JANT 한국항행학회논문지
Journal of Advanced Navigation Technology

응용 및 융합 기술

J. Adv. Navig. Technol. 20(6): 617-624, Dec. 2016

한글 워드임베딩과 아프리오리를 이용한 검색 시스템의 질의어 확장

Query Extension of Retrieve System Using Hangul Word Embedding and Apriori

신 동 하 · 김 창 복*
가천대학교 에너지 IT학과

Dong-Ha Shin, Chang-Bok Kim*
Department of Energy IT, Gachon University, Gyeonggi-do 13120, Korea

[요 약]

한글 워드임베딩은 명사 추출과정을 거치지 않으면, 학습에 필요하지 않은 단어까지 학습하게 되어 효율적인 임베딩 결과를 도출할 수 없다. 본 연구는 한글 워드임베딩, 아프리오리, 텍스트 마이닝을 이용하여, 특정 도메인에서 질의어 확장에 의해 보다 효율적으로 답변을 검색할 수 있는 모델을 제안하였다. 워드임베딩과 아프리오리는 질의어에 대해서 의미와 맥락에 따라 연관 단어를 추출하여, 질의어를 확장하는 단계이다. 한글 텍스트 마이닝은 명사 추출, TF-IDF, 코사인 유사도를 이용하여, 유사답변 추출과 사용자에게 답변하는 단계이다. 제안모델은 특정 도메인의 답변을 학습하고, 연관성 높은 질의어를 확장함으로서 답변의 정확성을 높일 수 있다. 향후 연구과제로서, 데이터베이스에 저장된 사용자 질의를 분석하고, 보다 연관성 높은 질의어를 추출하는 연구가 필요하다.

[Abstract]

The hangul word embedding should be performed certainly process for noun extraction. Otherwise, it should be trained words that are not necessary, and it can not be derived efficient embedding results. In this paper, we propose model that can retrieve more efficiently by query language expansion using hangul word embedded, apriori, and text mining. The word embedding and apriori is a step expanding query language by extracting association words according to meaning and context for query language. The hangul text mining is a step of extracting similar answer and responding to the user using noun extraction, TF-IDF, and cosine similarity. The proposed model can improve accuracy of answer by learning the answer of specific domain and expanding high correlation query language. As future research, it needs to extract more correlation query language by analysis of user queries stored in database.

Key word : Word embedding, Word2vec, Apriori, Cosine similarity, TF-IDF.

https://doi.org/10.12673/jant.2016.20.6.617

Received 25 November 2016 **Revised** 29 November 2016
Accepted (Publication) 27 December 2016 (30 December 2016)

***Corresponding Author ; Chang-Bok Kim**

Tel : +82-10-8908-3946
E-mail : cbkim@gachon.ac.kr

Ⅰ. 서 론

검색시스템은 질의어를 입력하여 정보를 검색하는 시스템으로, 질의 유형은 내용검색, 사이트 검색, 서비스 검색 등이 있다. 사용자 질의어는 평균 2.21개 정도로서 함축적인 의미를 가지며, 검색을 위한 판단 기준이 되지만, 사용자가 질의어를 직접 선정해야 하며, 검색을 위한 적합한 질의어를 연상하는데 어려움이 있다[1]. 검색시스템은 이러한 문제점을 해결하기 위해, 질의어의 의미와 맥락에 따른 연관성을 이용하여, 질의어를 확장하는 방법과 클릭 로그데이터를 이용하는 방법들이 연구되어 왔다[2]. 질의어 확장은 질의어와 의미와 맥락이 유사한 단어를 추출해야 하며, 질의어와의 유사도를 측정할 수 있어야한다. 기존의 질의어 확장 방법은 시소러스(thesaurus), 클러스터링(clustering), 적합도 피드백(relevance feedback)등이 있다[3]. 그러나 이러한 방법들은 질의어와 어느 정도 연관관계가 있는지 다소 객관적인 측정이거나, 특정 도메인에서의 검색에 대해서는 질의어의 연관관계는 차이가 있다. 최근 자연어 처리 분야에서 많은 연구가 진행 중인 워드임베딩(word embedding)은 컴퓨터가 자연어를 인지할 수 있도록 단어 간 의미와 맥락에 따라 상관관계를 다차원 벡터로 수치화 하는 방법으로 벡터 연산을 통해서 추론까지 가능하다[4].

워드임베딩은 인공신경망 기반 학습모델로서, NNLM (neural net language model), RNNLM (recurrent NNLM), word2vec 등이 있다. NNLM은 학습 단어의 수를 정해주어야 하며, 이전의 단어들에 대해서만 학습할 수 있다. RNNLM은 단어를 순차적으로 입력하는 방식으로, 각 스텝이 Short-Term 메모리 역할을 하면서 이전 단어들과의 상관관계를 유지하면서 학습한다. word2vec은 기존 모델에 비해 학습 속도를 개선하여, 현재 가장 많이 사용하는 워드임베딩 모델이다[5]. 한글 워드임베딩은 반드시 핵심 단어인 명사를 효율적으로 추출하는 과정을 수행해야 하며, 만약, 명사 추출과정을 거치지 않으면, 조사, 부사, 관형사 등 학습에 필요하지 않은 단어까지 학습하게 되어 효율적인 임베딩 결과를 도출할 수 없게 된다[6].

