Post Processing of Word2vec for Category Classification based on Semantic

SungEn Kim
Computer Engineering
Hanbat National University
Republic of Korea
vinke@hanmail.net

SinYeong Ahn
High Performance Computing Group
Electronics and Telecommunications Research Institute
Republic of Korea
syahn@etri.re.kr

GuiHyun Baek
Computer Engineering
Hanbat National University
Republic of Korea
beik2000@naver.com

SuKyoung Kim
Computer Engineering
Hanbat National University
Republic of Korea
kimsk@hanbat.ac.kr

***Abstract*— Researchers need to research equipment for their research, so they support money from national research development business. Nation made a search platform for using research equipment efficient. However, it is difficult for researchers to search their intent and their need. Although the equipment existed, researcher bought again because they cannot find them. For this reason, this paper analyzes the similarity of equipment using Word2Vec Model for recommending equipment to researchers. And For experimenting superiority of proposed method, we compare another method that is Latent Semantic Analysis.**

Keywords—word2vec; document similarity; word similarity; deeplearning; research equipment;

I. INTRODUCTION

National Research Development Business has a goal that is supported researcher for developing national R&D. Like fig1 [1] , investment costs occupy more over 3% about research equipment and research facilities in this business. In other words, researchers steadily buy research equipment for R&D. Nation needs a plan about advancement and expansion of research equipment and facilities based on cooperating interagency for pushing efficient and balanced R&D business. One of the method for this, nation suggested researchers invigorating common use of research equipment and facilities. [2] Like this, it is important to use efficient and avoid buying duplicate or similar research equipment and facilities due to high price.

Korea constructed the research equipment search platform that is called 'ZEUS', so people can get information about equipment and small businesses can rent the high-price research equipment. Like Fig2. , however, it is difficult to search similar equipment that is needed research. Therefore, there is a waste of national finance that is caused by buying duplicated or similar equipment.

Fig. 1 . The Trend of Investing Research Equipment in National Research Development Business [1]

Fig. 2 . Duplicated Equipment in 'ZEUS' Site

Preventing this, search platform can recommend similar equipment that is suitable intention of user, however it is difficult to recommend similar equipment since involving many characteristics and natural language processing (e.g. use English mixed with Hangul or Chinese, number, unit, sign, synonym and etc.) described equipment title or details in equipment data. Resolving this problem, this research analyzes the similarity of equipment using Word2Vec Model that performed to express the word in vector space based on a surrounding context that is regardless type, construction of language, after that calculate similarity between words. And

increasing the accuracy of similarity calculation, it recommend similar equipment based on metadata of equipment. For experimenting superiority of proposed method, also, we apply Latent Semantic Analysis (LSA) method that is used in analyzing latent semantic and association of word recently. And this method has more accuracy about 14% rather than LSA in the experiment's result.

II. Related Research

A. Word2Vec and model

Traditionally, people expressed the word in computer, as 'One-Hot' expression [3] that expressed the word to 0 or 1. It is, however, difficult to calculate similarity between words. For this reason, people proposed 'Word-Embedding' expression [4], as developing machine learning, computer could have been learning large scale data. Word2Vec is proposed by Tomas Mikolov who is Google Researcher in 2013. [5] It reduced a lot of calculation in training method based on Neural Network. Word2Vec is based on two models that is Continuous Bag-of-Words (CBOW) and Skip-gram. [6] CBOW model predicts target word based on surrounding words like fig3. , and Skip-gram model predict generating of surrounding words based on center word like fig3. Traditionally, people used CBOW model for vetorlized the word because of having small dataset, but as increasing dataset, people almost used Skip-gram model. This paper use Skip-gram model for vectorlizing the word.

