http://dx.doi.org/10.5392/JKCA.2016.16.10.687

Word2vec을 활용한 문서의 의미 확장 검색방법

Semantic Extention Search for Documents Using the Word2vec

김우주*, 김동희**, 장희원*
연세대학교 정보산업공학과*, 한국철도기술연구원**

Woo-ju Kim(wkim@yonsei.ac.kr)*, Dong-he Kim(kdh777@krri.re.kr)**,
Hee-won Jang(hwjang1000@gmail.com)*

요약

기존의 문서 검색 방법론은 TF-IDF와 같은 벡터공간모델을 활용한 키워드 기반 방법론을 사용한다. 키워드 기반의 문서검색방법론으로는 문제가 몇몇 문제점이 나타날 수 있다. 먼저 몇 개의 키워드로 전체의 의미를 나타내기 힘들 수 있다. 또 기존의 키워드 기반의 방법론을 사용하면 의미상으로 비슷하지만 모양이 다른 동의어를 사용한 문서의 경우 두 문서 간에 일치하는 단어들의 특성치만 고려하여 관련이 있는 문서를 제대로 검색하지 못하거나 그 유사도를 낮게 평가할 수 있다. 본 연구는 문서를 기반으로 한 검색방법을 제안한다. Centrality를 사용해 쿼리 문서의 특성 벡터를 구하고 Word2vec알고리즘을 사용하여 단어의 모양이 아닌 단어의 의미를 고려할 수 있는 특성 벡터를 만들어 검색 성능의 향상과 더불어 유사한 단어를 사용한 문서를 찾을 수 있다.

■ 중심어 : | 시맨틱 검색 | 문서특징벡터 | 벡터공간모델 | Word2vec |

Abstract

Conventional way to search documents is keyword-based queries using vector space model, like tf-idf. Searching process of documents which is based on keywords can make some problems. it cannot recogize the difference of lexically different but semantically same words. This paper studies a scheme of document search based on document queries. In particular, it uses centrality vectors, instead of tf-idf vectors, to represent query documents, combined with the Word2vec method to capture the semantic similarity in contained words. This scheme improves the performance of document search and provides a way to find documents not only lexically, but semantically close to a query document.

■ keyword : | Semantic Search | Document Feature Vector | Vector Space Model | Word2vec |

Ⅰ. 서론

정보검색의 대표적인 방법인 벡터공간모델에 기반을 둔 검색은 질의에 나타난 키워드들(입력 문서의 단어들)을 인덱스로 하여 해당 단어가 이 문서에서 어느 정도의 가중치를 가지고 있는가를 기준으로 우선순위를 부여한다. 대표적인 벡터공간모델인 TF-IDF는 문서에 등장한 단어들의 중요도를 나타내는 특성 값을 사용하여 문서의 특성벡터를 형성한다. 이러한 단어의 출현을 기반으로 문서들의 특성벡터를 추출하고 특성벡터들의

* 본 연구는 한국철도기술연구원 주요사업비 지원으로 수행되었습니다.
접수일자 : 2016년 09월 27일
수정일자 : 2016년 10월 10일
심사완료일 : 2016년 10월 10일
교신저자 : 장희원, e-mail : hwjang1000@yonsei.ac.kr

유사도를 사용해 검색하는 기존의 방법이 가진 문제점은 비교대상이 되는 두 문서에 모두 포함된 단어만이 유사도에 영향을 미친다는 점이다. [그림 1]은 그에 대한 예시이다. [그림 1]의 두 문서의 경우 일견하기에 유사도가 매우 높을 것 같지만 Cosine Similarity를 사용해 유사도를 계산할 경우 유사도는 약 0.07이다. 이러한 벡터공간모델은 출현 단어에 대해선 검색결과를 보이지만 문맥과 의미를 고려하지 않기 때문에 검색성능에 한계가 있음을 알 수 있다.