본 연구는 한글 워드임베딩 및 아프리오리(apriori)와 한글 텍스트 마이닝을 이용하여, 특정 도메인의 사용자 질의에 대해 보다 효율적인 질의어 확장과 답변을 검색할 수 있는 방법을 제안하였다. 본 연구의 한글 워드임베딩, 아프리오리, 텍스트 마이닝은 빅 데이터 통계 분석 및 그래픽 등으로 사용되는 R을 이용하였다. 또한, 시뮬레이션을 위해 샤이니(shiny) 패키지를 이용하여, 웹을 통해 인터렉티브하게 결과를 확인하였다[7].

본 연구의 워드임베딩과 아프리오리는 사용자 질의어에 대해서 의미와 맥락이 유사한 연관 단어를 추출하는 단계로서, 각 단어와 상관관계가 높은 두 단어 또는 세 단어를 추가하여, 사용자 질의를 확장하는 단계이다. 워드임베딩은 깃허브(git hub)의 word2vec를 사용하였다[8]. 한글 텍스트 마이닝은 질의어 확장에 따라 답변을 추출하는 단계로서, 특수문자, 불용어, 비속어 등을 제거하기 위해 "tm" 패키지를 이용하였으며, 효율적인 명사추출과 단어빈도수(term frequency), 역문서빈도수(inverse document frequency)를 이용한 단어 가중치를 추출하기 위해서 "KoNLP" 패키지를 이용하였다. 또한, 코사인 유사도(cosine similarity) 알고리즘으로 질의와 답변의 유사도 계산을 하여, 최종 답변을 추출하였다.

본 논문은 2장에서 관련연구로서 질의어 확장과 답변 유사도 알고리즘에 대해서 서술하였다. 또한, 3장에서 질의응답 시스템을 제안하였으며, 4장에서 실험 결과를 나타내고, 마지막으로 결론에 대해서 서술하였다.

Ⅱ. 질의어 확장 및 텍스트 마이닝

2-1. 질의어 확장

자연어 처리(natural language processing)는 컴퓨터가 인간의 언어를 학습을 통해 의미와 맥락을 분석하며 처리하는 분야이다. 일반적으로 컴퓨터는 단어를 유니코드(unicode)로 처리하기 때문에, 컴퓨터가 단어를 인지하기 위해서는 수치적인 방식으로 단어를 표현해야 한다. 기존의 one-hot encoding은 단어 수치화를 통해, 단어를 구분은 할 수 있었으나, 단어의 의미나 맥락에 대해서는 고려하지 않은 방법이다. 워드임베딩은 단어의 의미와 맥락을 고려하여, 연관단어들을 가까운 거리에 위치하도록, 다차원 공간에 수치화 하여 벡터로 표현한 것이다.

word2vec은 말뭉치(corpus)를 입력으로 텍스트를 처리하는 인공 신경망이며, 최근 가장 많이 사용하는 워드임베딩이다. word2vec은 CBOW(continuos bag of words)와 skip-gram모델이 있다. CBOW은 주변 단어가 만드는 맥락을 이용해 타겟 단어를 예측하는 것이고, skip-gram은 CBOW와는 반대 방향의 모델로서 한 단어를 기준으로 주변에 올 수 있는 단어를 예측하는 방식이다. 그림 1과 그림 2에 word2vec의 CBOW와 skip-gram 모델에 대해서 나타냈다.

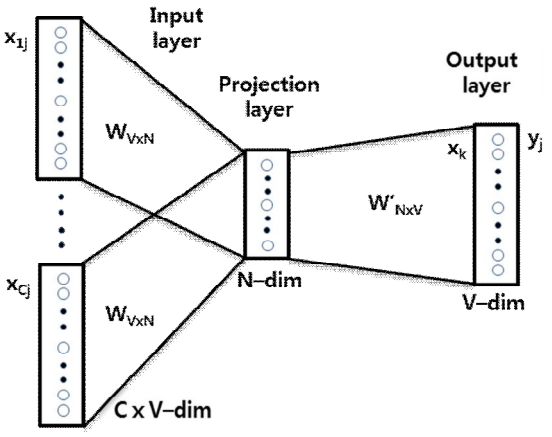

그림 1. CBOW 모델
Fig. 1. CBOW model.

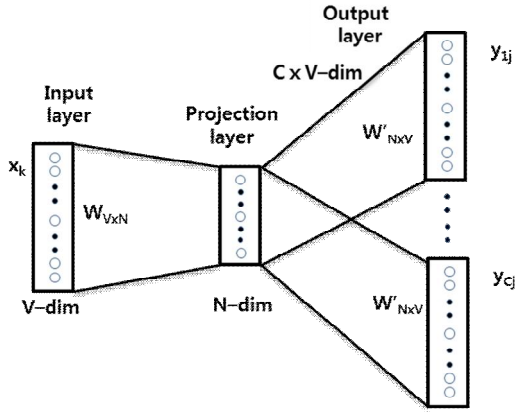

그림 2. skip-gram 모델
Fig. 2. skip-gram model.

