ISSN 2007-9737

A Novel Multimodal Deep Neural Network Framework for Extending Knowledge Base

Zhao Yu[1], Gao Sheng[1], Patrick Gallinari[2], Guo Jun[1]

[1] Beijing University of Posts and Telecommunications, Beijing,
China

[2] LIP6, Universite Pierre et Marie Curie, Paris,
France

{yu-zhao, gaosheng, guojun}@bupt.edu.cn, patrick.gallinari@lip6.fr

Abstract. Knowledge base is a very important database for knowledge management, which is very useful for Question Answering, Query Expansion and other AI tasks. However, due to the fast-growing knowledge on the web and not all common knowledge expressed in the text is explicit, the knowledge base always suffers from incompleteness. Recently many researchers are trying to solve the problem as link prediction, only using the existing knowledge base, however, it is just knowledge base completion without adding new entities, which emerges from unstructured text not in existing knowledge base. In this paper, we propose a multimodal deep neural network framework that trying to learn new entities from unstructured text and to extend the knowledge base. Experiments demonstrate the excellent performance.

Keywords. Extending knowledge base, deep neural network, word embedding, embedding-based model.

1 Introduction

Knowledge base is a very useful database for knowledge management, such as *Wordnet* [11], *Freebase* [1], etc. They consists of a great amount of knowledge facts, the form of which is triplet like (*left-entity, relationship, right-entity*). It means there have the relationship between the left-entity and right-entity. Knowledge base is very important and is very useful for human reasoning, question answering, query expansion, and other AI tasks. But it usually suffer from incompleteness due to a large volume increasing knowledge and lack of additional new entity.

There have much work to complete knowledge base [4, 2, 12, 3]. However, most of the models just do knowledge base completion that to predict how likely some additional facts (triples) are held by only using existing knowledge in KBs. They can not add additional new entity. In this paper, we propose a new framework for extending knowledge base that can add additional new entity to knowledge base, by connecting the free text and knowledge base.

Our contributions in this paper are followings:

- We propose a new perspective to extend knowledge base by add new additional entity from free text;
- We present a framework to extend knowledge base with DNN, word embedding and entity latent representation;
- Empirical experimental results demonstrate that our models perform excellently.

In the rest of the paper, we first show some related work in Section 2, and then we introduce a framework for extending knowledge base in Section 3. In Section 4, we show several experiments on real data sets. We finally conclude by sketching some future work directions in Section 5.

doi: 10.13053/CyS-20-3-2472

2 Related Work

We briefly introduce some of the related work in this section.

2.1 Word Embedding

Word representation (Distributed Representation, Word Embedding) was first proposed by Hinton [7]. Word representation is learned a vector from a large number of unlabeled free text corpus by language model. The idea of using neural network to train language model was first presented by Xu [15]. Word representations learned by neural network language models can be used for many NLP tasks such as POS tagging, chunking, named entity recognition, semantic and syntactic similarity. And subsequently many language models are proposed in [5]. (Mikolov et al. (2013) proposed two state-of-the-art models (CBOW and Skip-gram)[1] to capture better semantic and syntactic word similarity [8, 9, 10].

2.2 Embedding-based Model

Many recent energy-based embedding models [7, 8, 9] are proposed, focusing on increasing expressivity, but resulting in higher computational cost. Bordes et al. [10] proposed a simpler model (TransE), however, the drawback of it is that it only can model linear triplets. Wang et al. proposed an extended model of TransE, TransH, which faces the same issue. Several models have been recently proposed for that purpose [16].

2.3 Deep Neural Network

The Deep Learning models is very hot recently, such as DBN [13] , DBM [14], Deep-autoencoder, etc. They have been used in many applications and perform excellent, such as speech recognition, image recognition and natural language processing, etc.

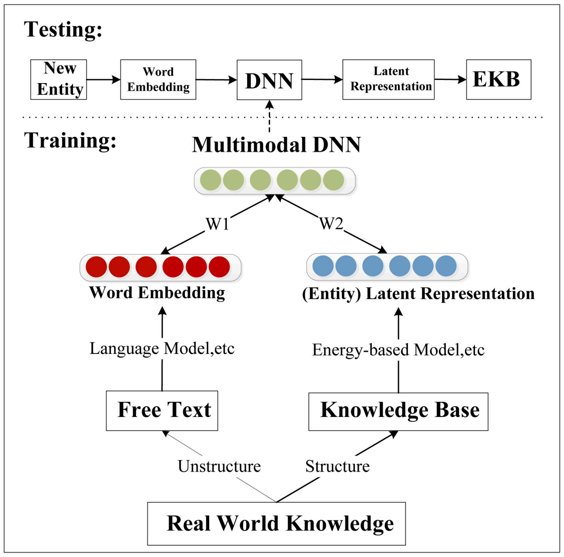

Fig. 1. Framework of Connecting Free Text and Knowledge Base with Multimodal Deep Neural Network (DNN) for Extending Knowledge Base (EKB)