그림 1. 기존 키워드 기반 문서검색의 문제점

본 연구는 벡터공간모델의 이러한 단점을 보완하기 위해 단어 간 유사도를 사용하여 출현하지 않은 단어의 특성 값을 간접평가하고 같은 문서내의 유사한 단어의 출현에도 가중치를 주는 방법을 제안하고자 한다. 단어 간 유사도를 구하기 위해서 Word2vec 알고리즘을 사용하여 단어들이 가지는 의미 또한 고려하고자 하였다.

II. 관련연구

1. Pagerank

페이지랭크[1] 알고리즘은 네트워크 그래프에서 노드들의 중요도를 측정해주는 알고리즘이다. 본 연구에서는 쿼리 문서가 포함하고 있는 단어들의 중요도를 판단하기 위하여 페이지랭크를 활용하였다. 쿼리 문서에 포함된 단어들을 하나의 노드로 생각하고 단어 간의 관계를 확률적으로 계산해주는 PMI(point-wise mutual index) 지수를 노드사이의 관계, 즉 엣지로 설정하였다.

2. Word2vec

최근 neural network를 활용하여 Stochastic language model을 학습시킨다. [4-6]은 neural network를 사용하여 language model을 구축할 수 있고 [10]은 Deep neural network를 사용하여 다양한 language model이 구축가능하다고 하였다. 그 과정은 [그림 2]처럼 t번째 등장한 단어 W를 통해 전후의 등장한 단어들로 단어들의 뭉치를 만들고 t-2번째, t-1번째, t+1번째, t+2번째 등장하는 단어들을 추측할 수 있는 가중치를 학습시킴으로써 단어의 의미를 표현하고자 하였다. [7]에서는 기존의 [4-6]의 연구에 binary-tree개념을 사용하여 발전된 language model을 제시하기도 하였다. Word2vec 알고리즘역시 이러한 방법론의 하나로써 단어를 벡터로 표현할 수 있게 해준다[2][3]. 단층의 신경망을 이용한 neural language model로 각각의 단어를 표현하는 벡터를 만들게 된다. 각각의 단어를 벡터로 표현할 수 있다면 단어들의 유사도 또한 코사인 유사도를 통해 쉽게 구할 수 있다. 단어의 벡터를 학습시킬 때 [그림 2]와 같이 특정 문맥에서 유추할 수 있는 단어의 확률을 최대화하는 방법으로 학습을 시키기 때문에 [그림 3]처럼 유사한 단어들은 비슷한 벡터의 위치를 가지게 되고 유사도는 높아지게 된다. 단어벡터의 위치는 각 단어의 의미를 나타내고 있기 때문에 각 단어의 상관관계는 벡터의 거리로 표현할 수 있다. 연구에서는 단어 간의 유사도를 Word2vec을 통해 추출한 단어벡터들로 코사인 유사도를 구해 거리가 얼마나 가까운지로 평가한다.

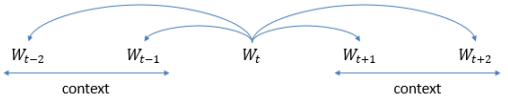

그림 2. 단어를 통해 문맥을 학습시키는 Word2vec

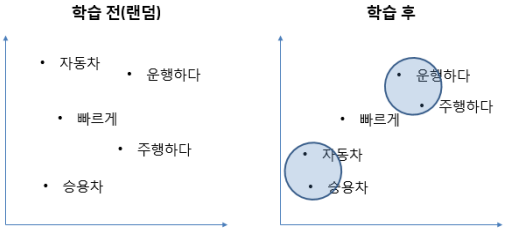

그림 3. 학습 전, 후 단어벡터의 위치변화

Ⅲ. Word2vec을 활용한 문서검색방법론

방법론의 전체적인 도식은 [그림 4]와 같다. 본 연구의 목적인 문서 간 검색 시스템은 먼저 검색 대상이 되는 문서들을 TF-IDF를 활용해 문서에 포함된 단어들과 그 중요도를 나타내는 벡터공간모델[8][9]의 형태로 저장을 하고 쿼리 문서를 입력받아 변환시키고 이 문서들과의 유사도 비교를 통해 최종적으로 검토해야할 문서들을 추천해 주게 된다. 쿼리 문서의 변환 과정은 페이지랭크 알고리즘을 통해 TF-IDF 벡터와 같은 형태로 문서의 단어별 중요도를 추출하게 되고 단어 간 유사도를 활용해 문서의 의미를 확장시킨다. 세부적인 변환과정은 아래와 같다.