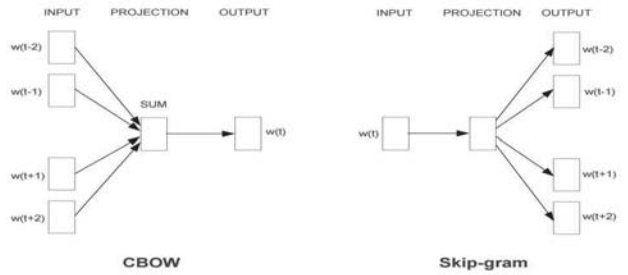

Fig. 3 . Two model of Word2Vec

B. Lantent Semantic Analysis (LSA)

LSA is utilized Latent Semantic Indexing (LSI) method, originally made for information search, in psychology research of word meaning by Landauer and Dumais (1997). [7] It can search close meaning between word and word, word and document, document and document not like keyword matching. LSA vetorlized the word through TF-IDF like (1). [8] And it calculates the weight of word vectors and reduce dimensions through Singular Value Composition (SVD) like (2). [9] Finally, it calculates similarities between words or documents combining sets that calculated through SVD. We calculate the similarities between documents through LSA using (3). [10]

$$\mathrm{TF-IDF} = W_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right) \quad (1)$$

$$\mathrm{SVD} = \mathrm{A} = \mathrm{UD}V^{\mathrm{T}} \quad (2)$$

$$(\mathrm{U} \in \mathbb{R}^{m\times n}, D \in \mathbb{R}^{n\times n}, V \in \mathbb{R}^{n\times r})$$

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} \quad (3)$$

In (1), $tf_{i,j}$ means number of occurrences of i in j, df_i means number of documents containing i. N means total number of documents. In (2), U and V mean orthogonal matrix, and D mean diagonal matrix. In (3), A means $U_{i,;}$ matrix, B means $D_{;,i}$ matrix, and n means dimension of matrix.

III. DataSet

We used 101,931 equipment that was registered 'ZEUS' portal site. For deciding proposed method's validation, we analyzed to choose top 10 of the branch name of equipment. And comparing two case, one case was that we extracted 100 equipment each chosen branch. Second case was that we extracted all data of top 10 branch like Table1. Each equipment had metadata like fig 3. In this metadata, we calculated to apply equipment's model name, Korean name, English name, manufacture and details.

TABLE I. Extract Top 10 of the Branch Name of Equipment

No	Branch Name	Count of Equipment	Percentage of Branch (%)
1	Liquid Chromatograph Mass Spectrometer	1,583	1.55
2	Super Computer	1,434	1.41
3	Optical Microscope	1,300	1.28
4	Server	1,044	1.02
5	Electric/sintering furnace	1,034	1.01
6	Thermal Analysis	946	0.93
7	Laser Generator Unit	831	0.82
8	Response Equipment	719	0.71
9	Input/Output Unit	706	0.69
10	Voltage/Current/Measurement of power Testing Equipment	684	0.67
Sum		10,281	10.09

equip_no	model_name	korean_name	english_name	manufacture	branch_name	equipment_detail
NFEC-2013-06-179831	Samba TM 1000mW	녹색광 레이저	Samba TM 1000mW	Cobolt Ab	레이저발생장비	- Samba TM 1000mW (freque
NFEC-2015-03-200644	개별장비	다목적 수직가열로 제작	Multipurpose Vertical Furna	중우엔지니어링	전기/소결로	- 수직 구획부재에 대하여 국내
NFEC-2011-02-143530	모델명 없음	분위기 제어로	muffle kiln	탑브	전기/소결로	- 용도 : 전구재와 리튬의 혼합
NFEC-2013-09-182517	TZSO-20	엔디-야그 유브이 레이	NdYAG laser	Litron Lasers	레이저발생장비	- 펄스타입의 비행시간형 질량
NFEC-2014-02-185783	11300	바이어스 전류 테스트 시	Bias Current Test System	Chroma ATE	전압/전류/전력측정시험장비	- 0~100A DC 바이어스 인가 7
NFEC-2015-01-195590	TITAN	100 TW 레이저 여기용	NdYAG pump laser for 100	Amplitude Technologies	레이저발생장비	- 100 TW 레이저 증폭기를 펌
NFEC-2011-08-147223	Nexus 7000	서울분원 백본스위치	Backbone Switch	Cisco	입출력장치	- 10기로이상의 모듈러 타입 C
NFEC-2015-04-201323	SCIENV-AN-25-1500	고온열처리장비	High Temperature Heating	케이엔텍	전기/소결로	- 1100℃/1500℃에서 시료의 일
NFEC-2013-08-181970	모델명 없음	개질기테스터	Reformer tester	알티아이엔지니어링	반응장비	- 1kW / 5kW급 천연가스 개질
NFEC-2011-01-133244	모델명 없음	수냉 도가니 및 주변 장	Cold Crucible and Melt Del	광지하이텍	전기/소결로	- 200 kg 이상의 UO2와 ZrO2