3 A Framework for Extend Knowledge Base with Multimodal Deep Neural Network, Word Embedding and Entity Latent Representation

In this section, we will introduce a framework that used to extend knowledge base, as in Fig. 1. First of all, we will present the approach of word embedding and another latent model that used to learn the entity latent representation. And then we will present a multimodel deep neural network. Finally we show the implementation to extend knowledge base.

3.1 Word Representation learned by Language Model from Unstructured Data (Free Text) and Encoding Structured Data (Knowledge Base)

Word representation are word vector that can be used as features for other models. In this work, we choose two typical word representations. One is the approach of (Mikolov et al., 2013), known as *Word2vec*. It used two main language models: n-grams and skip grams for

[1] https://code.google.com/p/word2vec/

learning. Here resulting word vectors have some interesting character and capture many linguistic regularities. Take the famous two as example here, vector operations vector('China') - vector('Beijing') + vector('Tokyo') results in a vector that is very close to vector('Japan'), and vector('king') - vector('man') + vector('woman') is close to vector('queen'). In Word2vec, it has offered more than 1.4M pre-trained entity vectors with naming from *Freebase*. The other work is proposed in [6], as SINNA. We also directly utilize the resulting work vectors from it, pre-trained word embedding of *Wordnet*, and how to train them is not our focus in this paper.

In order to connect latent knowledge of Knowledge Base with word representations learned by language model from free text, we propose to encode the entities and relationships of the triplets into latent embedding space. This allow us to build a model, like deep neural network model, which can calculate the plausibility of additional new triplets for extending knowledge base. As introduced in Section 2, several models have been recently proposed for that purpose. In this work, we choose to follow the Pairwise-interaction Differentiated Embedding model (PIDE) [16], which is claimed to performed excellent.

In the next, we briefly review the Pairwise-interaction Differentiated Model. Given a training set H of triplets `(subject s, predicate p, object o)`, $s, o \in E$ (the set of entities) and $p \in R$ (the set of predicate relationships), the PIDE model learns entities and predicate relationships latent vector representations. It considers that the reason why a triplet `(subject s, predicate p, object o)` should be hold is due to its inner pairwise interactions: `(subject, object)`, `(subject, predicate)` and `(predicate, object)`, each of which should contribute to the confidence of the triplet. The more closer between `subject` and `object` (`(subject and predicate)` or `(predicate and object)`), the more likely the knowledge fact triplet hold. In addition, it also consider that the entity in knowledge triplet should possess semantic and syntactic information, and the predicate relationship should have syntactic information. The semantic information of entity represents its content information, while the syntactic information represents their position and order information. So it can obtain the interaction function between subject entity and object entity $f_{sem+syn}(s,o)$ using their semantic and syntactic information. Also it can get the interaction function between subject entity and predicate relationship $f_{syn}(s,p)$ and the interaction function between predicate relationship and object entity $f_{syn}(p,o)$ using their syntactic information. The PIDE model scoring function is defined as follows:

$$\begin{aligned} g(s,p,o) &= f_{sem+syn}(s,o) + f_{syn}(s,p) + f_{syn}(p,o) \\ &= [e_{s1}, e_{s2}] \begin{bmatrix} e_{o1} \\ e_{o2} \end{bmatrix} + [e_{s2}] \cdot [e_{p1}] + [e_{o2}] \cdot [e_{p2}] \end{aligned}, \tag{1}$$

where $e_s = [e_{s1}, e_{s2}]$, $e_o = [e_{o1}, e_{o2}]$ and $e_p = [e_{p1}, e_{p2}]$,$e_{s1}, e_{s2}, e_{p1}, e_{p2}, e_{o1}$ and $e_{o2} \in \mathbb{R}^k$. The model is trained with contrastive max-margin optimization criterion. The main idea is that each triplet in the training set should receive a higher score than a corrupt triplet in which one of the entities is replaced with a random entity.

