KIPS Trans. Softw. and Data Eng.
Vol.9, No.6 pp.195~204 pISSN: 2287-5905



https://doi.org/10.3745/KTSDE.2020.9.6.195

# An Advanced Search that Converts Natural Language into the Logic Advanced Search and with Developed History Search Method

Daehong Lee† · Hansuk Yu† · Sangwon Park††

## ABSTRACT

Nowadays there are over 1.6 billion web pages and it is hard to get necessary results that user wants. Most search engines allow you to search with logical form to get accurate results. However, normal users are not familiar to search information as logical form. Therefore, they search in natural language rather than in complicated logical form. In this paper there are some suggestions to improve quality of searching results, converting natural language input by the user into logical form which can able to use advanced search engine. Users tend to make short searches due to the 'Simplicity' which is one of the features of the search form. Therefore we suggest history retrieval method; advanced version of previous suggestion to provide convenience to the normal users. We had improvement on accuracy of the search results converting natural languages to logical form and also can contain every keyword without missing any keywords using searching methods on this paper. It is expected that these search methods will contribute to the development of search engines.

Keywords : Search Engine, Advanced Search, Natural Language Search, History Search

# 자연어의 논리식으로의 변환을 이용한 고급검색 및 이를 활용한 히스토리 검색

이 대 홍† · 유 한 석† · 박 상 원††

## 요 약

현재 웹에서 존재하는 웹페이지는 16억개 이상이며 이중에서 원하는 검색결과를 얻기란 쉽지 않은 일이다. 대부분의 검색엔진에서는 정밀한 검색결과를 제공하기 위하여 논리식의 형태로 검색할 수 있게 하고 있다. 하지만 일반적인 경우 사람들은 원하는 정보를 논리식 형태로 검색하는데 익숙하지 않다. 때문에 복잡한 논리식 형태로 검색하기 보다는 자연어로 검색한다. 따라서 본 논문에서는 사용자가 입력하는 자연어 질의를 검색엔진의 고급검색을 사용할 수 있는 논리식으로 변환하여 검색결과의 품질을 향상시켜주는 검색방법을 제안한다. 또한 사용자들은 검색형태의 특징 중 하나인 단순성에 의해 길게 검색하기 보다는 여러 번의 짧은 검색을 이용하는 경우가 훨씬 많다. 이에 따라 사용자들에게 편리성을 제공하기 위하여 앞에서 제안한 검색방법을 활용한 히스토리 검색방법을 제안한다. 본 논문의 검색방법들을 사용한 결과 자연어 상태의 검색결과보다 논리식으로 변환한 검색결과의 정확도가 개선되었고 누락되는 키워드 없이 사용자가 검색하고자하는 모든 키워드를 반영할 수 있다. 이러한 검색방법이 검색엔진의 발전에 기여할 것으로 기대한다.

키워드 : 검색엔진, 고급검색, 자연어 검색, 히스토리 검색

## 1. 서 론

1982년 최초로 인터넷이 대한민국에 도입되었다. 그 후 각 가정마다 컴퓨터가 도입되고 스마트폰으로 발전해가면서 인터넷은 사람들의 삶에서 빠질 수 없는 존재가 되었다. 2008년도만 하더라도 젊은층 중에서 76.5% 정도가 인터넷을 사용하였지만 2018년도에는 전 연령대에서 91.5%의 사람들이 사용하고 있을 정도로 널리 보급되었다[1].

인터넷의 보급이 확대됨에 따라 정보의 양은 기하급수적으로 증가되고 있다. 웹 서버 조사기관인 Netcraft에 의하면 2019년을 기준으로 웹 페이지는 약 16억개 이상으로 추정되고 있으며 그 양은 계속적으로 증가하고 있다[2]. 정보의 양이 증가함에 따라 사용자는 많은 양의 정보 보다는 양질의 정보를 얻는 것이 중요하게 되었다[3]. 하지만 현재의 검색방법

※ 이 연구는 2017년도 한국외국어대학교 교내학술 연구비의 지원에 의해 이루어진 것임.
† 준 회 원 : 한국외국어대학교 정보통신공학과 학사
†† 종신회원 : 한국외국어대학교 정보통신공학과 교수
Manuscript Received : January 14, 2020
First Revision : February 28, 2020
Accepted : March 11, 2020
* Corresponding Author : Sangwon Park(swpark@hufs.ac.kr)

으로는 많은 양의 검색결과에서 양질의 정보를 얻기까지 많은 시간과 노력이 필요하다[4].

구글은 키워드만으로는 좋은 품질의 검색결과를 얻을 수 없으며 쓰레기 정보로 인해 사용자가 진정으로 관심있어 하는 정보가 가려진다고 하였다[5]. 이에 구글은 검색결과의 질을 높이기 위하여 웹 문서에서 상대적 중요도에 따라 가중치를 부여하는 페이지 랭크 알고리즘[5]을 사용하였고, 또한 주기적으로 페이지들을 업데이트한 후 품질을 새로 평가하여 더 나은 컨텐츠를 보유하면 랭킹을 올려주거나 가치가 없는 사이트들의 랭킹을 떨어트리거나 삭제시키는 구글 프레드[6]를 사용한다.

추가적으로 검색엔진을 운용하는 포털 사이트들은 키워드를 이용한 세부적인 검색을 위하여 “고급검색”이라는 기법 지원한다. 이것은 “고급검색”은 “상세검색”이라고도 하며, 검색할 때 꼭 포함되는 단어를 설정하거나 검색결과에서 특정 단어를 포함하지 않게 하는 등의 기능을 말한다. 네이버의 경우에는 현재 AND, OR, NOT으로 이루어진 불린(Boolean) 검색이 가능하며 구글의 경우에는 이외에도 해시태그나 파일타입 지정과 같은 다양한 형태의 검색 옵션을 지정해 줄 수 있다[7]. 하지만 대부분의 일반 사용자는 원하는 정보를 불리언 연산식 형태로 표현하는데 불편함을 느끼고 있다고 한다.

이에 관련하여 우리는 85명의 대학생을 대상으로 한 설문을 하였다. 이에 따르면 60%의 학생들이 고급검색 기능을 모르는 것으로 나타났다. 이는 사람들이 제한된 검색만을 이용한다는 것이다. 또한 고급검색 기법을 아는 학생들마저도 단순한 불린 연산자를 활용하거나 기간설정 같은 기법만 사용하는 것으로 나타났다. 불린 연산을 사용한 검색결과라도 양이 방대하여 사용자가 원하는 정보를 찾기 쉽지 않아 효율적인 검색이 되지 못한다. 고급검색은 검색엔진에서 제공하는 기능으로 방대한 웹에서 사용자가 원하는 정보를 상세히 기술하여 찾을 수 있게 도와주는 기능이지만 많은 사람들이 이 기능을 모르거나 알더라도 모든 기능을 알지 못하였다.