Fig. 4 . Metadata about equipment

IV. Proposed Method

Proposing method in this paper, it calculate similarity between documents based on similarity between words. Method divide three parts such as preprocessing, composing Word2vec learning model, postprocessing.

A. Preprocessing

Before Word2vec model learning, we consider equipment's model name, Korean name, English name, manufacture, details as one document. After that, we remove postposition, adverb, special letters, and plural form using customizing Mecab [11] morpheme analyzer. And then, all English words convert small letter. Finally, we divide the word by white space in documents.

B. Word2Vec Learning Model Construction

We used API that was provided by 'DeepLearning4j' [12] , for constructing Word2vec learning model, and we used Skip-gram model. When we constructed Word2vec learning model, we used TABLE 2. Setting.

TABLE II. Setting for Constructing Word2vec Learning Model

No	Parameter Name	Value
1	Window Size	4
2	Layer Size	300
3	Batch Size	100
4	Iterations	500
5	Epoch	5
6	minWordFrequency	0
7	minLearningRate	0.01
8	Negative Sampling	2

When model learns data, window size decides how many words that is in front of target word or behind target word, reflecting in the context learning. Layer Size expresses the number of features in the word vector. Batch Size decides how many words the computer reads. Iterations decides how many time does the model learn data in batch size. MinLearningRate decides as least of the word count. Finally, Negative Sampling [13] decides to extract part of softmax calculation, generalize rest of data instead of calculation all data.

C. PostProcessing

Vectoralized words can be calculated through fig4.

Algorithm 1. Document Similarity using Word2Vec

Input : Vectorlized Words of Document $\{W_1(V_1), W_i(V_i)\}$

Output : a Set of Similarity between Documents $\{S_1(D_1, D_2), S_n(D_k, D_j\}$

Foreach(n < Document Size)

  Foreach(k < Document Size)

    Foreach(i < Word Size in Document n)

      Foreach (j < Word Size in Document k)

        $D^n_{W_i}$ Cosine Similarity $D^k_{W_j}$

    Exclude n = k

  O = Extract Word similarity high rank 10% between W_i and W_j

  (When calculating high rank 10%, exclude similarity between W_i and W_j

  L = Extract Word Similarity high rank 40% between $\{S_1(W_1, W_2), S_L(W_i, W_j)\}$

  P = Mean L

  Return P

Fig. 5 . Postprocessing of Document similarity using Word2Vec Model

W means word. V means word vector. D means documents. O is a set of high rank 10% that is calculated similarity between word of criteria document and word of target document. L is a set of high rank 40% that is extracted word similarities of criteria document. P is a set of means that is extracted similarities between criteria document and target document.

V. Result

Table 3 is an environment of experiment for testing proposed method. Fig 5 is a result that is accuracy of similarity between equipment that is extracted 100 equipment each chosen branch. We measured the accuracy of similarity based on Meta classification criteria that is made by experts.

TABLE III. Environment of Experiment

No	Type	Measurement Environment
1	CPU	Intel® Xeon E5-2620 v4
2	RAM	256GB
3	Java Version	JDK 1.8
4	IDE	Eclipse Mars.2

