized by a distribution over words. However, the probability distribution generated from LDA prefers to describe the statistical relationship of occurrences rather than real semantic information embedded in words, topics and documents. Also LDA will assign high probabilities to high frequency words and those words with low probabilities are hard to be chosen as representatives of topics. But in practice, low probability words sometimes distinguish topics better. For example, LDA will assign higher probability and choose "*food*" as representative other than "*cheeseburger*", "*drug*" other than "*aricept*" and "*technology*" other than "*smartphone*".

Recently, distributed representations with neural probabilistic language models (NPLMs) [1] were proposed to represent words and documents as low-dimensional vectors in one semantic space, and achieved significant results in many NLP and ML tasks [2], [5], [8], [10], [16], [17]. In particular, Word2Vec proposed by [5] could automatically learn concepts and semantic-syntactic relationships between words like vec("*Berlin*") - vec("*Germany*") = vec("*Paris*") - vec("*France*"). Doc2Vec (Para2Vec) proposed by [8] achieves state-of-the-art performance on sentiment analysis. Naturally, in this paper, we want to answer the question that, what will happen if we embed topics in the semantic vector space?

Following the ideas of previously proposed models for words and documents, we propose the model Topic2Vec as shown in Fig. 1. Based on the Word2Vec, we incorporates topics into the NPLM framework for learning distributed representations of topics in the same semantic space with words. Furthermore, words and topics naturally can estimate similarity and relevance with each other such as using cosine function rather than using probability.

In the experiments, we evaluate two different topic representations including embedding of Topic2Vec and probability of LDA in two aspects: listed examples and t-SNE 2D embedding of nearest words for each topic. The experimental results show that our Topic2Vec achieves distinctive and meaningful results compared to LDA.

II. Related Models

A. Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) [4] is a probabilistic generative model that assumes each document is a mixture of latent topics, where each topic is a probability distribution over all words in vocabulary. Briefly, LDA generates a sequence of words as follows:

- For each of the N word w_n in document d:
  - Sample a topic $z_n \sim$ Multinomial(θ_d)
  - Sample a word $w_n \sim$ Multinomial(ϕ_{z_n}).

By Gibbs Sampling [1] estimation, we obtain document-topic probability matrix Θ and topic-word probability matrix Φ. For a new document of arbitrary length, we can infer its involved latent topics and meanwhile we will assign a topic label for each word in the document.

[1] http://gibbslda.sourceforge.net/

978-1-4673-9596-0 /15/$31.00 ©2015 IEEE

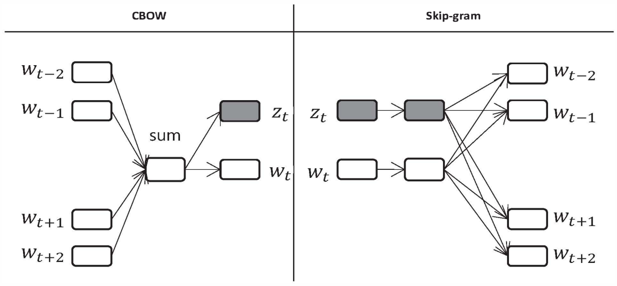

Figure 1. Learning architectures of Topic2Vec.

B. Word2Vec

Inspired by Neural Probabilistic Language Model (NPLM) [1], [5] proposed Word2Vec including Continuous Bag-of-Words (CBOW) and Skip-gram for computing continuous vector representations of words from large data sets.

When training, given a word sequence $D = \{w_1, ..., w_M\}$, the learning objective functions are defined to maximize the following log-likelihoods, based on CBOW and Skip-gram, respectively.

$$\mathcal{L}_{CBOW}(D) = \frac{1}{M}\sum_{i=1}^{M} \log p(w_i|w_{cxt}), \tag{1a}$$

$$\mathcal{L}_{Skip-gram}(D) = \frac{1}{M}\sum_{i=1}^{M} \sum_{-k\leq c\leq k, c\neq 0} \log p(w_{i+c}|w_i). \tag{1b}$$

Here, in Equation (1a), w_{cxt} indicates the context of the current word w_i. In Equation (1b), k is the window size of context. For any variables w_j and w_i, the conditional probability $p(w_j|w_i)$ is calculated using softmax function as follows,

$$p(w_j|w_i) = \frac{\exp(\mathbf{w_j} \cdot \mathbf{w_i})}{\sum_{w\in W} \exp(\mathbf{w} \cdot \mathbf{w_i})}, \tag{2}$$

where $\mathbf{w}$, $\mathbf{w_i}$ and $\mathbf{w_j}$ are respectively the word representations of word w, w_i and w_j, W is the word vocabulary.