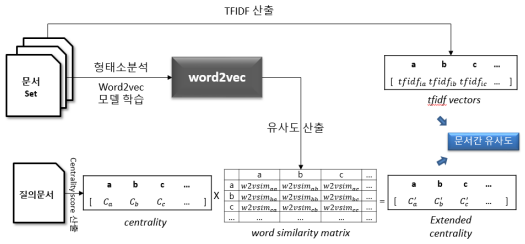

그림 4. 문서 검색방법론의 전체 도식

1. 쿼리 문서의 그래프 추출

관련연구에서 설명했듯이 쿼리 문서의 단어별 유사도를 페이지랭크 알고리즘을 통해 구하기 위해서는 문서를 그래프로 변환하는 과정이 필요하다. 본 연구에서는 [그림 5]와 같이 문서에 포함된 단어를 각각 노드로 보았고 단어 간의 관계를 확률적으로 계산해주는 PMI 지수를 활용해 노드 간의 관계를 표현하였다.

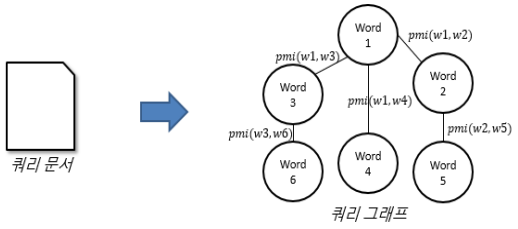

그림 5. 쿼리 문서를 쿼리 그래프로 변환

$$pmi(u,v) = \log\left(\frac{p(u,v)}{p(u)p(v)}\right) \quad (1)$$

PMI를 계산할 때 사용하는 단어의 출현확률을 구하기 위해서 네이버 검색결과를 활용했다. 각각의 단어의 출현 문서 개수와 단어의 동시출현 문서 개수로 PMI를 계산하였다.

2. 페이지랭크를 사용한 쿼리 문서의 단어별중요도 추출

위에서 구한 쿼리 그래프와 노드별 중요성을 추출할 수 있는 페이지 랭크 알고리즘을 활용하면 [그림 6]처럼 검색대상이 되는 문서들의 TF-IDF와 같은 쿼리문서의 단어별 중요도를 추출할 수 있다. 우리는 이렇게 추출한 쿼리 문서의 단어별 중요도를 Centrality라고 명명하였다.

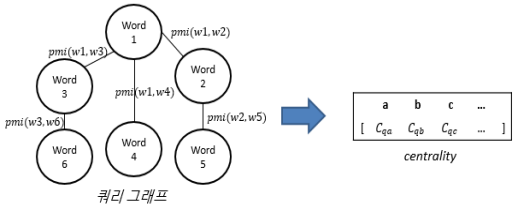

그림 6. 쿼리 그래프를 Centrality로 변환

$$\boldsymbol{C}(t+1) = (1-d)\frac{1}{N}\boldsymbol{1} + d\boldsymbol{C}(t) \quad (2)$$

N은 쿼리 그래프 안의 노드의 숫자를 C는 각 노드의 중요도를 의미한다.