3.2 Learning Multimodel DNN with Connecting Word Embedding and Entity Latent Representation

As we have introduced above, we could learning two type of latent representations for a same word (entity)[2]. The word embedding of the entity is learning from free text by *word2vec*. We also learn the another latent representation of the word (entity) from structured data (knowledge base). Theoretically both the two latent representation of the same entity should indicate its semantic information. But its two semantic vector expressions would be different because they derive from two different latent semantic space, one is free corpus ant the other is structured knowledge base. So we are trying to find a pipeline that can somehow connect them or transform them from each other. In this paper, we propose to use Deep neural network model for connecting word embedding and entity latent representation. We use a great amount of pre-learned pairs (*word*

[2] In Free text, we call it *word*, while we call it *entity* in knowledge base. Indeed it it the same.

embedding, entity latent representation) to train the deep neural network (DNN) model. By this, we could build a pipeline across the two latent semantic space. If there have a additional new entity which exists in free text but does not exist in knowledge base, namely the word embedding of the additional new entity in free text is known and not in knowledge base. Through the DNN model as pipeline, we can obtain latent representation of the additional new entity by put the its word embedding as input.

3.3 Implementation for Extending Knowledge Base

In the next, we present the workflow of the implementation and show how to extend the knowledge base from the beginning. As in the Fig.1, here comes a new entity at testing phase. We find out its word embedding pre-learned from free text. We put the word embedding as input into Multimodal DNN. Through the model, we obtain the entity representation that would be the latent semantic vector in knowledge base space. Finally we can calculate the scoring function of all the possible triplets with the new entity. The candidate triplet holds if its score is larger than the threshold. So we extend the knowledge base in this way.

4 Experiments

Our proposed framwork is evaluated on the data sets extracted from WordNet (Miller 1995) and Freebase (Bollacker et al. 2008) for extend knowledge base. In this section, we will introduce the datasets, evaluation metrics and baseline for the experiment. And then we present the experiments results for extend knowledge base.

4.1 Data Set

WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. Examples of triplets are (_payment_NN_1, _hyponym, _recompense_NN_1) or (_flint_NN_3, _part_of, _wolverine_state_NN_1).[3] Here we do not yet distinguish the same entity with different part-of-speech tag because their word representations learn by language model from free text is unique. We create a data set with WordNet and Word Representations (which were trained for about 2 month over *Wikipedia*) provided by *SENNA* [6][4]. This data set include 47176 triplets with 10556 entities and 18 relationships which were randomly split into three parts (Train, Valid, Test). The training data set includes 35,016 triplets with 8,541 entities and 18 relationships, which would be further split into three parts for PIDE model learning in KB. The valid data set includes 4,283 triplets with 723 extra new entity which would be not emerge in training data set. *LEFT-NEW, RIGHT-NEW, BOTH-NEW* indicated that the left entity or right entity or both entities of triplets were new entities. The test data set includes 7,877 triplets with 1,292 extra new entity. This data set is denoted *WN-WR10K* in the rest of this section

Freebase is a large collaborative knowledge base of general facts, currently including around 1.2 billion triplets and more than 80 million entities. We created the data set with *FB15K* [3] extracted from Freebase and Freebase entities vectors[5] (word represenation) which trained on 100B words from various news articles. *FB15K* is subset of Freebase including 592,213 triplets with 14,951 entities and 1,345 relationships. We remove the entities which do not have the word representation from FB15K. This resulted in 471,648 triplets with 13,868 entities and 1271 relationships. Likewise we deal with it further as *WordNet*. This data set is denoted *FB-WR14K* in the rest of this section which is shown in Table 1.

4.2 Evaluation Metrics

In the experiment, we use the ranking criteria (Bordes et al. 2011) for evaluation. Firstly for each test triple, we remove the subject entity and replace it by each of the entities of the dictionary in turn.

[3] The entities of WordNet are denoted by the concatenation of a word, its POS tag and a digital number. The number refers to its sense. E.g. "_payment_NN_1" encodes the first meaning of the noun "payment".