CBOW은 입력층, 투영(projection)층, 출력층 등으로 인공 신경망이 구축되어 있다. 입력층은 학습 단어 C를 기준으로, 전후 C/2개의 단어를 one-hot encoding으로 투영시킨 후, 그 벡터들의 평균을 구해서 투영층에 보낸다. 투영층은 N개의 노드를 이용하며, 투영 결과와 가중치 곱한 후, 출력층에서 소프트맥스(softmax) 함수를 이용하여 에러를 계산한다. 또한, 학습을 통해 가중치를 조절하여 에러를 줄기 위한 학습을 반복하게 된다. 따라서, 입력층과 투영층에 $V\times N$ 매트릭스가 있고, 투영층에서 출력층에는 $N\times M$ 매트릭스가 있다. 그림 1에 CBOW 모델에 대해서 나타냈다 [5].

skip-gram은 주어진 단어 주위의 단어들의 등장 여부를 유추하는 것으로, 근접해 있는 단어는 더 많은 의미가 있는 관련 있는 단어라는 개념을 적용한 모델이다. 다음은 계산량을 줄이기 위해 이진 완전트리를 이용한 계층 소프트맥스(hierarchical softmax)를 적용한 CBOW과 skip-gram 학습 전체 계산량에 대해서 나타냈다[9].

$$Q = C\times N + D\times \log_2(V) \tag{1}$$

$$Q = C(N + N\times \log_2 V) \tag{2}$$

여기서 C는 학습 단어 수이며, N은 투영층 노두 수이고, V는 사전크기 수이다. word2vec 모델은 NNLM 및 RNNLM모델의 방대한 계산량에 의한 학습 속도 문제점을 개선하였다.

아프리오리는 항목들의 연관규칙을 나타내는 알고리즘이며, 항목들의 출현 빈도를 나타내는 지지도(support)와 신뢰도(confidence)를 이용한다. 지지도는 항목들이 동시에 출현하게 될 경우의 확률이며, 신뢰도는 예측력이나 정확도에 대한 측정이다[9].

$$\text{지지도}(A) = \frac{count(B)}{N} \tag{3}$$

$$\text{신뢰도}(A\rightarrow B) = \frac{\text{지지도}(A,B)}{\text{지지도}(A)} \tag{4}$$

N은 모든 항목 집합이며, $count(B)$는 특정 항목집합의 개수이다. 아프리오리는 다음과 같은 단계로 생성된다[10],[11].

- 항목집합 S에서 결정된 최소 지지도 이상의 지지도를 가지는 항목집합들의 모든 집합들을 빈발 항목집합들을 찾아낸다.
- 모든 빈발 항목집합 S에서 공집합이 아닌 모든 부분집합을 찾는다. 각각의 부분집합 A에 대하여, 최소 신뢰도 $C_{\min}$이상의 규칙을 출력한다.

$$C_{\min} \le \frac{\text{지지도}(S)}{\text{지지도}(A)}, a\Rightarrow (S-A) \tag{5}$$

2-2. 한글 텍스트 마이닝

한글 문서비교는 질의어에 대해서 가장 유사한 답변을 추출하는 것이다. 한글 문서비교는 모든 답변 문서에 대해 말뭉치를 생성하고, 말뭉치에서 특수문자, 불용어, 비속어, 한 단어 제거 등을 이용하여 전처리한다. 또한, 중요한 단어인 명사를 추출한 후에, 단어 빈도수와 역 문서 빈도수 방법을 이용하여, 각 단어에 대한 가중치를 부여한다. 최종적으로 질의어와 가장 유사한 문장을 코사인 유사도를 통하여 추출한다.

단어 빈도수 방법은 텍스트의 각 단어에 대한 출현 빈도수를 가중치로 이용하는 방법이다[12].

$$tf(k_{ij}) = f(k_{ij}) \tag{6}$$

$$tf(k_{ij}) = 1 + \log f(k_{ij}) \tag{7}$$

$$tf(k_{ij}) = 1 + \log(1 + \log f(k_{ij})) \tag{8}$$

식 3은 가장 일반적인 가중치 공식으로, $w(k_{ij})$는 문서 i의 단어 j의 가중치로서 문서 i의 단어 j의 빈도수 $f(k_{ij})$를 가중치로 갖게 된다. 식 3은 특정 단어의 빈도수가 특이하게 높다면, 질의에 대한 답변과 유사하지 않은 경우에도 유사도가 높게 책정될 수 있기 때문에, 식 4와 식 5와 같이 상대적인 단어가중치로 보완하여 사용한다[13].

