

# 후위 표기법을 사용한 수학적 색인 및 랭킹<sup>1)</sup>

이세희<sup>o</sup>, 신준수, 김학수  
강원대학교 컴퓨터정보통신공학전공  
{nlpslee, nlpsjs, nlpdrkim}@kangwon.ac.kr

## Indexing and Ranking Mathematical Equations Using Postfix Notation

Sehee Lee<sup>o</sup>, Junsoo Shin, Harksoo Kim  
Program of Computer and Communications Engineering, Kangwon National University

### 요 약

최근 인터넷 및 컴퓨터의 사용이 활발해짐에 따라 문서의 디지털화가 빠르게 진행되고 있다. 이런 변화에 따라 수학적 식이 많이 사용되는 과학, 공학, 수학 등의 분야와 관련된 문서들을 검색해야 할 필요성이 늘고 있다. 그러나 현재 일반 검색엔진은 텍스트 검색만을 제공하며 별도의 수학적 검색은 제공하지 않는다. 따라서 본 논문에서는 수학적 검색이 가능하도록 수학적 식의 색인 방법 및 랭킹 방법을 제안한다. 제안하는 색인 방법은 MathML로 입력되는 수학적 식을 후위 표기법과 일반 색인 방법의 두 가지로 색인하며, 언어모델을 사용하여 질의에 적합한 수학적 식을 랭킹한다. 일반 검색 엔진의 성능과 비교하기 위하여 2-포아송 모델과 제안 모델을 비교하였으며, 그 결과 제안 모델의 성능이 더 우수함을 보였다.

주제어: 수학적 검색, 수학적 색인, 수학적 랭킹, 후위 표기

### 1. 서론

과학, 공학, 수학 등 전문 분야의 많은 문서들이 디지털화되어 출판되며 이런 종류의 문서에는 다양한 수학적 식이 포함되어 있다. 또한 웹에 존재하는 많은 문서들 역시 수학적 식이 포함되어 있다. 그러나 현재 사용되는 일반 검색엔진에서 별도의 수학적 검색은 제공하지 않는다. 그러므로 사용자가 특정 수학적 식을 검색할 목적으로 검색을 시도하여도 일반 텍스트 검색 방법에 의존하여 문서를 검색하기 때문에 사용자가 원하는 검색 결과를 얻지 못하는 경우가 많다. 일반 검색 엔진에서 수학적 검색을 제공하기 어려운 이유는 색인하기 위한 수학적 식을 텍스트로 변경하기 어렵고 수학적 색인 및 랭킹에 관한 적합한 방법에 대한 연구가 부족하기 때문이다. 수학적 식을 텍스트로 변경하기 어려운 이유는 디지털 문서들과 웹에 존재하는 대부분의 수학적 식이 이미지로 저장되어 있기 때문이다. 따라서 이미지가 수학적 식을 나타내는 이미지인지 아닌지 판단해야 하며, 수학적 식을 나타내는 이미지라면 컴퓨터가 처리할 수 있도록 해당 이미지의 수학적 식과 동일한 의미의 텍스트로 변환하는 작업이 필요하다. 그리고 다양한 문서 포맷은 수학적 식을 표현하기 위해 각자 고유한 표현방법을 사용하므로 수학적 색인을 더욱 어렵게 한다. 현재 수학적 식을 표현하기 위해 TeX, MathML 등이 사용되고 있지만 많은 사람들이 그 존재를 알지 못하며 문서 작성시 또는 웹에서의 사용이 편리하게 제공되지

않는다. 이러한 문제를 해결하기 위해 한국정보통신학회에서 수식의 표시 및 의미 정보 표현 지침에 관한 표준[1]을 제시하였지만 가이드라인 수준에 머물고 있다.

일반 문서 검색과는 다르게 수학적 검색에서 고려해야 될 사항들은 크게 3가지가 있다.