III. Topic2Vec

Inspired by word2vec, we incorporate topics and words into the NPLM. We propose Topic2Vec as shown in Fig. 1 for learning distributed topic representations together with word representations. Topic2Vec is also separated in CBOW and Skip-gram situations. For instance, given a word sequence $(w_{t-2}, w_{t-1}, w_t, w_{t+1}, w_{t+2})$, in which w_t is the current word assigned with topic z_t by LDA. The CBOW predicts the word w_t and topic z_t based on the surrounding words $(w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2})$, while the Skip-gram predicts surrounding words $(w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2})$ given current w_t and z_t.

When training, given a word-topic sequence of a document $D = \{w_1 : z_1, ..., w_M : z_M\}$, where z_i is the word w_i's topic inferred from LDA, the learning objective functions can be defined to maximize the following log-likelihoods, based on CBOW and Skip-gram, respectively.

$$\mathcal{L}_{CBOW}(D) = \frac{1}{M}\sum_{i=1}^{M} (\log p(w_i|w_{cxt}) + \log p(z_i|w_{cxt})), \tag{3a}$$

$$\mathcal{L}_{Skip-gram}(D) = \frac{1}{M}\sum_{i=1}^{M} \sum_{-k\leq c\leq k, c\neq 0} (\log p(w_{i+c}|w_i) + \log p(w_{i+c}|z_i)). \tag{3b}$$

Topic2Vec aims at learning topic representations along with word representations. Considering the simplicity and efficient solution, we just follow the optimization scheme that used in Word2Vec [5]. To approximately maximize the probability of the softmax, we use Negative Sampling without Hierarchical Softmax [6]. Stochastic gradient descent (SGD) and back-propagation algorithm are used to optimize our model. By the way, complexity of our Topic2Vec is linear with size of dataset, same with Word2Vec.

IV. Experiments

A. Dataset

We use the English Gigaword Fifth Edition[2] as our training data for learning fundamental word and topic representations. We randomly extract part of documents and construct our training set described as follows: we chose 100,000 documents, where each consists of more than 1,000 characters from subfolder ltw_eng (Los Angeles Times) containing 411,032 documents. Besides, we eliminate those words that occur less than 5 times and the stop words. In the end, training set contains about 42 million words and the vocabulary size is 102,644.

B. Evaluation Methods

In experiments, we run Topic2Vec in Skip-gram and learn topic representations together with word representations. And then we evaluate topic representations via comparing Topic2Vec with LDA in two aspects: (1) we select most related topics or words conditioned on selected topics and (2) we embed these related words or topics in 2D space using t-SNE [32]. During the process, we cluster words into topics as follows:

- LDA: each topic is a probability distribution over words. We select the top $N = 10$ words with highest conditional probability.
- Topic2Vec: topics and words are equally represented as the low-dimensional vectors, we can immediately calculate the cosine similarity between words and topics. For each topic, we select higher similarity words.

[2] https://catalog.ldc.upenn.edu/LDC2011T07

	Topic_6		Topic_19		Topic_27		Topic_47	
LDA	word	prob.	word	prob.	word	prob.	word	prob.
	food	0.027	drug	0.031	medical	0.033	dog	0.011
	restaurant	0.008	drugs	0.019	hospital	0.024	garden	0.009
	eat	0.008	cancer	0.019	care	0.019	tree	0.009
	more	0.005	study	0.011	patients	0.018	dogs	0.009
	chicken	0.005	patients	0.011	doctors	0.016	plants	0.008
	cooking	0.005	treatment	0.009	health	0.013	trees	0.008
	eating	0.005	fda	0.009	doctor	0.009	animal	0.007
	one	0.005	heart	0.008	patient	0.009	plant	0.007
	good	0.005	risk	0.008	surgery	0.008	animals	0.006
	foods	0.005	more	0.007	center	0.008	zoo	0.006
Topic2Vec	word/topic	cos.	word/topic	cos.	word/topic	cos.	word/topic	cos.
	cheeseburgers	0.564	topic_62	0.618	topic_19	0.519	dogwood	0.498
	meatless	0.535	aricept	0.531	topic_62	0.478	dogwoods	0.494
	smoothies	0.534	topic_27	0.519	neonatal	0.466	topic_33	0.485
	topic_95	0.533	memantine	0.514	topic_13	0.457	bark	0.484
	meatloaf	0.530	enbrel	0.512	anesthesiologists	0.445	fescue	0.483
	tastier	0.530	gabapentin	0.511	anesthesia	0.439	aphids	0.478
	topic_52	0.527	colorectal	0.509	reconstructive	0.437	mulched	0.478
	cheeseburger	0.525	prilosec	0.507	comatose	0.437	azaleas	0.477
	concoctions	0.522	placebos	0.507	hysterectomy	0.433	shrub	0.475
	vegetarians	0.515	intravenously	0.504	ventilator	0.432	camellias	0.472
	Topic_53		Topic_67		Topic_79		Topic_93	
LDA	word	prob.	word	prob.	word	prob.	word	prob.
	government	0.022	www	0.028	computer	0.016	russia	0.028
	africa	0.015	com	0.023	technology	0.010	russian	0.027
	people	0.015	hotel	0.018	phone	0.009	putin	0.017
	african	0.011	travel	0.015	software	0.009	soviet	0.013
	country	0.009	trip	0.011	digital	0.008	moscow	0.012
	international	0.008	night	0.010	apple	0.008	president	0.010
	darfur	0.007	per	0.009	use	0.007	country	0.007
	sudan	0.007	day	0.008	system	0.006	former	0.007
	south	0.007	tour	0.008	microsoft	0.006	state	0.007
	human	0.007	cruise	0.007	up	0.006	union	0.006
Topic2Vec	word/topic	cos.	word/topic	cos.	word/topic	cos.	word/topic	cos.
	mozambique	0.428	fairmont	0.569	wirelessly	0.584	topic_88	0.469
	uganda	0.423	motorcoach	0.553	handhelds	0.573	boris	0.435
	ghana	0.419	stateroom	0.547	desktops	0.572	leonid	0.411
	addis	0.417	uniworld	0.540	pda	0.566	dmitry	0.404
	darfur	0.412	maarten	0.533	smartphone	0.566	vladimir	0.397
	burundi	0.408	tourcrafters	0.529	megabyte	0.562	mikhail	0.397
	lanka	0.407	wyndham	0.528	macbook	0.556	dmitri	0.396
	congo	0.406	cunard	0.527	handheld	0.549	alexei	0.394
	ababa	0.403	safaris	0.522	treo	0.549	eduard	0.392
	darfurians	0.402	trafalgar	0.518	modems	0.548	kasparov	0.391