역 문서 빈도수는 전체 문서에서 나오는 특정 단어에 대한 빈도수에 대한 가중치를 부여하는 것이다. 즉, 모든 문서에서 사용되는 단어는 불용어일 가능성이 크기 때문에, 가중치를 낮춰주고, 소수의 문서에서 사용되는 단어는 높은 가중치를 주어야 할 것이다. 역 문서 빈도수의 공식은 다음과 같다.

$$idf(j) = \log(1 + \frac{N}{f(j)}) \tag{9}$$

단어 가중치는 일반적으로 단어 단어빈도와 역 문서빈도를

곱하여 결정한다. i 문서의 단어의 가중치는 다음과 같다.

$$tfidf(i,j) = tf(i,j) \times idf(j) \tag{10}$$

코사인 유사도는 텍스트 유사도 측정에 적합한 알고리즘으로서, 단순하고 계산이 빠르며, 검색성능을 향상시키고, 부분정합으로 질의에 근접한 답변 검색이 가능하다.

$$\frac{d_x \cdot d_y}{|d_x| \times |d_y|} = \frac{\sum_{a=1}^{t} w(k_{xa}) \times \sum_{a=1}^{t} w(k_{ya})}{\sqrt{\sum_{a=1}^{t} w(k_{xa})^2} \times \sqrt{\sum_{a=1}^{t} w(k_{ya})^2}} \tag{11}$$

Ⅲ. 제안 모델

제안모델은 사용자 질의, 한글 워드임베딩, 텍스트 마이닝으로 구분 된다. 사용자 질의어는 사용자의 질의 유형과 사용자에 적합한 답변을 분석하기 위해 데이터베이스에 저장된다. 한글 워드임베딩은 word2vec 인공 신경망 알고리즘을 이용하여, 단어 간 상관관계를 추출하여 질의어를 확장하는 부분이다. 다음은 워드임베딩 과정이다.

- 한글 워드임베딩을 위해 모든 답변들을 전처리과정을 거쳐 하나의 데이터로 통합되며, 통합된 데이터를 읽어서 tm 패키지를 이용하여 특수문자, 불용어, 숫자 등을 제거한다.
- word2vec 학습은 단어별로 학습을 하기 때문에, 핵심 단어인 명사를 추출하는 과정을 수행해야 한다. 만약, 명사 추출 과정을 거치지 않으면, 조사, 부사, 관형사 등 학습에 필요하지 않은 단어까지 학습하게 된다. 이를 위해, 전처리 과정에서 생성된 데이터에서 명사를 추출한다. 또한, 한글자이거나 5글자 이상의 명사를 제거하였다.
  - 모든 전처과정을 수행한 후, word2vec학습을 통해 워드임베딩 모델을 생성한다.
- 질의어 확장 단계로서, 사용자 질의어에서 명사만을 추출하여 word2vec 모델에 입력하여, 의미와 맥락이 유사한 단어들을 추출한 후, 유사도가 높은 두 단어 혹은 세 단어를 추출한다.

한글 텍스트 마이닝은 질의어와 워드임베딩 모델에서 추출된 연관단어를 핵심 질의어로 문서비교 그리고 답변을 추출 부분이다. 다음은 텍스트 마이닝 과정이다.

- 사용자 질의어 및 확장 질의어와 전체 답변을 하나의 말뭉치를 생성하고, 명사를 추출한다. 이때 워드임베딩 과정과 마찬가지로 명사 추출과정에서 보다 정확한 답변을 검색하기 위해서 특수문자, 불용어, 비속어 한 단어 등을 제거하였다.

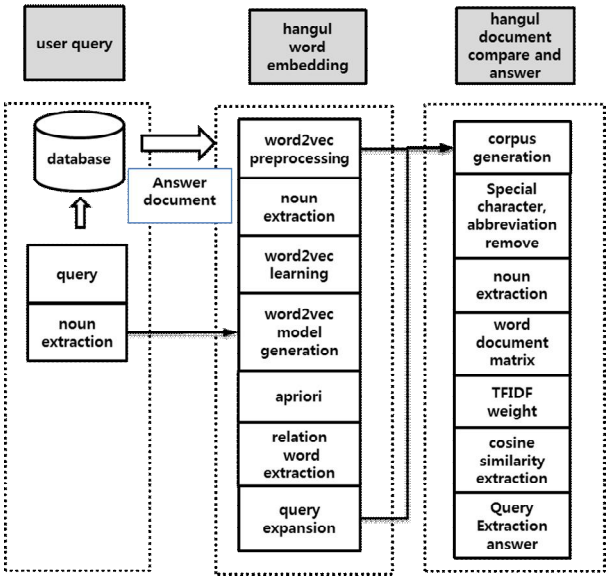

그림 3. 제안 모델
Fig. 3. Proposed system.

- 질의어와 답변 문서에 대한 단어-문서 매트릭스를 생성한다. 단어문서 매트릭스의 값은 각 단어에 대한 각 문서의 TF-IDF 가중치이다.
- 코사인 유사도 알고리즘으로 사용자 질의어 및 확장 질의어에 대한 답변들의 유사도를 측정하여, 유사도가 가장 높은 답변을 출력한다.

제안모델은 시뮬레이션을 위해 웹과 상호작용하고 웹을 통한 질의를 가능하게 함으로써, 분석결과와 그래프를 실시간으로 확인할 수 있는 사이니 패키지를 사용하였다. 샤이니는 사용자 인터페이스인 ui.R과 서버인 server.R 스크립트로 구현된다. 다음은 ui.R 스크립트의 일부분이다.

```
shinyUI(pageWithSidebar(
  headerPanel("자동응답시스템"),
  sidebarPanel(
    textInput(inputId = "question", label = "question",
     value = "질문 작성"),
   radioButtons("doc", "질의어확장시스템",
           c("TermDocument frequency",
             "질의답변유사도",
             "질의답변",  "유사답변보기",  "word2vec",  "apriori",
"word2vecplot", "wordcloud","clustering"), selected = ""),
   numericInput("num", "상관관계 단어수", 3),
   submitButton("Submit")
 )
    mainPanel(
  textOutput("textDisplay") ,
  verbatimTextOutput("term"),
  plotOutput("plotDisplay")
 )))
```

그림 4. 사용자 인터페이스
Fig. 4. User interface.