1)질의 입력 방법: 사용자가 검색하려는 수학적 식을 입력하도록 하는 질의 입력 인터페이스를 어떤 방식으로 제공할지 결정해야 한다. 일반 검색 엔진에서 제공하는 방법과 같이 텍스트로 입력을 받거나[2]의 방법과 같이 검색하고자 하는 수학적 식을 수학적 입력 인터페이스를 사용하여 입력받을 수 있다. 또한 앞의 두 방법을 합쳐서 사용할 수도 있다. 그리고 사용자가 입력한 검색어에 색인에 사용된 알고리즘과 동일한 방법을 사용하여 색인어를 생성해야 한다.

2)수학적 검색 방법: 수학적 검색의 핵심부분으로 질의와 유사한 수학적 식이 잘 검색되도록 하는 것이 목적이다. 수학적 식의 색인을 위해서 문서에서 수학적 식을 구별하는 방법이 있어야 한다. 그리고 수학적 식에서 색인어를 생성해야 하며, 사용자가 입력한 질의와 색인된 수학적 식간의 유사도를 계산하여 순위를 계산하는 방법이 필요하다.

3)결과 출력 방법: 사용자가 원하는 결과를 쉽고 빠르게 찾을 수 있도록 검색된 결과를 잘 보여 주는 것이 목적이다. 일반 검색 엔진은 사용자가 입력한 질의와 일치하는 부분을 강조하여 보여주지만 수학적 식은 이미지파일

1) 이 논문은 2008년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(KRF-2008-313-D00907).

문서의 객체, MathML과 같이 다양한 형태로 존재하기 때문에 텍스트 검색과 다르게 다루어져야 한다

수학식 검색 시스템에서 우선적으로 해결해야 할 문제는 색인어의 결정이다. 수학식은  $(4x+a)^{2x+y}$ 와 같이 수학적 특성에 의해 구조화되어 있으며 이렇게 구조화된 수학적 식을 인식하고 수학적 특성에 맞는 색인어를 생성해야 한다. 한국어를 대상으로 하는 문서 검색은 색인어 단위로 형태소를 사용하지만 수학식 검색의 경우 색인어를 결정하는 방법에 따라 동일한 수학식에 대하여 서로 다른 여러 개의 색인어를 생성할 수 있다. 또한 수학식 검색에서 색인어의 가중치는 sin, cos과 같은 함수명,  $\Sigma$ 와 같은 기호, 변수, 상수 등에 모두 다르게 적용할 수 있으며, 수학식의 구조적 특성 때문에 검색 결과의 랭킹 방법도 일반 문서 검색 방법과 차이가 있어야 한다 따라서 수학식 검색에 적합한 새로운 색인 및 랭킹 방법이 필요하다. 본 논문은 검색 대상이 되는 수학식이 MathML형태로 인식된 상태라고 가정하고 수학식 검색에 적합한 색인 생성 및 랭킹 방법을 제안한다

본 논문의 구성은 다음과 같다 2장에서 수학식 검색에 관한 연구들을 살펴보고, 3장에서 제안하는 수학식 색인 시스템에 대해 설명하고, 4장에서 실험 결과를 설명하고, 마지막으로 5장에서 결론을 맺는다.

## 2. 관련 연구

수학식 검색에 관해 한국에서 수행된 연구는 없는 것으로 생각되며 해외에서 진행된 연구를 중심으로 관련 연구를 살펴본다. 최근의 연구로 GUI기반의 질의 입력 인터페이스에 관한 연구가 있었다[2]. 이 연구는 수학식 검색 질의를 GUI 인터페이스를 통해 입력받고 질의를 MathML로 변환하도록 하였다. [2]와는 반대로 검색 질의를 텍스트로만 입력 받을 수 있도록 한 연구도 있었다 [3]. [3]은 '+', '-'와 같은 기호에 'plus', 'minus'와 같은 단어를 할당함으로써 텍스트 질의로 검색이 되도록 하였다. 정규 표현을 사용하여 색인어를 생성하는 연구도 있었다[4]. [4]는 MathML태그로 구성된 정규표현에 키워드를 할당하고, 처리할 수학식에 정규표현 매칭을 시도하여 해당 수학식에서 키워드를 추출하였다. 색인 대상 수학식을 확장하여 여러 색인어를 생성하는 연구도 있었다[5]. [5]는 특정 수학식과 동일한 의미를 가지는 N개의 수학식을 생성하고 후위표기를 사용하여 색인어를 추출하였다. 색인어의 출현 빈도보다 색인어 자체의 중요도를 이용한 연구도 있었다[6]. [6]은 'sin', 'tan'과 같은 함수명에 가중치를 가장 많이 할당하고 '+', '/'와 같은 연산자는 함수명보다 조금 더 작은 가중치를 할당

하고, 변수명과 숫자에는 가장 낮은 가중치를 할당하는 방법을 사용하였다. [6]에서 제안한 방법은 DLMF 프로젝트[7]에 적합한 모델로써 잘 구조화된 웹페이지에 적용할 수 있도록 설계되었다.