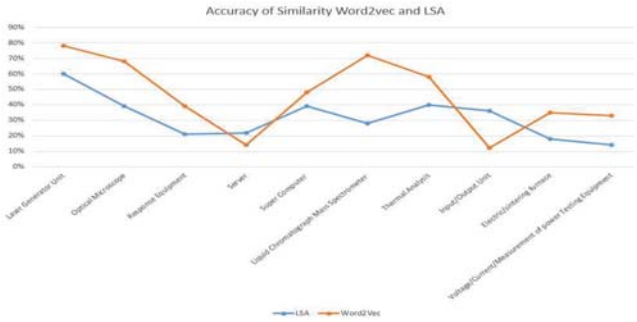

Fig. 6 . Accuracy of Similarity Word2vec and LSA based on 1,000 Equipment

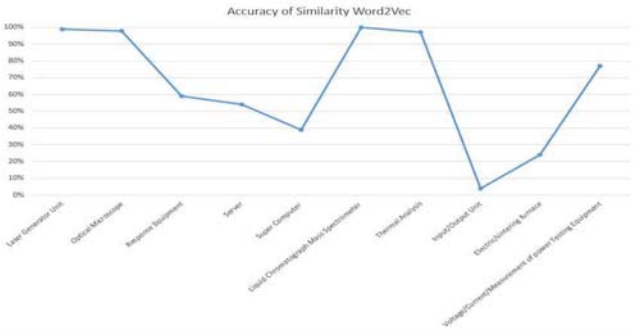

Fig. 7 . Accuracy of Similarity Word2Vec based on 10,281 Equipment

TABLE IV. Analysis Each Algorithm using Resource

No	Data Scale	Type	Use time (Unit : Minute)	Use Memory (Unit: MB)	Note
1	Small Data (518KB)	Word2vec	35.88	1,016.53	
		LSA	5.86	396.03	
		Gap	30.2	620.5	
2	Large Data (6,937KB)	Word2Vec	660.5	49,999.5	cannot implement LSA
		LSA	∞	∞	
		Gap	-	-	

Fig6 is accuracy of similarity word2vec and LSA. Although Word2vec used system resources more than LSA, but Word2vec's accuracy of similarities are better than LSA about 14%. Two data's accuracies are better than Word2vec, but the gaps are a little. Fig 7 is increase the dataset using experiment. As a result, most of Word2Vec's accuracy of similarities are increase about 19%, but LSA cannot implement with larger data.

VI. Conclusion

This paper analyze the similarity of equipment using Word2Vec Model. Measuring the accuracy, we compare another algorithm which is used to analyzing latent semantic and association of words recently, with proposed method. Therefore, proposed method uses more computer resource, but it has more accuracy about 14% rather than LSA based on 1,000 equipment. And we found the more data learn in Word2vec model, the better accuracy of similarity we will get.

This method can save national finance constructing the service that prevents to buy duplicated equipment. In the future, also, if we were decreasing time complexity of proposed method, and we could utilize the service that can recommend where the data is classified when new data were entered in search platform.

Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.2016-0-00087, Development of HPC System for Accelerating Large-scale Deep Learning)

References

[1] Ministry of Science, ICT and Future Planning, "Report for National Research Facitilies Equipment Investigation. Analysis in 2014", 2014.

[2] Moon Ji Choi, "Advanced Plan for Operating and Utilzing National Research Facilities Equipment", 2013.4.

[3] "One-Hot Expression's Definition", < https://en.wikipedia.org/wiki/One-hot >.

[4] "Word Embedding's Definition", < https://en.wikipedia.org/wiki/Word_embedding>.

[5] "Word2Vec", < https://code.google.com/archive/p/word2vec/>.

[6] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean "Efficient Estimation of Word Representations in Vector Space," arXiv preprint arXiv:1301.3781, 2013.

[7] Beo Mmo Kang, "Text Context and Word Meaning: Latent Semantic Analysis", Linguistics 68, 2014.4.

[8] "TF-IDF's Definition", < https://en.wikipedia.org/wiki/Tf%E2%80%93idf>.

[9] "Singular Value Decomposition", < https://en.wikipedia.org/wiki/Singular_value_decomposition>.

[10] "Cosine Similarity", <https://en.wikipedia.org/wiki/Cosine_similarity>.

[11] "Mecab", Mecab : Yet Another Part-of-Speech and Morphological Analyzer.

[12] "Word2Vec Java API", <https://deeplearing4j.org>.

[13] Tomas Mikolov, "Distributed Representations of Words and Phrases and their Compositionality", Advances in neural information processing system, 3111-3119, 2013.