본 연구에서는 키워드만으로 검색하는 것보다 더 나은 검색결과를 제공하기 위하여 사용자가 검색 의도를 가장 정확하고 가장 편리하고 자연스럽게 표현할 수 있는 인터페이스인 자연어[8] 질의를 검색엔진의 고급검색에서 사용할 수 있는 논리식으로 번역하여 검색하는 검색 시스템을 제안한다. 또한 인터넷 사용자들은 검색을 할 때 원하는 정보를 모두 입력하여 질의의 길이가 길게 검색하기보다는 단순한 키워드를 짧게 여러 번에 걸쳐 입력하여 검색을 하는 경향이 있다[9]. 이를 검색형태의 단순성이라고 한다. 이러한 점을 고려하여 사용자가 처음 검색한 질의에 추가하고자하는 단순한 키워드의 정보를 입력하면 이전 질의를 기억하고 있다가 추가하고자하는 짧은 키워드를 연결하여 긴 문장과 동일한 질의를 내부적으로 만들어 사용자가 원하는 정보를 검색할 수 있게 도와주는 검색방법을 히스토리 검색방법이라고 정의하며 이러한 검색방법을 제안하였다. 본 연구에서 제시한 시스템은 자연어를 일반적인 논리식으로 변환하기 때문에 고급검색을 지원하는 구글이나 네이버와 같은 다양한 검색엔진에서 적용 가능하다.

본 논문에서 사용한 방법의 정확성을 검증하기 위하여 두 가지를 비교하였다. 첫째, 자연어로 검색한 결과와 논리식으로 변환하여 검색한 결과의 품질을 비교하였다. 둘째, 자연어로 검색하면 검색하고자 하는 모든 키워드가 반영된 검색결과를 얻는 것이 아니라, 일부 키워드가 제외된 검색결과를 얻게 된다. 이것을 누락된 키워드라고 하는데, 본 논문에서는 검색결과의 품질을 비교하는 두 번째 방법으로 누락된 키워드의 개수와 총 검색결과에서 누락된 키워드로 얻은 검색결과의 비율을 비교하였다. 본 논문에서 사용한 예제를 적용할 경우, 기존의 자연어 검색은 38.6%의 정확도를 보였고 본 논문에서 제안하는 방법을 이용할 경우에는 87%의 정확도를 보여 본 논문에서 제안한 방법을 사용하면 자연어 검색보다 정확도가 많이 향상되는 것을 알 수 있었다. 누락된 키워드의 수를 비교하는 경우, 기존의 자연어 검색은 77%의 누락율을 보였지만 제안한 시스템에서는 0%의 누락율을 보였다. 이것은 본 논문에서 제시한 방법을 이용할 경우 사용자가 요구하는 키워드가 검색결과에 모두 반영되었다는 것을 의미한다. 이에 따라 자연어의 논리식으로의 번역을 통해 검색결과의 정확도가 크게 향상되었으며 검색하는 내용의 키워드들을 잘 반영하여 정확한 검색이 가능하다고 볼 수 있었다.

## 2. 관련 연구

메타 검색엔진은 자체적인 데이터베이스를 갖지 않고 다른 검색엔진들에서 찾은 결과를 사용자에게 보여주는 검색엔진이다. 키워드 검색 쿼리를 전송하면 서버가 이를 받아 미리 지정한 검색엔진들에 맞는 질의로 변환한 후, 검색결과를 검색엔진으로부터 받아서 결과를 사용자에게 보여준다[10, 11]. 메타 검색엔진은 한 번의 키워드 입력으로 여러 검색엔진의 출력 결과를 모두 얻을 수 있지만, 여러 개의 검색엔진에서 얻은 결과가 중복될 수 있어 사용자에게 혼란을 일으킬 수 있다. 또한 이것은 다수의 검색엔진에 질의하여 결과를 구하므로 검색속도가 느리다는 단점이 있다.

키워드형 검색엔진은 찾고자 하는 정보와 관련된 핵심어로 검색하는 방법이다. 즉, 검색어를 입력하여 정보를 찾는 방법이다. 키워드형 검색엔진은 현재 가장 널리 사용되는 방식으로 구글, 네이버, 야후 등 거의 모든 검색엔진에서 지원하고 있다. 이는 많은 사이트들에 대한 정보를 데이터베이스로 구축해 두었다가 사용자가 검색어를 입력하면 해당 사이트를 찾아주는 방식을 사용한다. 그러나 필요로 하는 정보의 명칭(용어)을 알지 못하는 경우에는 정보 검색에 상당한 시간이 소요된다[12].

구글과 같은 키워드 검색엔진에서는 정보를 더욱 정확하게

1) 구글 검색엔진의 경우 표에 나타난 것 이외에도 웹페이지의 제목에서 검색하거나, 날씨, 지도 등에서 찾는 등의 기능을 제공한다.

Table 1. Function of search engine

| | 구글[1)] | 야후 | 네이버 | 다음 |
|---|---|---|---|---|
| AND | O | O | O | O |
| OR | O | O | O | X |
| NOT | O | O | O | X |
| 최근 업데이트 | O | O | O | O |
| 숫자범위 | O | X | X | X |
| 파일타입 지정 | O | O | X | X |
| 사이트내 검색 | O | O | X | O |
| 해시태그 | O | X | X | X |

찾아낼 수 있도록 AND, OR 등 여러 가지 검색옵션을 지정할 수 있으며 검색결과에 대한 신뢰도 점수나 가중치를 보여주기도 한다[13]. 그러나 검색결과 집합의 크기를 제어하기가 힘들어 검색결과가 너무 많거나 전혀 없는 경우가 있다[14].

그렇기 때문에 구글이나 네이버와 같은 검색엔진들은 Table 1에서 보는 바와 같이 고급검색 기능을 제공한다. 고급검색은 검색할 때에 꼭 포함되는 단어를 설정하거나 검색결과에서 제외할 단어 등을 설정하는 기능이다. 각 검색엔진들은 Table 1에서 제시하는 모든 연산자를 지원하지는 않는다. 예를 들어 네이버의 경우에는 AND, OR, NOT으로 이루어진 불린 검색만이 가능하다. 반면에 구글 검색엔진에서는 불린 연산자를 이용하는 논리식 외에도 추가적인 논리식이 표현가능하다.