기존의 수학식 검색에 관련된 연구에서 색인 및 랭킹 방법은 수학적 구조를 잘 반영하지 못하거나 특정 웹사이트에 대한 검색을 목적으로 하였다. 그리고 수학식 검색을 위해 만든 검색 모델에 대한 평가가 부족한 것이 단점이었다. 따라서 본 논문에서는 수학적 특성을 반영하며 범용으로 사용이 가능하도록 색인어를 생성하고 랭킹 방법을 제안하며, 수학식 검색 결과를 일반 검색 방법과 비교 한다.

## 3. 수학식 색인 및 랭킹 모델

수학식 검색은 사용자가 질의한 수학식과 완전히 일치하는 수학식뿐만 아니라 부분적으로 일치하는 수학식 및 유사한 수학식까지도 보여줄 수 있어야 한다. 본 논문은 부분적으로 일치하는 수학식이나 질의와 유사한 수학식은 질의와 완전히 일치하는 수학식보다 낮은 랭킹의 결과를 보여야 한다고 가정한다. 본 논문에서는 색인 대상이 되는 MathML 데이터를 사용하여 색인을 수행한다. 수학식 색인은 수학식 정규화, 후위표현 변환, 색인어 생성, 색인어 저장 단계를 거친다. 검색시에는 질의에서 색인 방법과 동일하게 색인어를 생성하고 수학식 랭킹 방법을 사용하여 랭킹된 결과를 보여준다.

### 3.1 수학식 색인

수학식의 정규화 단계는 후위표현을 사용한 색인어 생성 단계에서 동일한 의미의 서로 다른 수학식이 검색되도록 한다. 정규화는 입력되는 수학식을 높은 차수, 변수의 알파벳, 상수를 기준으로 정렬한다. 예를 들어  $a+2+b^2$ 과 같은 수학식이 있다면, 우선순위에 따라 가장 높은 차수인  $b^2$ 이 처음에 오고, 낮은 차수인  $a$ , 상수 2의 순서가 되어 최종적으로  $b^2+a+2$ 로 정규화가 이루어진다. 만일 정규화 과정을 거치지 않으면  $k+x$ 과  $x+k$ 와 같이 동일한 의미를 가진 수학식을 후위표현으로 변환하였을 경우 생성되는 색인어는 각각 'kx+' 과 'xk+' 가 되기 때문에 변수의 위치가 변경된 수학식은 검색할 수 없다.

정규화 단계를 거친 수학식은 후위표기로 변환한다. 후위표현은 복잡한 괄호표기 없이 구조적인 수학적 의미를 유지한 상태의 일차원적인 텍스트로 변환할 수 있는 것이 장점이다. 또한 후위표기를 사용함으로써 부분적으로 일치하는 수학식을 검색할 수 있다. 예를 들

어 수학적 식  $a+b+c$ 는 ‘ab+’, ‘c+’ 두 개의 색인어로 생성되며 수학적 식  $a+b$ 는 하나의 색인어 ‘ab+’로 생성된다. 사용자가  $a+b$ 를 질의로 입력하였을 경우  $a+b+c$ 도 검색 대상이 된다.

```

Input:
Token_vec - 후위표기 토큰 벡터
Buffer - 색인어 버퍼

Output:
Index_vec - 색인어 벡터

Process:
Buffer 초기화;
for all  $v_i$  in Token_vec
{
  if( $v_i$ ==연산자){//현재가 연산자
    if( $v_{i+1}$  != end){//현재가 끝이 아닐 경우만
      if( $v_{i+1}$  == (변수 or 숫자)){//다음이 변수, 숫자 경우
        Buffer +=  $v_i$ ;
        Index_vec += Buffer;
        Buffer 초기화;
      }
      else{//현재 다음이 연산자일 경우
        Buffer +=  $v_i$ ;
        Index_vec += Buffer;
      }
    }
  }
  else{//현재가 마지막 토큰
    Buffer +=  $v_i$ ;
    Index_vec += Buffer;
  }
}
else{//숫자, 변수
  Buffer +=  $v_i$ ;
  if( $v_{i+1}$  != end){
    if( Buffer.length != 0 )
      Index_vec += Buffer;
    Buffer 초기화;
  }
}
}