그림 4에 제안시스템의 유저인터페이스에 대해서 나타냈다. 그림에서 "TermDocument frequency"은 모든 답변에 대해서 단어 빈도수를 내림차순으로 정렬하여, 출력하여 특정 도메인에서 많이 사용되는 단어들을 확인하여, 주요 질의어를 유추할 수 있다. "질의 답변유사도"는 질의어와 각 답변 간에 유사도 결과를 출력하는 기능으로, 사용자 질의어 뿐 아니라 워드임베딩을 통해 질의어를 확장하여 유사도를 추출할 수 있다. "질의 답변"은 제안시스템의 핵심 기능으로서 사용자 질의어 뿐 아니라 워드임베딩을 통해 질의어를 확장하여 보다 정확성 높은 답변을 추출할 수 있다. "유사답변보기"는 유사도 순서대로 정해진 개수만큼 답변을 출력하는 기능이다. "word2vec" 기능은 특정 도메인의 모든 답변을 이용하여, 생성된 워드임베딩 모델을 이용하여, "상관관계 단어수"에 입력된 값 만큼 사용자 질의어와 의미와 맥락이 유사한 단어를 추출한다. "apriori"는 장바구니 분석을 통해 질의어와 연관이 있는 단어를 추출하는 기능으로, 또 다른 질의어 확장 기능이다. "word2vecplot"는 각 단어의 코사인 거리를 2차원으로 도식한 것이다. "wordcloud"는 단어의 빈도수에 따라 단어의 크기를 도식한 것으로 특정 도메인에서 많이 사용되는 주요 단어들을 화면을 통해 확인할 수 있다. 다음은 워드임베딩과 한글 문서 비교 및 답변에 대한 샤이니 서버 코드의 일부이다.

```
  if(input$doc=="word2vec") {     # 워드 임베딩
    query<-extractNoun(input$question)
    prep_word2vec("./data/","./res1/co.txt",lowercase=T)
   # 특수문자, 불용어 제거
    noun <- sapply(noun, function(x) {Filter(function(y) {nchar(y)
<= 4 && nchar(y) > 1 && is.hangul(y)},x)} )
  write(unlist(noun), "./res1/co.txt")
    model=train_word2vec("./res1/co.txt", output="./res1/co_vectors.
bin",  threads = 3,vectors = 30, window=10)
    n<-1
    while(n<=length(query)) {
    k<-as.matrix(nearest_to(model,model[[query[n]]],3))
    print(k[0:3,])
    n<-n+1
  }  }
```

```
 if(input$doc=="질의답변") {  # 한글 문서 비교 및 답변
  query<-input$question                    # 질의 입력
  corp<-Corpus(VectorSource(query))
  docst<-c(docs, corp)                     # 코퍼스 병합
 # 특수문자, 불용어
 tdmat<-TermDocumentMatrix(docst,     control=list(tokenize=kon,
wighting=function(x)weightTfIdf(x,TRUE),wordLengths=c(2,Inf)))
 m <- as.matrix(tdmat)
  norm_vec <- function(x) {x/sqrt(sum(x^2))}
  tdmat <- apply(m, 2, norm_vec)
  docord <- t(tdmat[,k]) %*% tdmat[,1:k-1]  # 코사인 유사도
  docord_max<-max.col(docord)      # 유사도가 가장 높은 답변
  print(txt[docord_max])                   # 답변 출력
}
```

input$은 사용자 인터페이스에서 문자를 입력받기 위한 코드이며, 입력받은 문자는 wod2vec와 사용자 질의어이다. extractNoun(input$question)은 사용자 질의어에 대해서 명사를 추출하는 과정이다. prep_word2vec()은 word2vec 전처리 과정으로 모든 답변을 합치고, 영문자를 모두 소문자로 변환하여 하나의 데이터로 통합하는 과정이다. 전처리과정에서 생성된 통합 데이터 워드임베딩 모델을 생성하기 위해, 명사 추출 및 특수문자, 불용어 제거 등의 전처리 과정을 실행한다.

train_word2vec()은 word2vec으로 학습하여, 워드임베딩 모델을 생성한다. 최종적으로 워드임베딩 모델은 사용자 질의어 확장에 사용하였으며, 질의어의 각 단어들에 대해서 상관관계 높은 3개의 단어를 출력하였다.

한글 문서 비교 및 답변 코드에서 c(docs, corp)은 질의과 답변 코퍼스를 병합하는 코드이며, TermDocumentMatrix()는 코퍼스에서 명사추출하고 TF-IDF 처리하여, 단어 매트릭스 형태로 가중치를 출력하는 코드이다. 본 제안 모델은 각 가중치에 대해서 다음과 같은 식으로 정규화하였다.

$$w(k_{ij}) = \frac{f(k_{ij})}{\sqrt{\sum_{j=1}^{k} f(ij)^2}} \tag{12}$$

이와 같이 제안 모델은 정규화 결과를 이용하여, 코사인 유사도로 질의어와 워드임베딩을 통해서 추출된 확장 질의로 부터 모든 답변들에 대해서 유사도를 추출하였으며, 최종적으로 유사도가 가장 높은 답변을 출력하였다.