구글의 경우 특정 웹페이지에 존재하는 것만 검색하고 싶을 때는 “site:”라는 단어에 특정 웹페이지의 도메인을 입력하면 지정한 웹페이지에서의 검색 결과만을 필터링하여 보여준다. 또는 “filetype:pdf”와 같은 파일형식도 지정 가능하다. 이러한 특정 웹페이지에서의 검색이나 파일형식 지정 등의 검색을 제한하는 수식어를 사용하는 것은 검색질의 품질을 향상시킬 수 있다[12]. 예를 들어 유튜브 사이트에 존재하는 모든 영국남자와 치킨에 관련한 자료를 검색하기를 원하는 경우, 구글에서 제공하고 있는 고급검색을 위한 웹페이지의 메뉴 중 “다음 단어 모두 포함” 메뉴에 키워드로 “영국남자 치킨”을 입력하고 사이트를 유튜브로 한정하기 위하여 “사이트 또는 도메인” 메뉴에는 “www.youtube.com”을 입력하면 원하는 결과를 얻을 수 있다.

다른 한편으로 구글은 해시태그를 검색하는 기능을 제공하지만 네이버는 해시태그를 고려한 검색기능을 제공하지 않는다. 이처럼 각 검색엔진은 고급검색으로 제공하는 기능이 각각 다르다는 것을 알 수 있다.

## 3. 자연어의 논리식으로의 변환

이 장에서는 기존 검색엔진이 어떻게 질의를 처리하는지에 대한 설명과 본 연구에서 제공하는 Rule에 따른 자연어의 고급검색을 바탕으로 한 논리식으로의 변환에 대한 설명한다.

### 3.1 기존 검색엔지의 자연어 질의 처리

대부분의 검색엔진에서는 불린 연산자를 이용한 논리식 형태의 검색을 이용한다. 대표적인 불린 연산자로는 AND, OR, NOT이 있다. 일반적으로 검색엔진에서는 단어가 나열되어 있으면 이와 같은 단어들을 AND로 취급한다. 그 이유는 단어가 모두 등장하는 것이 우선순위가 높게 취급되기 때문이다. 그러므로 본 논문에서는 AND 연산자를 따로 두지 않고 단어를 나열하는 형태로 표현하였다. 반면에 OR의 경우에는 두 단어가 같은 수준의 우선순위로 나타나야 한다. 그러므로 OR 연산자는 제공되어야 한다. 마지막으로 단어가 나타나지 않아야하는 경우에는 NOT 연산자를 제공해야 한다. 하지만 사람들은 하나의 문장처럼 이루어진 자연어를 검색하는 경우가 일반적이다[2)]. 이러한 경우 대부분의 검색엔진에서는 공백 문자열을 기준으로 문장의 모든 단어를 AND로 처리하여 검색하게 된다.

질의 1: 인스타그램에서 치킨 중 간장을 제외하고 양념 혹은 후라이드가 해시태그된 것으로 찾아줘
질의 2: 페이스북에서 커플여행 가는데 가평을 제외하고 맛집 혹은 카페 추천해줘
질의 3: pdf 형식으로 신재생에너지는 제외하고 온실가스 에너지 보고서를 검색해줘
질의 4: 유튜브에서 돼지고기가 제외된 김치찌개 레시피를 검색해줘
질의 5: 초콜릿 선물 중에서 화이트초콜릿은 제외하고 아몬드초콜릿이 해시태그 된 것으로 인스타그램에서 검색해줘
질의 6: 유튜브에서 정렬 알고리즘 중 선택정렬을 제외하고 찾아줘
질의 7: 자바칩은 제외하고 스타벅스 음료 제조법 찾아줘
질의 8: 페이스북에서 자유한국당은 빼고 더불어 민주당의 정책 대한 기사를 검색해줘
질의 9: 홍대를 제외하고 인스타그램에서 칵테일바 추천해줘
질의 10: pdf 형식의 국가재난 대비 훈련 지침서에서 평가는 빼서 찾아줘

Fig. 1. A Question in Natural Language

Fig. 1은 본 논문에서 사용한 자연어로 된 질의들의 예이다. Fig. 1의 질의 1을 구글 검색엔진에서 검색하게 되면

“인스타그램에서 AND 치킨 AND 중 AND 간장을 AND 제외하고 AND 후라이드 AND 혹은 AND 양념은 AND 해시태그된 AND 것으로 AND 찾아줘”

2) 구글의 경우 AND는 +, OR는 |, NOT은 -와 같은 기호를 이용하여 표기하기도 한다.

로 검색된다. 하지만 질의 1을 불린 연산자로 변환하면 “인스타그램에서 -간장 (후라이드OR양념) 해시태그”와 같이 바뀌어야한다. 이러한 형태의 논리식은 모든 검색엔진에 중립적인 표현이다.

검색엔진 중 가장 대표적인 구글과 네이버의 검색엔진에서는 각각 고유한 논리식을 사용한다. 가령 Fig. 1의 질의 1을 구글에서 사용하는 고급검색 논리식으로 변환하게 되면

“치킨 -간장 #양념 OR 후라이드 site:www.instagram.com”

과 같은 형태로 표현해야한다. 마찬가지로 같은 질의를 네이버의 고급검색 논리식에 맞게 변환한다면

“치킨+-간장+후라이드+양념+해시태그”

로 표현해야 한다. 이렇게 완성된 논리식은 두 검색엔진 모두 HTTP GET 방식으로 URL에 질의를 첨부하여 결과를 구하게 된다. 아래의 Fig. 2는 Fig. 1의 질의 1을 구글에서 최종적으로 검색하는 경우의 URL이다.

```
https://www.google.com/search?&q=치킨 -간장
#양념 OR 후라이드 site:www.instagram.com
```

Fig. 2. HTTP GET Message for Google Search Engine

Fig. 2와 같은 형태의 논리식으로 변환하여 검색하는 것은 질의 1의 자연어 형태의 문장을 바로 검색한 것보다 검색 결과의 질이 향상되는 것을 알 수 있다. 하지만 검색엔진의 논리식은 일반 사용자들에게 익숙하지 않은 문법으로 표현된다. 또한 동일한 질의라도 각 검색엔진에서 표현하는 논리식이 다르기 때문에 여러 검색엔진을 이용하여 논리식으로 검색할 경우 각 검색엔진의 표현법을 익혀야 한다. 이것은 사용자의 부담을 증가시켜 논리식으로 질의하는 것을 어렵게 한다. 이와 같은 문제를 해결하기 위하여 본 논문에서는 자연어 형태의 질의를 각 검색엔진에 맞는 논리식으로 번역하는 기법을 제공하여 사용자의 편의성을 증대하려고 한다. 번역을 위해서는 먼저 자연어 상태의 문장에서 검색 키워드를 추출해야 한다.