```

그림 1. 색인어 생성 방법 알고리즘

[그림 1]은 후위표기를 이용한 색인어의 생성 방법 알고리즘이다. 후위표기로 입력되는 토큰 리스트의 앞에서부터 뒤로 탐색해 나가면서 연산자를 만나기 전까지 출력 색인어 버퍼를 채운다. 연산자를 만났을 경우 뒤에

변수 또는 숫자가 오는 부분을 경계로 구분하여 현재까지의 색인어 버퍼를 색인어로 결정함으로써 하나의 색인어를 생성한다. 만일 연산자 바로 뒤에 또다시 연산자가 존재한다면 현재까지 저장된 색인어 버퍼를 색인어로 저장한 후, 버퍼를 비우지 않고 뒤에 존재하는 연산자를 추가하여 다시 하나의 색인어를 생성한다 이 과정을 토큰 리스트의 마지막까지 도달하거나 변수 또는 숫자를 만날 때까지 반복한다.

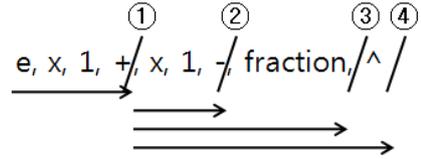


그림 2. 색인어 생성 방법 예

[그림 2]는 수학적 식  $e^{\frac{x+1}{x-1}}$ 을 후위표기로 변환하여 색인어를 생성하는 방법을 나타내며 화살표는 생성되는 색인어를 표시한다. [그림 2]에서 ‘fraction’은 분수를 나타내며, ‘^’는 지수를 나타내는 연산자이다. ①의 경우 색인어 토큰 생성 알고리즘에 의해 후위표기의 앞에서부터 뒤로 이동하면서 ‘+’ 연산자를 만나게 된 부분이며 ‘+’ 뒤에 변수  $x$ 가 존재하기 때문에 색인어 ‘ex1+’이 생성된다. ①의 뒤로 이동하면서 [그림 2]의 ②의 위치에서 연산자 ‘-’를 만나 색인어 ‘x1-’가 생성되지만 ‘-’ 뒤에 연산자 ‘fraction’이 존재하기 때문에 색인어 버퍼에 ‘fraction’을 추가하여 위치 ③까지 색인어 ‘x1-fraction’을 생성하고, 다시 뒤에 연산자 ‘^’이 존재하므로 ④까지 색인어 ‘x1-fraction^’을 생성한다. ‘^’는 토큰 리스트의 마지막이므로 색인어 생성 알고리즘을 종료한다. 색인어 생성 결과 ‘ex1+’, ‘x1-’, ‘x1-fraction’, ‘x1-fraction^’의 총 4개의 색인어가 생성된다.

후위표기법과 동시에 함수 패턴 사전을 사용한다 함수 패턴사전은 미리 정의된 특수한 수학기호 또는 함수의 지정된 위치에 있는 인자를 색인할 수 있도록 한다. 예를 들어  $\sum_{i=1}^n$  이와 같은 수학적 식이 함수 패턴 사전에 등록되어 있다면 색인어는 ‘ $\Sigma$ ’, ‘sigma\_start\_i=1’, ‘sigma\_end\_n’과 같이 3개가 생성되며,  $\log_e a+x+4$ 과 같은 수학적 식은 ‘log’, ‘log\_base\_e’, ‘log\_number\_ax+’, ‘log\_number\_4+’와 같이 4개의 색인어가 생성된다.