3. Word2vec 모델링

Word2vec을 활용해서 단어 간 유사도를 추출하는 과정은 지금까지와 달리 전체 문서에 대해서 진행한다. 이는 대상 단어가 많을수록 정확한 문맥을 통해 단어벡터를 학습시킬 수 있기 때문이다. [그림 2]와 같이 단어를 넣고 그 단어를 통해 주변문맥에 맞는 단어를 정확히 추측할 수 있게 학습을 시키게 된다. 모델의 최적화 함수는 다음과 같다.

$$\max_\theta \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c}\log p(w_{i+j}|w_i;\theta) \quad (3)$$

c는 학습시키는 전체 단어의 개수, w는 등장하는 단어를 의미한다. 주어진 단어를 통해 문맥을 맞게 예측하도록 학습이 되기 때문에 같은 의미의 단어는 비슷한 문맥에서 여러 번 등장하게 되고 학습이 진행되면서 비슷한 의미의 단어들의 벡터들은 비슷한 위치로 나타나게 된다. 학습된 Word2vec모델을 사용하면 단어 간 유사도를 나타낼 수 있다. 본 연구에서 단어간 유사도를 활용하기 위해 [그림 7]과 같은 유사도 행렬을 사용한다.

	자동차	승용차	운행하다	주행하다	빠르게
자동차	1	0.7	0	0	0
승용차	0.7	1	0	0	0
운행하다	0	0	1	0.5	0
주행하다	0	0	0.5	1	0
빠르게	0	0	0	0	1

그림 7. 단어의 유사도행렬 예시

4. 단어의 유사도를 이용한 쿼리확장

쿼리 확장을 위해서 단어의 유사도를 활용한다. 앞에서 페이지랭크를 활용해 쿼리 문서에 속한 단어들의 중요도를 평가하였고 Word2vec을 활용해 단어의 유사도 행렬을 구할 수 있었다. 우리는 이를 활용하여 행렬 곱을 통해 쿼리 확장을 시도하였다. 사용한 수식은 다음과 같다.

$$\tilde{C}_i = C_i M \quad (4)$$

우변의 확장된 쿼리 벡터는 Extended Centrality로 명명하였고 C는 페이지랭크를 통해 구했던 Centrality, M은 단어의 유사도 행렬을 의미한다. 쿼리 확장의 결과는 [그림 8]의 예시와 같다. 방법론을 통해 쿼리를 확장하면 앞선 예제에서 (자동차, 승용차), (운행하다, 주행하다) 와 같이 같은 뜻을 가졌지만 평가할 수 없었던 요소들의 중요도도 함께 평가할 수 있다.

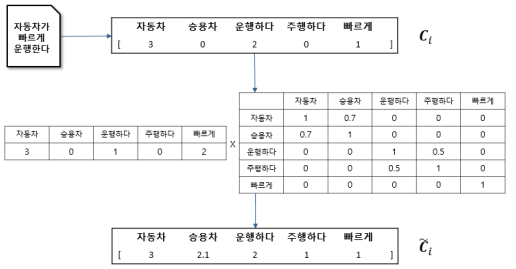

그림 8. 쿼리 확장의 예시

5. 문서 간 비교

변환이 완료된 쿼리 문서의 Extended Centrality와 검색대상이 되는 문서 셋의 TF-IDF 벡터들과의 유사도를 구하고 이 유사도를 기준으로 유사한 문서를 추천한다. 유사도비교에는 코사인 유사도를 활용하였다. [그림 9]와 같이 단어 간 유사도를 활용하여 비슷한 단어를 사용한 문서와의 유사도를 정당하게 평가할 수 있다.

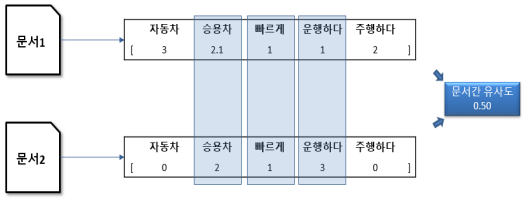

그림 9. 쿼리 확장과 유사도 비교 결과 예시

Ⅳ. 실험

실험은 철도기술 연구원의 바이모달(BTS) 연구요약서를 쿼리 문서로 활용하여 총 78개의 법령과의 유사도를 비교하였다. 78개의 문서 중 22개의 문서는 관련이 있고, 12개의 문서는 애매한 문서, 나머지 문서는 관련이 없는 문서로 연구계획서와의 유사도의 크기로 순위를 측정하였다. [그림 1]과 같은 전체 과정에서 78개의