본 논문에서는 형태소 분석기로 트위터에서 만든 한국어 처리를 위한 오픈소스인 TwitterKoreanProcessor를 사용하였다[3]. 이것은 자연어를 입력으로 받아들여서 구문트리로 바꾸거나 품사의 리스트로 POS(Part Of Speech) 값을 출력한다. POS는 각 단어의 품사를 의미한다. 우리는 자연어를 단순한 논리식으로 변환하기 때문에 문장의 의미를 파악하기 위한 구문트리보다는 키워드를 논리식으로 변환하는 것이어서 구현이 단순하여 POS 값을 이용하는 방법을 선택하였다.

3) https://github.com/twitter/twitter-korean-text

아래 예제는 Fig. 1의 질의 1을 형태소 분석기를 통하여 명사와 조사의 형태로 분리한 것이다. 질의 1의 질의문은 형태소 분석기를 통하여 분석하면

“인스타그램(Noun)/에서(Josa)/치킨(Noun)/중(Noun)/간장(Noun)/을(Josa)/제외(Noun)/하고(Josa)/양념(Noun)/혹은(Adverb)/후라이드(Noun)/가(Josa)/해시태그(Noun)/된(Verb)/것(Noun)/으로(Josa)/찾아줘(Verb)”

와 같은 리스트를 얻을 수 있다. 이러한 POS 값을 이용하여 논리식으로 변환하였다.

먼저 형태소 분석기에서 구한 토큰들 중 검색에서 사용되지 않는 불용어들을 모두 제거한 뒤 검색엔진의 논리식에서 중요하다고 판단되는 키워드만을 추출하였다. 위 예제에서 불용어로는 “에서”, “을”, “하고”, “가”, “으로”와 같은 조사가 있다. “제외”나, “혹은” 등의 키워드들은 논리식에서 NOT이나 OR에 해당하는 키워드로 앞과 뒤에 존재하는 명사와의 관계를 판단하여 논리식으로 변환한다. 위의 예제에서 “간장을 제외”라는 문장이 나오는데, 이를 논리식으로 “-간장”으로 변환하여 간장이라는 키워드가 등장하는 웹문서를 결과에서 제외되도록 한다.

### 3.2 자연어의 논리식으로 변환 규칙

본 절에서는 3.1에서 언급한 것을 토대로 본 논문에서 정의한 자연어의 고급검색을 바탕으로 한 논리식으로 번역하는 규칙들을 제시하였다. 이것은 고급검색 기능을 제공하는 검색엔진 중 가장 기능이 많은 구글을 바탕으로 한 규칙들이다. 네이버의 경우는 이와 유사하기 때문에 본 논문에서 제시하지는 않았다.

Rule 1. OR 계열의 연결사로 표현하는 경우

A (또는|혹은|이나) B → A OR B

“A 또는 B”, “A 혹은 B”, “A 이나 B”와 같이 OR 계열의 연결사로 표현된 문장의 경우 “A OR B”로 변환한다.

Rule 1의 예로 “햄버거 혹은 피자”의 경우 “햄버거 OR 피자”로 변환한다.

Rule 2. 숫자를 이용하여 범위형태로 표현하는 경우

A (~|에서) B [단위] [까지] → A..B [단위]

Rule 2는 숫자로 범위를 설정하여 질의하는 기능이다. A와 B는 숫자를 의미하며 ‘~’는 일반적으로 값과 값의 사이라는 표현으로 사용한다. 이와 같은 기호뿐만 아니라 A에서 B까지와 같이 자연어로 질의해도 동일한 의미로 처리

한다. ‘A~B’나 ‘A에서 B까지’ 들어가면 ‘A..B’로 바꿔준다. 예를 들어 “5~10만원”이라고 검색하면 “5..10 만원”이라 검색된다.

Rule 3. NOT 계열의 연결사로 표현하는 경우

A(를 (제외하고|빼고)) → -A

Rule 3은 다음 단어 제외(NOT)에 대한 기능이다. ‘제외’, ‘빼고’라는 단어가 들어간 경우 앞의 단어에 ‘-’를 붙여준다. 예를 들어 “아이폰은 제외하고 핸드폰 추천”라는 검색을 하면 “-아이폰 핸드폰 추천”으로 변환한다.

Rule 4. 파일형식을 지정하는 경우

(PDF|엑셀|워드|PPT) (형식|파일)[(으)로|에서] [검색]
→ filetype:(pdf|xls|doc|ppt)

Rule 4는 PDF, PPT 등의 파일에서 검색하기 위하여 특정 포맷의 파일 형식으로 검색하는 경우와 이를 변환하는 규칙에 관한 것이다. 예를 들어 위 규칙에는 ‘PDF|엑셀|워드|PPT’를 나타냈는데, 이것은 각각 해당 파일 형식을 의미한다. 이 형식 이외에도 다른 파일 형식을 지원할 수 있도록 하기 위하여 파일 형식을 등록하여 확장할 수 있도록 되어 있다.

현재 지원하고 있는 다른 형식으로는 ps, pwf, kml, kmz, rtf, swf가 있다. 이처럼 총 10가지의 파일 형식을 지원한다. 각 형식을 지칭할 때는 대소문자를 구분하지 않는다.

예를 들어 “인구밀도 보고서 pdf 형식으로”라고 검색하면 “인구밀도 보고서 filetype:pdf”라 검색된다. 이렇게 질의하면 PDF 문서에서 “인구밀도 보고서”라는 단어가 들어있는 것을 검색한다.

Rule 5. 사이트 또는 도메인을 지정하는 경우

(인스타그램|페이스북|유튜브|네이버)에서
→ site:(www.instagram.com|www.facebook.com|www.youtube.com|www.naver.com)

Rule 5.는 특정 사이트 내에서 검색하는 기능이다. 예를 들어 특정 단어를 페이스북에서만 검색하고자 한다면 이 기능을 사용해야 한다. 검색엔진에서는 도메인 이름을 지정해야한다. 그러나 사용자가 도메인 이름을 기억하기는 어렵다. 예를 들어 페이스북이라고 지칭하지 www.facebook.com이라고 지칭하지는 않는다. 이와 같은 사이트의 예로 네이버, 다음, 인스타그램, 유튜브, 아마존, 배달의 민족, 옥션, 11번가 등이 있다. 이와 같은 사이트들의 도메인 이름을 기억하는 것은 쉽지 않은 일이다.

이와 같은 질의를 쉽게 하기 위하여 도메인을 직접 입력하지 않고 회사 이름, 사이트 이름 혹은 서비스 이름으로 검색할 수 있도록 하였다. 새로운 서비스는 추가할 수 있도록 하여 확장 가능하도록 설계하였다. 예를 들어 “인스타그램에서 자전거 여행”이라고 검색하면 “site:www.instagram.com 자전거 여행”이라 검색된다. 이렇게 질의하면 인스타그램에서 “자전거 여행”이라는 단어가 들어있는 페이지를 검색한다는 의미이다.