[4] http://ml.nec-labs.com/senna/

[5] https://code.google.com/p/word2vec/

Table 1. Statistics of the data sets used for the experiment and extracted from the two knowledge bases, *WordNet* and *Freebase* [3]

DATASET	WN-WR10K		FB-WR14K	
ENTITIES	10556		13868	
RELATIONSHIPS	18		1271	
TRAIN. EX.	#ENTITIES 8541		#ENTITIES 11964	
	#TRAIN-TN 33216	35016	#TRAIN-TN 344700	354700
	#TRAIN-VD 800		#TRAIN-VD 5000	
	#TRAIN-TT 1000		#TRAIN-TT 5000	
VALID. EX.	#NEW ENTITIES 723		#NEW ENTITIES 891	
	#LEFT-NEW 1774	4283	#LEFT-NEW 26898	54704
	#RIGHT-NEW 1777		#RIGHT-NEW 25849	
	#BOTH-NEW 732		#BOTH-NEW 1957	
TEST. EX.	#NEW ENTITIES 1292		#NEW ENTITIES 1013	
	#LEFT-NEW 3200	7877	#LEFT-NEW 30941	62244
	#RIGHT-NEW 3204		#RIGHT-NEW 28494	
	#BOTH-NEW 1473		#BOTH-NEW 2809	

The function values g(s, p, o) of the negative triples would be computed by the related models and then sorted by descending order. We can obtain the exact rank of the correct entity in the candidates. Similarly, we repeat the whole procedure while removing the object entity instead of the subject entity of the test triple. We have three kinds of test triplets: LEFT-NEW, RIGHT-NEW AND BOTH NEW. So we use two evaluation metrics for comparison in the three data sets: the mean of those predicted ranks and the proportion of correct entities ranked in the top 10 (Hits@10(%)). Finally they are LR: Left Rank[6]; LH10: Left Hits@10(%); RR: Right Rank; RH10: Right Hits@10(%); MR: Mean Rank; MH10: Mean Hits@10(%).

4.3 Baseline and Configuration

In this experiment, we choose two baseline models for comparison. The first one is the Un-transform model, in which we directly utilize the word embedding as the entity latent representation without any model as the pipeline to connect the two latent semantic space. The other one is linear transform model namely shallow neural network, using artificial neural network (ANN). We select 4 layers for the ANN model and use BP learning algorithm to train the model.

4.4 Learning the Entity Representation with PIDE Model

For learning the entities and predicate relationships latent representations in *TRAIN-TN* using PIDE model, we selected the best parameters: $\{\lambda_e$(learning rate of entities)=λ_r(learning rate of predicate relationships) = 0.1(when epoches $\leq$300), $\lambda_e = \lambda_r$ = 0.01(when epoches$>$ 300), κ=50, γ = 1$\}$ using valid data set *TRAIN-VD* on *WN-WR10K TRAIN EX.*; $\{\lambda_e$=0.01,λ_r = 0.001 , κ=50, γ = 1$\}$ on *FB-WR14K TRAIN EX.*. And in this experiment we also try use the TransE model as comparison, we selected paramters: $\{\lambda_e$=0.1(when epoches $\leq$300),λ_e= 0.01(when epoches$>$ 300),λ_r = 0.01, κ=20, γ =2$\}$on *WN-WR10K TRAIN EX.*;$\{\lambda_e$=0.01,λ_r = 0.001 , κ=50, γ = 1$\}$ on *FB-WR14K TRAIN EX.*. The number of iteration is 1000. We test it on test data set *TRAIN-TT* of the two data sets and evaluation results are shown in Table 2. Finally, we choose the entities representations encoded by PIDE model via comparison with TransE model. Other smarter methods to model knowledge base could be used but this is not our focus.

[6] The mean of those predicted ranks for LEFT-NEW;The others are similar.

Table 2. Knowledge base completion results of the different models. (The lower the better for Mean Rank, whereas Hits@10(%) is on the contrary.) {LR: Left Rank; LH10: Left Hits@10(%); RR: Right Rank; RH10: Right Hits@10(%); MR: Mean Rank; MH10: Mean Hits@10(%) }

DATASET	WR10K TRAIN EX.						FB-WR14K TRAIN EX.					
METRICS	LR	LH10	RR	RH10	MR	MH10	LR	LH10	RR	RH10	MR	MH10
TransE	109	73.9	109	73.9	109	73.9	173	40.5	77	52.1	125	46.3
PIDE	71	84.9	62	82.6	66.9	83.7	164	44.1	69	51.8	116	47.9

Table 3. Results of extending knowledge base in *Wordnet* with different models. (The lower the better for LR, RR and MR, whereas LH10, RH10 and MH10 are on the contrary.) {LR: Left Rank; LH10: Left Hits@10(%); RR: Right Rank; RH10: Right Hits@10(%); MR: Mean Rank; MH10: Mean Hits@10(%) }