표 1. 실험에 사용한 쿼리 벡터들

Occurrence	등장한 모든 단어의 중요도를 1로 평가
Centrality	페이지랭크로 단어의 중요도를 평가
Extended Centrality	Centrlity를 Word2vec으로 확장

문서를 문서 셋으로 설정하고 TF-IDF벡터 상태로 저장하고 쿼리문서를 벡터공간모델로 변환하여 평가지표는 랭크합, 가중랭크합, 11-point average precision을 활용하였다. 랭크합은 관련된 문서들의 순위 합을 나타내고 가중 랭크 합은 12개의 애매한 문서에 대한 가중치를 0.5로 설정한 랭크합과 같다. 랭크합과 가중 랭크합은 낮은 값을 나타낼수록 좋은 성능을 나타낸다. 11-point average precision은 정확률-재현률 곡선에서 재현률이 0, 0.1, …, 1 인 11개의 점에서의 정확률의 평균을 구하는 것이다. 11-point average precision은 높은 값을 가질수록 좋은 성능을 나타낸다. 평가지표들의 수식은 다음과 같다. M은 검색대상이 되는 문서들의 총 개수를 의미한다.

$$rank_i : predicted \sim ilarity\, rank\, of\, document\, i \quad (5)$$

$$w_i = \begin{cases} 1 & \text{if a} \\ & ted \\ 0.5 & \text{if } ambigous \\ 0 & \text{if } unrelated \end{cases}$$

$$ranksum = \sum_i^M rank_i$$

$$weightedranksum = \sum_i^M rank_i * w_i$$

$$P(r) = \max_{\tilde{r} \geq r} p(\tilde{r}) \quad (6)$$

$$11\, Average\, \Pr ecision = \frac{1}{11} \sum_{r=1}^{M} P(r)$$

바이모달(BTS) 연구서를 사용한 검색결과는 [그림 10][그림 11]과 같다. 사용한 세 지표 모두 Occurrence보다 Centrality가 Centrality보다 Extended Centrality가 더 좋은 성능을 가지고 있음을 나타낸다.

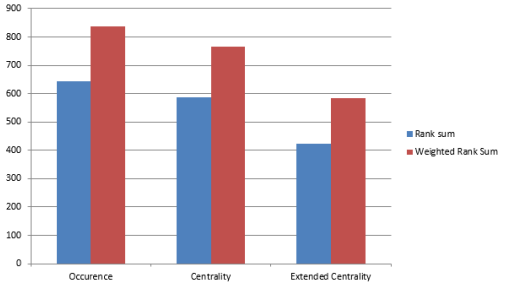

그림 10. 랭크합, 가중랭크합 결과

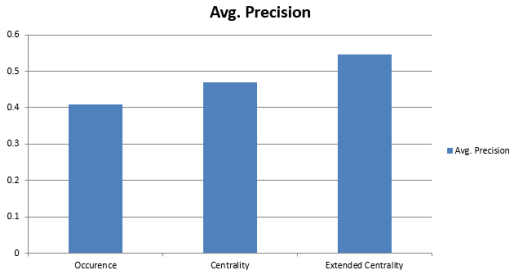

그림 11. 1-point average precision 결과

V. 결론

기존의 키워드 기반의 검색방법론과 달리 본 연구에서는 문서기반 검색방법론을 제시하였다. 연구의 단어의 유사도를 사용하여 문서의 특징벡터를 확장하는 과정은 제안한 내용에서 사용한 Centrality 뿐만 아니라 TF-IDF와 같은 기존의 벡터공간모델을 활용하는 방법론에도 적용가능하다. 단어의 유사도를 찾기 위해서 사용한 Word2vec은 문맥으로 단어의 의미를 학습하는 stochastic language models의 성질을 가지고 있기 때문에 단어의 유사도는 의미상으로, 문맥상으로 가까울수록 높게 측정되고, 이를 Centrality와 같은 쿼리 문서의 벡터공간모델에 적용하면 [그림 9]와 같이 단어의 의미를 고려해서 유사한 문서를 찾을 수 있었다.