Rule 6. 검색 날짜 지정

(하루|1일)이내 → &tqi=d
(한주|일주일)이내 → &tqi=w
(한달)이내 → &tqi=m
(일년|1년)이내 → &tqi=y

Rule 6은 한 달 이내의 기간과 같은 특정 기간에 작성된 문서를 검색하는 질의를 번역하는 것이다. 구글을 포함한 많은 검색엔진들은 검색일을 기준으로 정해진 기간 내의 결과만 제시하는 기능을 제공한다. 예를 들어 “하루 이내”와 같은 질의는 “tqi=d”로 변환된다. 구글과 네이버는 모두 GET방식으로 질의를 전달하기 때문에 위 질의도 URL로 변환되는데 기간에 해당하는 것은 tqi= 형식으로 표현된다. 이러한 방법으로 질의하여 검색결과를 원하는 기간에 해당하는 것으로 한정하도록 필터링 설정을 할 수 있다.

Fig. 3은 자연어의 논리식으로의 변환 과정을 그림으로 표현한 것이다.

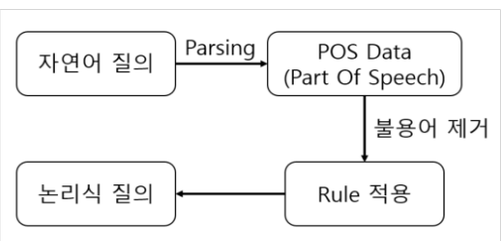


Fig. 3. The Convert Process of Natural Language into Logical Expression

Fig. 4는 Fig. 1의 질의들을 논문에서 제안하는 시스템을 통해 논리식으로 번역한 것들이다.

## 4. 검색특성을 바탕으로 한 히스토리 검색

웹 검색에서 사용자들의 질의 형태를 분석한 연구들은 대부분의 사용자들이 전문적이기 보다는 일반적이고 보편적인 용어들을 질의어로 사용하며 복잡한 검색 수식보다는 두 개 정도의 간단한 단어로 질의를 한다고 보고하고 있다[9]. 이것을 검색 형태의 단순성이라고 한다. 따라서 사람들은 일반적인 경우 이러한 단순성에 의해 몇 개의 단어로 구성된 짧은 문장으로 된 질의를 여러번 반복하여 검색을 한다. 하지

```
질의 1: 치킨 -간장 #후라이드 OR 양념
        site:www.instagram.com
질의 2: 커플여행 -가평 맛집 OR 카페
        site:www.facebook.com
질의 3: 온실가스 +에너지 +보고서
       -신재생 에너지 filetype:pdf
질의 4: -돼지고기 김치찌개 레시피
        site:www.youtube.com
질의 5: 초콜릿 선물 -화이트초콜릿 #아몬드초콜릿
        site:www.instagram.com
질의 6: 정렬 알고리즘 -선택 정렬
        site:www.youtube.com
질의 7: 스타벅스 음료 제조법 -자바칩
질의 8: -자유한국당 더불어 민주당 정책
        site:www.facebook.com
질의 9: -홍대 칵테일바 추천 site:www.instagram.com
질의 10: 국가재난 대비 훈련 지침서 -평가 filetype:pdf
```

Fig. 4. Logical Transtormation of Figure 1

만 짧은 문장으로 구성된 질의로는 정확한 검색 결과를 얻기 어렵다는 문제가 있다. 그러므로 이전 문장의 단어를 포함한 조금 더 상세한 질의를 작성하여 검색하는 것을 반복함으로써 검색 질의를 더욱 복잡하게 만들어가는 경향이 있다.

본 논문에서는 이전의 질의를 기억해 두었다가 이후에 입력하는 질의와 자동으로 결합되도록 하여 사용자는 계속적으로 짧은 질의를 통해 검색을 할 수 있도록 하는 히스토리 검색을 제안한다. Fig. 5는 이와 같은 히스토리 검색의 예이다.

Fig. 5에서 사용자는 처음에 “온실가스 에너지 보고서”에 대해서 검색하였다. 하지만 질의 1을 통하여 얻은 질의를 보고 사용자는 그 중 PDF 형식의 문서를 검색하고자 할 경우, 기존의 방법으로 질의하려면 “온실가스 에너지 보고서 pdf 형식으로”와 같이 질의하여야 한다. 이것은 질의의 단순성을 벗어나 복잡한 형태의 질의가 되는 형태이다. 본 논문에서는 질의 1을 통한 결과를 얻은 다음, 그 결과 창에서 단순히 질의 2를 입력하면 질의 1과 질의 2를 결합한 후, 이 결합된 질의를 논리식으로 번역하여 “온실가스 AND 에너지 AND 보고서 AND filetype:pdf”를 생성한 후 이것을 이용하여 질의한다. 마지막으로 질의 3으로 검색하면 질의 1, 질의 2 및 질의 3을 누적하여 질의를 생성하므로 “온실가스 AND 에너지 AND 보고서 NOT 신재생에너지 AND filetype:pdf”와 같은 논리식으로 질의하게 된다. 그 결과 구글에서 질의 1의 검색결과는 약 374,000개이며 질의 1에 질의 2를 누적한 검색결과는 약 67,400개이며 마지막으로 질의 3을 추가하였을 때는 65,700개의 검색결과가 얻을 수 있었다. 즉, 사용자가 히스토리 검색기법을 사용한다면 검색결과가 계속 주는 것을 볼 수 있다. 이것은 계속적인 질의를 통하여 사용자가 원하는 정확한 결과를 찾아가고 있는 것을 나타내며, 이때 사용하는 질의가 단순성에 기반한 질의가 되도록 하여 사용자가 복잡한 형태의 질의를 단순한 형태의 질의를 이용하여 단계적으로 만들어 나갈 수 있게 하여 편리하게 검색할 수 있게 하였다.

```
질의 1: 온실가스 에너지 보고서
질의 2: pdf 형식으로
질의 3: 신재생에너지는 제외하고
```

Fig. 5. Accumulation of Search Qqueries

## 5. 브라우저 구현

본 논문에서 제안한 방법의 효용성을 증명하기 위하여 자연어를 논리식으로 번역하는 시스템을 포함한 웹 브라우저(H 브라우저)를 Fig. 6에서 보는 바와 같이 제작하였다[4]. 본 논문에서 제안하는 논리식은 다양한 검색엔진에서 사용할 수 있다. 현재 H 브라우저는 구글과 네이버 검색엔진을 이용할 수 있도록 구현되어 있으며, 향후 다양한 검색엔진에 적용할 수 있도록 설계되어 있다. H 브라우저의 오른쪽 상단의 드롭박스를 클릭하면 사용할 검색엔진 리스트가 나온다. Fig. 6은 구글 검색엔진을 선택하여 질의한 결과이다. 그 옆의 체크박스를 선택하면 자연어를 논리식으로 번역하는 기능이나 히스토리 검색 기능 중 하나를 선택할 수 있다.