Ⅳ. 결과 및 검토

본 연구는 윈도우 환경에서, 통계 분석 및 그래픽 등으로 사용되는 R과 개발환경 툴인 RSudio를 이용하였으며, 결과를 시뮬레이션하기 위해 웹과 상호작용하고 분석결과와 그래프를 실시간으로 확인할 수 있는 사이니 패키지를 사용하였다. 또한, 한글 워드임베딩, 한글 문서비교 및 답변추출 부분을 시뮬레이션하기 위해서 word2vec, tm, KoNLP 등의 R 패키지를 사용하

였다. 제안모델은 사용자 질의어와 답변에 대해서 추후에 질의 내용을 분석하고, 답변을 추가하기 위해 DBI_0.3.1, RMySQL 패키지를 사용하여, MySQL 데이터베이스와 연동하였다. 특히, 본 연구는 제안시스템의 질의어 확장에 대한 시뮬레이션을 위해서 항공, 철도, 버스 등에 대한 질문에 해당하는 답변을 사용하였다. 그림 5에 모든 답변문서에 대한 단어 빈도수를 내림차순으로 정렬하여 출력한 결과를 나타냈다. 여기서 모든 답변에 대해서 빈도수가 높은 질의어를 검색할 수 있으며, 전 처리 과정에서 찾지 못했던 불용어를 찾을 수 있다. 그림 6에 word2vec을 이용한 질의어 확장에 대해서 나타냈으며, 그림 7에 아프리오리를 이용한 연관단어추출에 대해서 나타냈다.

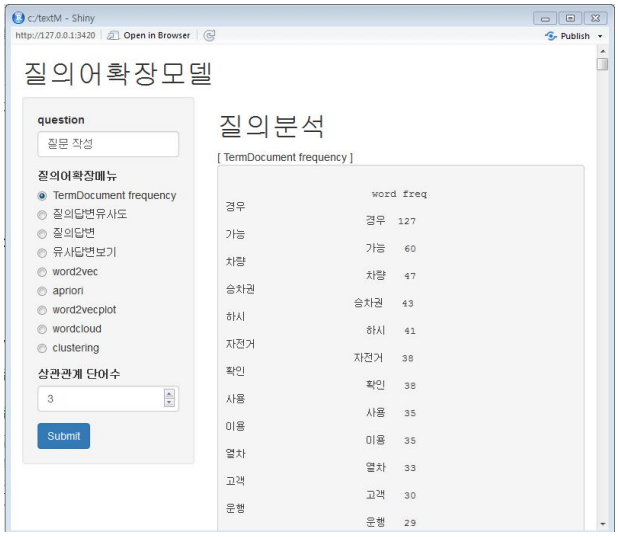

그림 5. 모든 단어와 단어빈도수
Fig. 5. All words and word frequency.

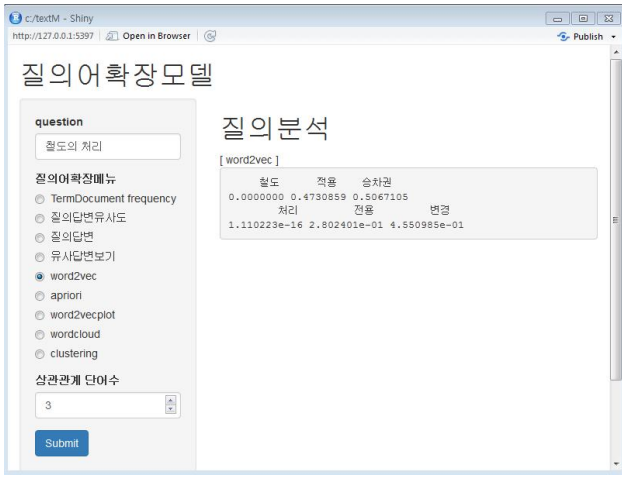

그림 6. 한글 워드임베딩 질의어 확장
Fig. 6. Hangul word embedding query expansion.

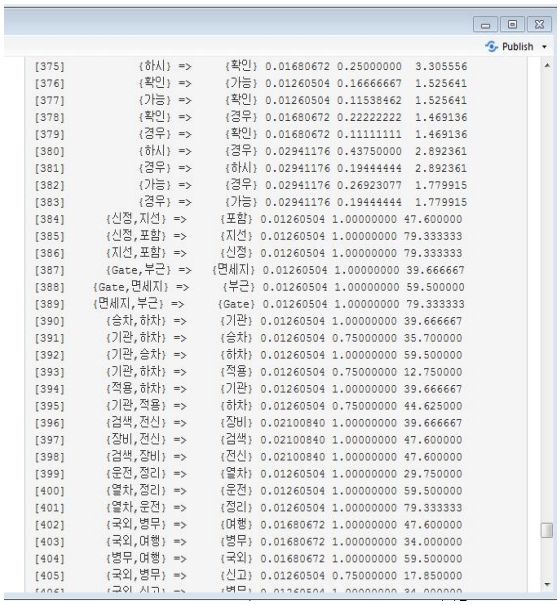

그림 7. 아프리오리 질의어 확장
Fig. 7. Apriori query expansion.