본 연구의 적용 예상 분야는 행정문서의 검색이다. IV장의 실험과 같이 연구과제의 요약서를 입력하면 연구 요약서와 관련된 법령과 법령 문서를 찾아 연구 과제를 검토하는 과정에서 참고해야할 법령 문서들을 추천 받을 수 있다. 이 과정에서 비전문가 입장에서 모를 수 있는 전문용어가 등장해도 쿼리 문서를 확장시키는 과정에서 사용된 Word2vec의 성질에 의해 비슷한 단어로 대체해서 비슷한 문서를 검색할 수 있다.

본 연구의 한계점은 크게 두 가지가 있다. 첫째로 단어의 출현순서가 나타내는 의미를 고려하지 못한다. 두 괄식 글의 경우 처음에 나온 문장이 전체를 대표할 수 있다고 할 수 있지만 본 연구에서 제안한 방법론은 단어의 출현 순서에 따른 가중치를 고려하지 못한다. 또한 문서 대 문서 검색의 특성상 서로 비교하는 문서의 크기가 커지면 커질수록 일치하지 않는 단어가 늘어날

것이고 유사도를 실제보다 낮게 평가될 수 있다.

참 고 문 헌

[1] S. Brin and L. Page, “The Anatomy of a Large-scale Hypertextual Web Search Engine," Computer Networks and ISDN Systems, Vol.33, pp.107-117, 1998.

[2] T. Mikolov, K. Chen, G. Corrado, and J, Dean “Efficient estimation of word representations in vector space,” arXiv preprint arXiv:1301.3781, 2013.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and theier compositionality,” Advances in neural information processing systems, 2013.

[4] Yoshua Bengio, New distributed probabilistic language models. Dept. IRO, University de Montreal, Montreal, QC, Canada, Tech. Rep, 1215, 2002.

[5] Yoshua Bengio and Samy Bengio, "Modeling high-dimensional discrete data with multi-layer neural networks," In NIPS, Vol.99, pp.400-406, 1999.

[6] Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Janvin, "A neural probabilistic language model," The Journal of Machine Learning Research, Vol.3, pp.1137-1155, 2003.

[7] Yoshua Bengio and Jean-Sebastien Senecal, et al. Quick training of probabilistic neural nets by importance sampling, In AISTATS Conference, 2003.

[8] Gerard Salton, Anita Wong, and Chung-Shu Yang, "A vector space model for automatic indexing," Communication of the ACM, Vol.18, No.11, pp.613-620, 1975.

[9] David Dubin, The most inuential paper gerard salton never wrote, 2004.

[10] Ronan Collobert and Jason Weston, A unied architecture for natural language processing: Deep neural networks with multitask learning, In Proceedings of the 25th international conference on Machine learning, pp.160-167, ACM, 2008.

저 자 소 개

김 우 주(Woo-ju Kim) 정회원

- 1994년 : KAIST 경영과학(박사)
- 1996년 : 전북대학교 산업정보시스템공학과 교수
- 2004년 ~ 현재 : 연세대학교 정보산업공학과 교수

<관심분야> : 시멘틱 웹, 지식 관리 및 인공지능 웹서비스 등

김 동 희(Dong-he Kim) 정회원

- 1994년 : 인하대학교 산업공학(석사)
- 2000년 : 인하대학교 산업공학(박사)
- 2000년 ~ 현재 : 한국철도기술연구원 책임연구원

<관심분야> : 철도운영시스템, 운영최적화 등

장 희 원(Hee-won Jang) 정회원

- 2016년 : 연세대학교 정보산업공학과(석사)
- 2016년 ~ 현재 : 연세대학교 정보산업공학과 박사과정

<관심분야> : 머신 러닝