Fig. 6에서 보는 바와 같이 H 브라우저는 두 개의 뷰로 화면이 분할되어 있다. 왼쪽 뷰는 본 논문에서 제안한 방식인 자연어를 논리식으로 변환한 후 구글의 고급검색 기능을 이용하여 검색한 결과이다. 오른쪽 뷰는 자연어를 변환하지 않고 그대로 검색엔진에 질의하여 검색한 결과이다. 이것은 Fig. 1의 질의 1을 실행한 화면이다. 왼쪽 뷰는 본 논문에서 제안한 방법으로 검색한 경우로서 검색 결과가 사용자가 원하는 결과인 것을 알 수 있는데 반하여, 오른쪽 뷰는 논리식으로 변환하지 않고 자연어로 검색한 결과를 나타낸 것으로 검색엔진은 이 질의를 단순히 단어 나열로 검색하기 때문에 검색 결과가 모두 질의에 만족하는 것은 아니라는 것을 알 수 있다.

특히, Fig. 6의 오른쪽 뷰 하단에 “누락된 검색어”가 있는 것을 알 수 있다. 누락된 검색어란 구글에서 제공하는 기능으로 질의가 두 개 이상의 키워드로 이루어진 경우, 검색 결과에 반영되지 않은 단어가 있음을 나타낸다. 이 예제에서는 “치킨, 간장, 양념, 혹은, 후라이드, 줘”가 누락된 키워드로서 검색결과에 이 단어들이 들어간 것은 없다는 것을 의미한다. 누락된 키워드가 있다는 것은 검색결과가 사용자 원하는 것이 아니라는 것을 뜻한다. 그런데 Fig. 6의 왼쪽 뷰에서 보는

4) https://github.com/Chockchockhancookie/Search_sys.git

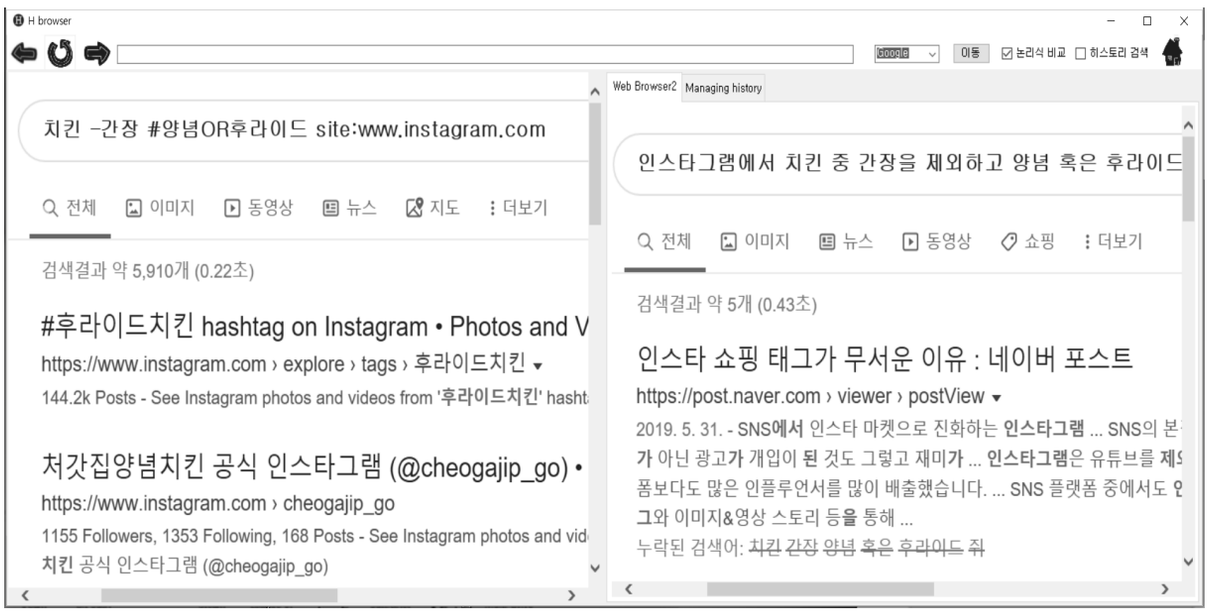


Fig. 6. H Browser : A web Browser for Logical Transformation of Natural Language

바와 같이 본 논문에서 제안한 방법을 이용하면 누락된 검색어가 나타나지 않는데, 이것은 논리식으로 변환하여 고급검색을 이용할 경우 누락된 키워드가 없이 사용자가 원하는 모든 키워드가 반영된 충실한 검색 결과를 얻을 수 있음을 의미한다.

Fig. 6에서 보는 바와 같이 자연어 상태의 질의로 검색을 하는 경우 키워드의 누락이 많이 일어나는 것을 알 수 있다. 이는 자연어 상태의 검색을 할 경우 모든 키워드들이 OR 처리되어 검색이 되기 때문이다. 이렇게 검색할 경우 구글의 랭크 알고리즘에 의해 하나의 키워드들이 포함된 모든 검색결과 중 가장 연관성이 높다고 판단되는 문서의 점수가 높아 그 결과를 상단에 제공하기 때문이다.

자연어를 논리식으로 변한하여 질의하는 것을 포함하여 본 논문에서는 히스토리 검색방법도 제공한다. H 브라우저는 히스토리 검색을 더욱 편리하게 사용할 수 있도록 도와주기 위하여 히스토리 편집기를 제공한다.


Fig. 7. History Editor

Fig. 7은 히스토리 편집기로서 사용자들이 이전에 검색한 질의들을 보여주는 화면이다. 사용자의 단순 질의를 리스트에서 보여주고 있는데, 이 질의는 내부적으로 이전에 질의와 결합되어 새로운 질의가 만들어진 후 논리식으로 변환하여 질의하게 된다. H 브라우저에서는 이를 더욱 편리하게 사용하기 위하여 누적된 질의를 편집할 수 있는 기능을 제공한다.