Figure 2. Nearest words and topics for each selected topic. Words are listed with conditional probabilities in LDA while words and topics are listed with calculated cosine similarity in Topic2Vec.

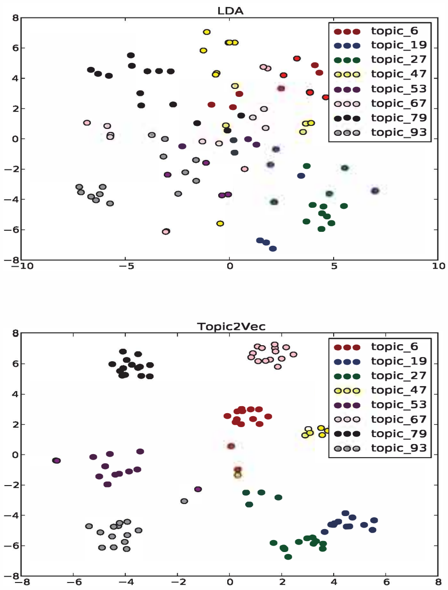

Figure 3. t-SNE 2D embedding of the nearest word representation for each topic in LDA (above) and Topic2Vec (below).

C. Analysis of Results

Fig. 2 shows top 10 nearest words from LDA and Topic2Vec for eight typically selected topics, respectively. We now give more detailed analysis to understand the difference between them. As shown in Fig. 2, in Topic_19, LDA returns the words like "*drug*", "*drugs*", "*cancer*" and "*patients*", while Topic2Vec returns "*aricept*", "*memantine*", "*enbrel*" and "*gabapentin*". In Topic_27, LDA returns the words of "*medical*", "*hospital*", "*care*", "*patients*" and "*doctors*", while Topic2Vec returns "*neonatal*", "*anesthesiologists*", "*anesthesia*" and "*comatose*". We only know that Topic_19 and Topic_27 share the same topic about "*patients*" or "*medical*", but we can't get their further difference from the results of LDA. But from the result of Topic2Vec, we can easily discover that Topic_19 focuses on a more specific topic about drugs ("*aricept*", "*memantine*", "*enbrel*" and "*gabapentin*"), while Topic_27 focuses on another specific topic about treatment ("*anesthesiologists*", "*anesthesia*" and "*comatose*"), they are absolutely different. Obviously, Topic2Vec presents more distinguished results between two similar topics.

Fig. 3 shows the 2D embedding of the corresponding related words for each topic by using t-SNE. Obviously, Topic2Vec produces a better grouping and separation of the words in different topics. In contrast, LDA does not produce a well separated embedding, and words in different topics tend to mix together.

In summary, for each topic, words selected by Topic2Vec are more typical and representative compared to those returned by LDA. Eventually, Topic2Vec can better distinguish different topics.