DATASET	WN-WR10K																	
	LEFT-NEW						RIGHT-NEW						RIGHT-NEW					
METRICS	LR	LH	RR	RH	MR	MH	LR	LH	RR	RH	MR	MH	LR	LH	RR	RH	MR	MH
Un-transform	4443	0.78	4865	0.0	4654	0.39	4869	0.0	4421	0.87	4654	0.43	996	34.9	999	34.9	997	34.9
ANN	2221	10.12	2331	12.23	2132	23.12	2121	9.29	1212	8.63	998	15.76	898	36.36	989	35.63	885	36.10
DNN	342	35.12	434	33.21	341	34.67	367	40.23	423	38.43	322	38.58	423	40.09	421	39.10	311	41.01

Table 4. Results of extending knowledge base in *Freebase* with different models. (The lower the better for LR, RR and MR, whereas LH10, RH10 and MH10 are on the contrary.) {LR: Left Rank; LH10: Left Hits@10(%); RR: Right Rank; RH10: Right Hits@10(%); MR: Mean Rank; MH10: Mean Hits@10(%) }

DATASET	FB-WR14K																	
	LEFT-NEW						RIGHT-NEW						RIGHT-NEW					
METRICS	LR	LH	RR	RH	MR	MH	LR	LH	RR	RH	MR	MH	LR	LH	RR	RH	MR	MH
Un-transform	5367	0.0	1774	9.5	3071	9.7	883	12.4	5484	0.0	3184	6.2	1988	7.5	2032	7.6	2010	7.6
ANN	2361	17.48	265	12.37	2519	13.84	2048	14.58	2858	12.48	2058	15.58	1037	12.49	1859	10.54	1594	9.29
DNN	390	30.50	328	32.26	355	34.38	329	36.10	402	28.39	302	35.58	478	25.85	403	31.09	392	29.57

4.5 Extending Knowledge Bases

Using the two data set: WN-WR10K and FB-WR14K, we test different models with extending the knowledge bases. Firstly, we directly use the word embedding in knowledge base for extending knowledge base (denoted Un-transform). For FB-WR14k, we first use PCA[7] to reduce the 1K dimension to 50 dimension for consistent with the dimension of PIDE model. As in ANN model, for WordNet configuration: 4 layers (50 500 200 100), the learning rate:0.0001. We choose the sigmod function as the active function. We train it with the epochs is 500, and we cascade the semantic and syntactic representation that pre-trained by PIDE model as the output of training data (dimension = 100). On FB-WR14K, we also choose 4 lyaers (1000, 500, 200,100), the other parameter is the same. For DNN model, we choose the deep autoencode model, the setting of layer is :(50 500 1000 200 100) on WN-WR10K and (1000 2000 1000 500 100) on FB-WR14K. We use the typical learning algorithm contrastive divergence for optimization.

We can observe from Table 3 and 4 that the Un-transform method performs worse. I think it exactly indicate that the two latent semantic space: word embedding latent space and entity latent space are not the same. Even the same word or entity, the semantic expression of them are different in the two latent space. We can not directly exchange them from one to another through no additional model. And then we can find that the results is better after we use the ANN as the pipeline for connecting the two latent space. We also could observe from the tables that the DNN performs best. ANN is the typical neural network, and it is hard and may not be effective to optimization. DNN is a deep learning model, it could be trained by contrastive divergence effectively.

[7] A typical method used for dimension reduction.

5 Conclusion and Future work

Extending knowledge base is a problem of great importance. In this paper, we propose a framework to extend the knowledge base with multimodal deep neural network, word embedding and entity latent representation. Experiments demonstrate its good performance. We will explore how to integrate the models and improve the performance in further.

Acknowledgments

This work was supported by the Natural Science Foundation of China under Grant No. 61300080, No. 61273217, the 111 Project under Grant No. B08004 and FP7 Mobile Cloud Project under Grant No. 612212. The authors are partially supported by the Key project of China Ministry of Education under Grant No. MCM20130310, Huawei's Innovation Research Program, Postgraduate Innovation Fund of SICE, BUPT, 2015 and BUPT Excellent Ph.D. Students Foundation.