## 6. 번역의 정확도 검증

본 연구에서는 자연어를 논리식으로 변환하는 것이 검색결과의 질을 어느 정도 향상시키는지 검증하기 위하여 검색결과의 정확도를 분석하였고, 이와 더불어 검색 키워드의 누락을 측정하였다. 본 논문에서는 정확도를 측정하기 위하여 검색결과 중 상위 60개의[5] 결과 중 질의와 일치하는 검색결과와 그렇지 않은 검색 결과의 개수를 측정하고 그 비율을 계산하였다. 예를 들어 Fig. 1의 질의 1을 검색하였을 때 검색 결과는 인스타그램의 게시물인지, 검색 내용에 치킨이 있는지, 간장이라는 키워드는 반영이 되지 않은 것인지, 후라이드나 양념이 해시태그 처리되었는지를 전부 확인하여 모두 적합하다면 일치하다고 판단하였다. 또한 누락된 키워드의 비율을 검증에 사용한 이유는 사용자가 검색을 했을 때 그 키워드들을 중요하다고 생각하여 검색한 것이기 때문이다. 누락이 되었다는 것은 사용자가 그만큼 원하는 검색 키워드를 반영하지 않는다는 의미이기 때문에 검증의 지표로 선택하였다.

5) 검색을 하는 경우 일반적으로 목적에 맞는 게시물일수록 상단에 게재된다. 이는 구글이 게시물들에 각각 랭크를 매겨 검색한 것과 관련이 있다고 판단되는 것을 상위에 제공하기 때문이다. 따라서 본 논문에서는 3페이지에 게재되는 60개의 검색결과가 넘어가는 검색결과는 의미 없다고 평가하였다.

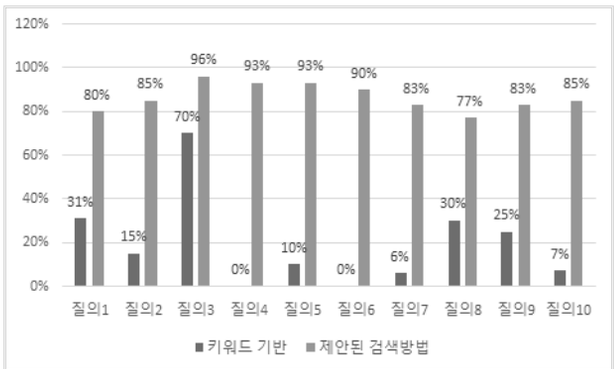


Fig. 8. A Comparison of the Correctness of Natural Language and Logical Expression

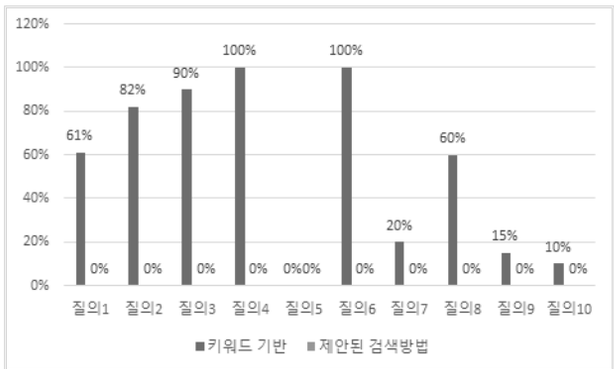


Fig. 9. Percentage of Missing Keyswords in Natural Language and Llogical Expression

Fig. 8은 Fig. 1의 자연어 질의와 Fig. 4의 논리식으로 번역된 질의의 검색 결과의 정확도를 비교한 것이고, Fig. 9는 누락된 검색어의 비율을 비교한 것이다. 본 실험에서는 질의에 나타난 키워드가 하나라도 누락되면 키워드가 누락된 결과로 판단하였다.

Fig. 8에서 자연어로 질의하였을 때 질의 1, 질의 2, 질의 3의 정확도는 각각 31%, 15%, 70%였다. 이 질의들을 논리식으로 변환하여 질의하였을 때의 정확도는 80%, 85%, 96%로 상승하는 것을 알 수 있다. 나머지 7개 질의들 또한 (Fig. 8)에서 나타나듯이 정확도가 상승한 것을 볼 수 있다. 이를 통해 자연어 질의를 논리식으로 변환하였을 때 결과의 질이 향상됨을 알 수 있다.

Fig. 9에서 자연어로 질의하였을 때 질의 1, 질의 2, 질의 3에서 누락된 키워드의 비율은 각각 61%, 82%, 90%였다. 이를 논리식으로 변환하여 질의하면 누락된 키워드의 비율은 모두 0%였다. 다른 질의들 역시 본 검색방법을 이용할 때 누락되는 것이 없다는 것을 알 수 있다. 예외로 질의 5의 경우 자연어 질의를 하였지만 키워드 누락은 일어나지 않았다. 하지만 키워드가 누락되지 않더라도 논리식으로 변환할 때와 정확도의 차이가 현저하게 나는 것을 확인할 수 있다. 이는 누락되지 않은 키워드들 중에 사용자가 원하지 않는 키워드가 반영되어 나타난 결과이다.

이렇게 실험을 통해 자연어 질의를 논리식으로 변환하게 되면 질의에서 원하는 키워드가 거의 누락되지 않는 것을 확인 할 수 있었다. 이러한 결과는 질의를 논리식으로 변환하게 되면 검색엔진에서 제안하는 검색 방법에 맞추어 검색을 하게 되고 그 결과 모든 키워드들을 반영하게 된다. 그리고 이는 키워드 누락비율의 저하로 나타난다. 키워드가 누락되지 않고 논리식에 정확하게 반영되고 필요없는 키워드는 제거하기 때문에 검색 결과 또한 변환하지 않은 질의보다 더 정확도가 증가한다. 따라서 본 논문에서 제안하는 방법대로 자연어 질의를 논리식으로 변환하여 검색한다면 그렇지 않은 경우보다 더욱 정확한 검색결과를 얻을 수 있다는 것을 알 수 있다.

또한 본 논문에서는 사용자의 편의성을 증대시키기 위해 히스토리 검색을 제안하였다. 4절에서 언급한 단순성에 의해 사람들은 검색을 할 때 일반적이고 특정성이 결여된 검색을 하기 쉽기 때문에 모호한 검색 결과를 얻는 경우가 빈번하다. 따라서 이러한 검색은 사용자의 정보요구를 정확히 표현하기에는 한계가 있다. 이러한 단점을 보완하기 위해 본 논문에서는 단순한 질의를 여러번 검색하면 이를 누적하여 보다 구체적인 질의를 자동으로 생성하고 검색하는 히스토리 검색을 제안한다. Fig. 10은 Fig. 1의 질의 1을 예시로 검색하여 히스토리 검색을 하는 과정을 나타낸 것이다.