V. Conclusions and Future Work

In this paper, via integrating NPLM, Word2Vec and LDA, we are the first to propose the Topic2Vec which successfully embeds latent topics in the same semantic vector space with words. In principle, our purpose clearly aims at learning new fashion embedded topic representation by Topic2Vec. From the observation of experiments, Topic2Vec presents more distinguished results than LDA and we have the conclusion that Topic2Vec can model topics better.

But now, we just qualitatively evaluate the performance of Topic2Vec compared to LDA and emphasize that they are inherently different. In the future, we will quantitatively do more detailed analysis about their difference, including exploiting Topic2Vec for traditional NLP tasks.

Acknowledgments

References

[1] Y. Bengio, R. Ducharme, P. Vincent and C. Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137-1155.

[2] R. Collobert and J. Weston. 2008. A unified architecture for natural language processing: deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine learning*, ICML '08, pages 160-167, New York, NY, USA. ACM.

[3] T. K. Landauer, P. W. Foltz, and D. Laham. 1998. An introduction to latent semantic analysis. Discourse processes 25.2-3 (1998): 259-284.

[4] D. M. Blei, A. Y. Ng and M. I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993-1022.

[5] T. Mikolov, K. Chen, G. Corrado and J. Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

[6] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111-3119.

[7] T. Mikolov, W. Yih, and G. Zweig. 2013c. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746-751, Atlanta, Georgia, June. Association for Computational Linguistics.

[8] Q. Le and T. Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31th International Conference on Machine Learning*, Beijing, China.

[9] R. Socher, E. H. Huang, J. Penniington, A. Y. Ng and C. D. Manning. 2011a. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Advances in Neural Information Processing Systems 24*.

[10] E. H. Huang, R. Socher, C. D. Manning and A. Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Annual Meeting of the Association for Computational Linguistics*.

[11] J. Pennington, R. Socher and C. D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Conference on Empirical Methods in Natural Language Processing*.

[12] Z. S. Harris. 1954. Distributional structure. *Word*.

[13] A. L. Mass, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng and C. Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*.

[14] F. Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Survey*, 34(1): 1-47.

[15] D. Yogatama, M. Faruqui. C. Dyer and N. A. Smith. 2014. Learning word representations with hierarchical sparse coding. In *NIPS Deep Learning and Representation Learning Workshop*, Montréal, Quebec, December 2014.

[16] A. Mnih and K. Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In *Advances in Neural Information Processing Systems 26 (NIPS 2013)*.

[17] A. Mnih and G. E. Hinton. 2009. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems*, pages 1081-1088.

[18] F. Morin and Y. Bengio. 2005. Hierarchical probabilistic neural network language model. In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, pages 246-252.

[19] J. Turian, L. Ratinov and Y. Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384-394. Association for Computational Linguistics.

[20] P. D. Turney and P. Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*.

[21] Y. Liu, Z. Liu, T. Chua and M. Sun. 2015. Topical word embeddings. In *Association for the Advancement of Artificial Intelligence*.

[22] P. Pantel and M. Rey. 2005. Inducing ontological co-occurrence vectors. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 125-132.

[23] L. Finkelstein, E. Gabrilocivh, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. 2001. Placing search in context: the concept revisited. In *Proceedings of the 10th International Conference on Word Wide Web*, WWW '01, pages 406-414, New York, NY, USA. ACM.

[24] W. Blacoe and M. Lapata. 2012. A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 546-556.

[25] J. Reisinger and R. J. Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Human Language Technologies: The 2010 Annual Conference of the North America Chapter of the Association for Computational Linguistics*, HLT '10, pages 109-117, Stroudsburg, PA, USA. Association for Computational Linguistics.

[26] N. Madnani and J. Tetreault. 2012. Re-examining Machine Translation Metrics for Paraphrase Identification. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 182-190, Montréal, Canada, June. Association for Computational Linguistics.

[27] B. Dolan, C. Quirk and C. Brockett. 2004. Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources. In *COLING*, 2004.

[28] S. Qiu, Q. Cui, J. Bian, B. Gao and T. Liu. 2014. Co-learning of Word Representations and Morpheme Representations. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 141-150, Dublin, Ireland, August 23-29 2014.

[29] A. Rajaraman. 2008. More data usually beats better algorithms. *Datawocky Blog*.

[30] N. Kalchbrenner, E. Grefenstette and P. Blunsom. 2014. A Convolutional Neural Network for Modelling Sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*.

[31] R. Kiros, R. Zemel, R. Salakhutdinov. 2014. A Multiplicative Model for Learning Distributed Text-Based Attribute Representations. In *Neural Information Processing Systems (NIPS)*, Montreal, Canada, December 2014.

[32] V. Maaten, Laurens, and G. Hinton. 2008. Visualizig data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605): 85.

[33] G. Salton and M. J. McGill. 1983. Introduction to Modern Information Retrieval. McGraw-Hill.

[34] T. Hofmann. 1999. Probabilistic Latent Semantic Analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*.