References

1. **Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008).** Freebase: a collaboratively created graph database for structuring human knowledge. *Proceedings of the 2008th ACM SIGMOD*, ACM.
2. **Bordes, A., Glorot, X., Weston, J., & Bengio, Y. (2013).** A semantic matching energy function for learning with multi-relational data. *Machine Learning*, Machine Learning.
3. **Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013).** Translating embeddings for modeling multi-relational. *Proceedings of Neural Information Processing Systems*, NIPS.
4. **Bordes, A., Weston, J., Collobert, R., & Bengio, Y. (2011).** Learning structured embeddings of knowledge bases. *Proceedings of the 25th Annual Conference on Artificial Intelligence(AAAI)*, AAAI.
5. **Collobert, R. & Weston, J. (2008).** A unified architecture for natural language processing: Deep neural networks with multitask learning. *Proceedings of the 25th international conference on Machine learning*, ACM, pp. 160–167.
6. **Collobert, R., Weston, J., & L., B. (2011).** Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, volume 12, JMLR, pp. 2493–2537.
7. **Hinton, G. E. (1986).** Learning distributed representations of concepts. *Proceedings of the eighth annual conference of the cognitive science society.*
8. **Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013).** Efficient estimation of word representations in vector space. *Proceedings of Workshop at ICLR, 2013*, ICLR.
9. **Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013).** Distributed representations of words and phrases and their compositionality. *Proceedings of NIPS*, NIPS.
10. **Mikolov, T., Yih, W.-t., & Zweig, G. (2013).** Linguistic regularities in continuous space word representations. *Proceedings of NAACL HLT*, NAACL.
11. **Miller, G. (1995).** Wordnet: A lexical database for english. *Communications of the ACM*, ACM.
12. **Socher, R., Chen, D., Manning, C. D., & Ng, A. Y. (2013).** Learning new facts from knowledge bases with neural tensor networks and semantic word vectors. *In Advances in Neural Information Processing Systems*, NIPS.
13. **Srivastava, N. & Salakhutdinov, R. (2012).** Learning representations for multimodal data with deep belief nets. *ICML Representation Learning Workshop*, ICML.
14. **Srivastava, N. & Salakhutdinov, R. (2014).** Multimodal learning with deep boltzmann machines. *Journal of Machine Learning Research*, JMLR.
15. **Xu, W. & Rudnicky, A. I. (2000).** Can artificial neural networks learn language models?
16. **Zhao Y, G. P., Gao S (2015).** Knowledge base completion by learning pairwise-interaction differentiated embeddings. *Data Mining and Knowledge Discovery*, volume 29, DMKD, pp. 1486–1504.

Yu Zhao received the B.S. degree from the Southwest Jiaotong University (SWJTU), Sichuan, in 2006 and the M.S. degree from Beijing University

doi: 10.13053/CyS-20-3-2472

of Posts and Telecommunications (BUPT), Beijing, in 2011. He is currently pursuing the Ph.D. degree with the School of Information and Communication, BUPT. He has visited in LIP6, Universit Pierre et Marie Curie (UPMC), Paris, France, from May to August 2015. He is visiting in Department of Computer Science, University of Rochester (UR), Rochester, USA from Sep. 2015 to Mar. 2017. His research interests include natural language processing, machine learning, recommendation system, etc.

Sheng Gao has been an Assistant Professor with the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, since 2012. He received the bachelor's and master's degrees from BUPT in 2003 and 2006, respectively, and the Ph.D. degree from Universite Pierre et Marie CURIE (Paris 6), Paris, France, in 2011. His current research interests include machine learning, data mining, information recommendation, and social network analysis. He has published over 20 academic papers on world-wide famous journals or conferences, including ISI, WWW, CIKM, and ECML.

Patrick Gallinari has been a Professor with Pierre et Marie Curie, Paris 6 University, Paris, France, since 1992, and the Director of the Computer Science Laboratory, LIP6, since 2005. He received the Ph.D. degree in computer science from the University of Compiegne, Compiegne, France, in 1985. His current research interests include machine learning applications and information retrieval.

Jun Guo is a Professor, Ph.D. Supervisor, Vice-President with the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, and the Dean of School of Information and Communication Engineering, BUPT, the Director of Pattern Recognition and Intelligent System Laboratory. He received the bachelor's and master's degrees from BUPT in 1982 and 1985, respectively, and the Ph.D. degree from Tohoku Gakuin University, Tohoku, Japan, in 1993. His current research interests include cross media information retrieval, web public sentiment information analysis, and network management and control. He is the person responsible for many projects funded by national 863 high-tech and national natural science foundation of China. He has published more than 100 papers on international journal and conference, including SCIENCE, Nature Online Magazine of Scientic Reports, and IEEE TPAMI.

Article received on 08/01/2016; accepted on 07/03/2016.
Corresponding author is Yu Zhao.