[첫번째 질의]
사용자가 검색한 내용: 인스타그램에서 치킨 중
변환된 논리식 : site:www.instagram.com 치킨

[두번째 질의]
사용자가 검색한 내용: 간장을 제외하고
변환된 논리식 : site:www.instagram.com 치킨 -간장

[세번째 질의]
사용자가 검색한 내용: 양념 혹은 후라이드는 해시태그
변환된 논리식: site:www.instagram.com 치킨 -간장
#후라이드 OR 양념

Fig. 10. Process for Using History Search System

Fig. 10은 Fig. 1의 질의 1을 Fig. 7에서 보는 바와 같이 짧게 세 번에 걸쳐 검색하였을 경우, 이러한 질의가 히스토리 검색을 통해 누적되고 논리식으로 변환 되는 것을 나타낸 것이다. 세 번에 걸쳐 검색을 진행함에 따라 기존의 질의가 누적되어 새로운 질의와 합쳐지고 이는 매번 논리식으로 다시 변환되어 질의한다. 즉 기존의 질의를 유지되면서 단순 질의

가 합쳐지면서 논리식으로 재구성 되는 것이다. 이는 질의 결과 내에서 다시 재검색을 하는 것과 유사한 의미이다. 사용자는 번거롭게 마우스를 이용해 검색 엔진의 결과 내 재 검색란을 찾거나 한번에 긴 문장을 완성시켜 검색할 필요없이 자연어 질의를 여러번 하는 것으로 정교한 검색을 완성시켜 나갈 수 있다는 것이다.

## 7. 결 론

본 논문에서는 형태소 분석기를 활용하여 자연어 질의를 검색엔진에 맞는 논리식으로 변환하는 검색방법과 이를 활용하여 짧은 여러 번의 질의를 검색 했을 때 질의들을 누적시켜 구체적인 최종질의로 검색하게 해주는 히스토리 검색방법을 제안하였다. 그리고 제안한 검색방법들을 구현하여 그 성능을 비교하였다. 그 결과 자연어 형태의 질의를 검색엔진의 고급검색에 맞는 논리식으로 변환하여 검색하였을 경우, 결과의 정확도가 개선되는 것을 확인할 수 있었고 누락되는 키워드 없이 사용자의 질의를 검색 결과에 100% 반영하여 더욱 정확한 결과를 얻을 수 있음을 확인할 수 있었다. 또한 검색 사용자들의 특성 중 하나인 단순성을 개선하기 위해 제안된 히스토리 검색기법을 통해 사용자가 검색을 좀 더 쉽게 사용할 수 있음을 확인하였다. 따라서 본 연구에서 제안하는 자연어를 고급검색기반의 논리식으로 변환하는 검색방법과 히스토리 검색방법을 이용할 경우 검색을 하는 이용자들의 편의성이 증대될 것이다.

## References


[1] Ministry of Science and ICT(2019). 2018 Survey on Internet Use.

[2] Total Number of Website [Internet], http://www.internetlivestats. com/total-number-of-websites/.

[3] In K. Lee, Seo H. Son, and Soon H. Kwon, "Knowledge-based semantic meta-search engine," *Journal of Korean Institute of Systems*, Vol.14, No.6, pp.737-744, 2004.

[4] 김준태, 윤견아 "인터넷 정보검색 시스템의 연구 동향," *The Korean Institute of Electrical Engineers*, Vol.48, No.3, pp.52-59, 1999.

[5] Young-Duk Seo, Jeong-Dong Kim, Chonghyeon Lee, and Doo-Kwon Baik, "A page rank algorithm for information retrieval in real time," *Proceedings of KIISE Fall Conference*, Vol.38, No.2C, pp.57-60, 2011.

[6] Wikipedia, "Google Fred" [Internet], https://en.wikipedia.org/wiki/ Google_Fred.

[7] Dong Kwon Kim, "Tips for using Google search," *Magazine of the SAREK*, Vol.44, No.2, pp.80-81, 2015.

[8] Sung-Hee Lee, "Syntactic analysis and keyword expansion for performance enhancement of information retrieval system," *Korea Academy Industrial Cooperation Society*, pp.139-142, 2004.

[9] Sung Hee Yoon, "Personalized web search using query based user profile," *Journal of the Korea Academia-Industrial Cooperation Society*, Vol.17, No.2, pp.690-696, 2016.

[10] Myung-Seok Yang, Seok-Hyung Lee, Nam-Kyu Kang, and Hwa-Mook Yoon, "A ranking method using link & description information in meta searching," *Proceedings of KIISE Fall Conference*, Vol.29, No.2, pp.118-120, 2002.

[11] 김성희, "인터넷상의 메타검색엔진 검색효율성에 관한 비교연구," *Korean Library and Information Society Summer Conference*, pp.75-91, 1997.

[12] Yong-Woon Han, "Applying korean linguistics to natural language question-answer search system," *International Language and Literature*, Vol.10-1, pp.36-51, 2004.

[13] Myeong Hee Lee, "An exploratory study of performances between a subject directory and keyword search engine in the network databases," *Journal of The Korean Society for Library and Information Science*, Vol.31, No.2, pp.177-197, 1997.

[14] Yong Kim and Ju Won Kyun, "Design and implementation of tag coupling-based boolean query matching system for ranked search result," *Korea Society for Information Management*, Vol.29, No.4, pp.101-121, 2012.

[15] Ryen W. White and Dan Morris, "Investigating the querying and browsing behavior of advanced search engine users," *SIGIR '07 Proceedings of the 30th Annual International ACM SIGIR Conference*, pp.255-262, 2007.

[16] Daehong Lee, Hansuk Yu, and Sangwon Park, "Searching system using advanced search of search engine," *Korea Software Congress 2019*.



**이 대 홍**

https://orcid.org/0000-0003-2950-2971
e-mail : mellow3632@naver.com
2020년 한국외국어대학교 정보통신공학과 (학사)
관심분야 : 시스템 프로그래밍, 영상처리, 기계학습, 웹서비스

유 한 석

https://orcid.org/0000-0003-0514-3740
e-mail : yusw10@naver.com
2020년 한국외국어대학교 정보통신공학과 학사
관심분야 : 시스템 프로그래밍, 증강현실, 웹서비스

박 상 원

https://orcid.org/0000-0003-0384-0964
e-mail : swpark@hufs.ac.kr
1994년 서울대학교 컴퓨터공학과(학사)
1997년 서울대학교 컴퓨터공학과(석사)
2002년 서울대학교 컴퓨터공학과(박사)
2002년 세종사이버대학교 전임강사
2003년 ~ 현 재 한국외국어대학교 정보통신공학과 교수
관심분야 : 데이터베이스, 플래시메모리, XML, 웹서비스, 모바일 컴퓨팅