그림 6은 본 논문의 핵심 기능인 word2vec를 이용한 한글 워드 임베딩 질의어 확장 기능으로서, 사용자 질의어에 대한 질의어 확장 결과이다. 워드임베딩을 위한 인공신경망 학습은 학습량이 많기 때문에, 3개의 멀티 쓰레드(thread)를 생성하여 처리하였으며, 워드임베딩 벡터는 15 차원이고, 학습하고자 하는 데이터의 윈도우는 10로 설정하였다. 이것은 처리하고자 하는 답변의 크기에 따라 실험을 통해 적절하게 설정할 수 있다. 그림에서 알 수 있듯이 word2vec를 이용한 한글 워드임베딩은 사용자 질의어에 포함된 모든 단어에서 명사를 추출하여, 추출된 명사를 키워드로 하여 의미와 맥락이 가장 가까운 단어를 추출하며, 코사인 거리(cosine distance)를 결과로 출력함을 알 수 있다. 예를 들어 그림에서 "철도의 처리"이라는 2개의 질의어에 대해서 모두 6개의 질의어로 확장할 수 있으며, 이러한 연관 질의어의 개수는 "상관관계 단어수"를 통하여 사용자가 선택할 수 있다.

그림 7은 또 다른 질의어 확장기능으로서 아프리오리를 이용한 연관단어 출력 결과이다. 그림에서 질의어에 해당하는 단어 연관성을 위해 지지도 0.1과 신뢰도 0.1로 단어 연관성 규칙 결과를 출력했다. 아프리오리 출력 결과는 워드임베딩과 마찬가지로 단어 연관성을 통해 질의어 확장과 복합명사 등을 추출할 수 있으며, 이러한 확장 질의어를 이용하여, 비교문서들의 유사도를 측정하여 순위화하여 출력할 수 있다.

그림 8에 질의 답변 유사도에 대해서 나타냈다. 그림에서 사용자 질의어에 대한 전체 답변 유사도에 대해서 나타냈다. 이때 word2vec를 이용해 사용자 질의 뿐 아니라 상관관계가 있는 단어와 함께 유사도를 추출할 수 있다. 그림에서 "철도의 처리 전용 변경"이라는 질의에 대한 답변 유사도에 대해서 나타냈다.

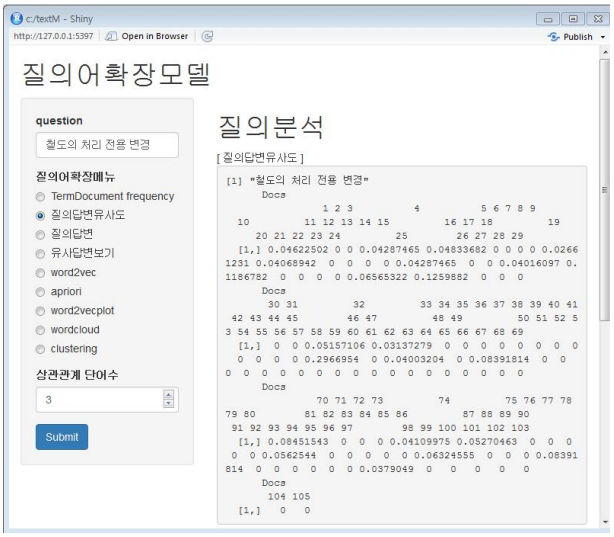

그림 8. 질의 답변 유사도
Fig. 8. Question and answer similarity.

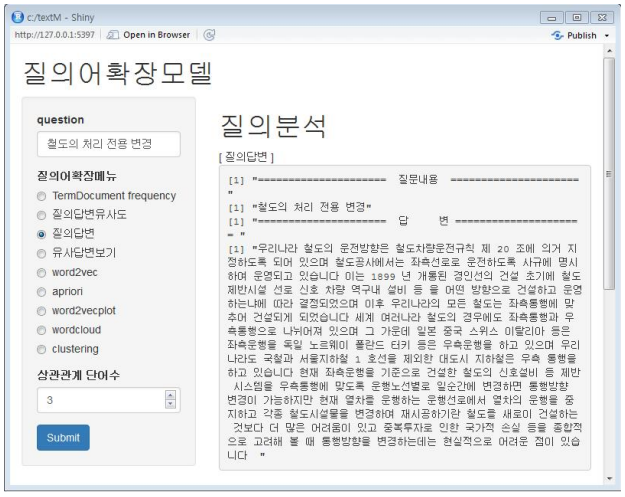

그림 9. 질의 답변
Fig. 9. Question and answer query expansion.

그림 9의 질의 답변은 사용자 질의단어 뿐 아니라 코사인 거리가 가장 가까운 두 단어 “전용 변경”을 추가 확장하여 가장 높은 유사도를 가진 답변을 출력한 예이다. 또한 유사 답변보기는 사용자가 원하는 정확한 답변이 아닐 수 있으므로, 답변 수를 적절히 설정하여 유사도가 높은 순으로 다른 답변을 보여주기 때문에, 보다 더 답변 적중률을 높일 수 있다.

Ⅳ. 결론 및 향후 과제

검색시스템에서 사용자 질의어는 사용자가 질의어를 직접 선정하여 입력해야 하며, 검색을 위한 적합한 질의어를 연상하는데 어려움이 있다. 본 연구는 word2vec를 이용한 워드임베딩과 아프리오리를 이용하여, 검색시스템에서 사용자 질의어를 확장할 수 있는 모델을 제안하였다. 한글 워드임베딩은 명사 추출과정을 거치지 않으면, 조사, 부사, 관형사 등 학습에 필요하지 않은 단어까지 학습하게 된다. 제안모델은 이를 위해 전처리 과정에서 생성된 파일을 읽어 특정 도메인의 모든 답변에 대해서 tm 및 KoNLP 패키지 등을 이용하여, 중요 키워드인 명사를 추출하였으며, 텍스트 마이닝을 통하여 효과적으로 답변을 추출하는 과정을 사이니 패키지를 사용하여 확인하였다. 제안모델은 특정 도메인의 답변을 학습하여, 연관성이 높은 질의어를 확장함으로서 답변의 정확성을 높일 수 있다. 본 연구는 적은 데이터로 시뮬레이션을 하였으나 보다 큰 도메인의 방대한 데이터를 가진 포털 사이트나 검색시스템의 질의어 확장을 위한 모델로 응용할 수 있다. 본 연구는 향후 방대한 빅데이터 학습을 통한 워드 임베딩 구축이 필요하며, 데이터베이스에 저장된 사용자 질의를 분석하여, 사용자 중심의 연관성 높은 질의어를 추출하는 방법과 더불어 특정 도메인에서의 사용자 요구 분석 및 예측에 관한 연구가 필요하다.

참고 문헌

[1] Y. A Kim, G. W. Park, “An efficient extended query suggestion system using the analysis of users' query patterns,” *Korea Institute of Communication Sciences,* Vol. 37, No. 7, pp. 619-626, June. 2012.

[2] Z. Mai, G. Pant, and O. R. Liu Sheng, “Interest-based personalized search,” *ACM Transactions on Information systems*, Vol. 25, No. 1, pp. 1-38, Feb. 2007.

[3] C. Buckley, G. Salton, and J. Allan, “The effect of adding relevance information in a relevance feedback environment,” *in Proceedings of 17th annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, Dublin : Ireland, pp. 292-300, July. 1994.

[4] J. Garten, K. Sagae, V. Ustun, “Combining distributed vector representations for words,” *in Proceedings of NAACL-HLT*, Denver: CO, pp. 95-101, May. 2015.

[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *in proceeding of Neural Information Processing Systems 26*, Lake Tahoe: NV, pp. 3111-3119, Dec. 2013.

[6] M. Tomas, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” in *Proceeding of International Conference on Learning Representations*, Scottsdale: AZ, pp. 01-09, May. 2013.

[7] B. Chris, *Web application with R using shiny*, 1st ed. Birmingham, England: Packt Publishing, pp.47-72, Oct. 2013.

[8] [Internet]. Available: https://github.com/bmschmidt/word

Vectors

[9] M. Andriy, and G. Hinton. “A scalable hierarchical distributed language mode,” in *Proceeding of Neural Information Processing Systems 21*, Vancouver: British Columbia, pp.1081-1088, Dec. 2008.

[10] Y. Kim, “A study on design and implementation of personalized information recommendation system based on apriori algorithm,” *Journal of Korean BIBLIA Society for Library and Information Science*, Vol. 23, No. 4, pp. 283-308, Dec. 2012.

[11] S. J. Ko, and J. H. Lee, “Weighted bayesian automatic document categorization based on association word knowledge base by apriori algorithm,” *Journal of the Korea Multimedia society*, Vol. 4, No. 2, pp. 171-181, Apr. 2001.

[12] H. S. Kim, S. C. Park, and S. H. Kim, “Measurement of document similarity using term/term-pair features and neural Network,” *Journal of Korean Institute of Information Scientists and Engineers*, Vol. 31 No. 12, pp. 1660-1671, Oct. 2004.

[13] D. Y. Park, “Pushing ahead context and project of capability education using national competency standards,” *Korea Research Institute for Vocational Education and Training, The Human Resources Development Review*, Vol. 16, No. 3, pp. 52-71, Sep. 2013.

신 동 하 (Dong-Ha Shin)

2016년 2월 : 가천대학교 에너지IT학과 (공학사)
2016년 3월 ~ 현재 : 가천대학교 대학원 IT융합공학과 석사 과정
※ 관심분야 : 딥러닝, 빅 데이터, IOT, 로봇제어, 로봇 액추에이터

김 창 복 (Chang-Bok Kim)

1986년 2월 : 단국대학교 전자공학과 (공학사)
1989년 2월 : 단국대학교 전자공학과 (공학석사)
2009년 2월 : 인천대학교 컴퓨터 공학과 (공학박사)
1994년 ~ 현재 : 가천대학교 IT대학 에너지 IT학과 교수
※ 관심분야 : 빅 데이터 마이닝, 분산처리시스템, 사물인터넷, 마이크로그리드