후위표기만을 사용하여 색인하면 유사한 수학적 식에 대한 검색이 안 되는 경우가 발생하여 재현율(recall)이 낮아진다. 예를 들어 질의로  $e$ 를 입력하면, 후위표기만을

사용하여  $e^{\frac{x+1}{x-1}}$ 를 색인한 경우에는 검색되지 않는다. 이런 문제를 해결하기 위해서 후위표기뿐만 아니라 기존의 정보 검색 방법과 같이 각각의 변수, 함수, 숫자, 연산자 토큰들을 함께 색인하여 재현율을 높인다. 일반적인 색인어 생성 방법을 사용하면  $e^{\frac{x+1}{x-1}}$ 의 색인어 리스트는 e, x, 1, +, x, 1, -, fraction, ^의 9개이다. 이와 같이 색인어 생성 과정을 거친 후 저장되는 색인어는 후위표기로 생성되는 색인어와 일반 색인어의 두 그룹으로 구분한다

### 3.2 수학적 랭킹

입력 질의와 색인된 수학적식의 유사도 계산을 통한 수학적식의 랭킹은 색인어 결정 이상으로 매우 중요하다. 본 논문에서는 언어모델을 사용하여 질의와 수학적식의 유사도를 계산한다. 정보검색의 랭킹에 사용되는 언어모델은 특정 문서에서 입력 질의가 생성될 확률을 계산하며 본 논문은 특정 수학적식에서 입력되는 수학적 질의가 생성되는 확률을 식 (2)를 통해 모델링한다.

특정 수학적식에서 질의가 생성될 확률은 입력 질의에 존재하는 토큰이 해당 수학적식에서 출현할 확률의 곱을 사용하여 식 (1)과 같이 계산할 수 있다. 식 (1)에서 Q는 입력 질의, Equation은 수학적식, l은 입력 질의의 수,  $q_i$ 는 입력 질의의 i번째 토큰을 각각 의미한다.

$$P(Q|Equation) = \prod_{i=1}^l P(q_i|Equation) \quad \text{식 (1)}$$

색인어는 후위표기와 일반적인 색인어 두 그룹으로 생성하였으며, 후위표기를 사용한 방법이 수학적 의미 정보를 더 많이 포함한다. 따라서  $\alpha$ 값을 사용하여 두 색인어 그룹에 서로 다른 가중치를 할당한다. 이를 식 (1)에 적용하여 풀면 식 (2)와 같다.

$$\begin{aligned} P(Q|Equation) &= \prod_{i=1}^l P(q_i|Equation) && \text{식 (2)} \\ &\approx \alpha \left\{ \prod_{i=1}^m P(pq_i|Equation) \right\} + (1-\alpha) \left\{ \prod_{i=1}^n P(gq_i|Equation) \right\} \\ &\approx \alpha \left\{ \prod_{i=1}^m \frac{tw(pq_i, Equation)}{\sum_{j=1}^m tw(pq_j, Equation)} \right\} + (1-\alpha) \left\{ \prod_{i=1}^n \frac{tw(gq_i, Equation)}{\sum_{j=1}^n tw(gq_j, Equation)} \right\} \end{aligned}$$

where  $tw = 1.0 + tf \cdot idf$

특정 수학적식에서 질의의 가중치는  $tf \cdot idf$ 를 사용하여 계산한다. 식 (2)에서 m은 후위표기를 사용한 질의어 토큰의 수, n은 일반적인 방법으로 생성한 질의어 토큰의 수이다.  $pq_i$ 는 후위표기로 생성된 i번째 질의어 토큰이며,  $gq_i$ 는 일반방법으로 생성된 i번째 질의어 토큰이다.

### 4. 실험 및 결과

실험은 고등학교 수학 문제집의 17개 단원에 존재하는 수학적식 430개를 수집하여 수행하였다. 수학적식 검색은 일반적인 정보 검색과 달리 알려진 평가 집합이 없으며 표준화된 평가 방법도 없다. 제안 검색 모델은 후위표기 색인 방법과 일반 색인 방법을 합쳐서 사용하기 때문에 입력 질의에 포함된 기호들은 모두 검색되어 재현율은 100%가 된다. 예를 들어 ‘+’를 입력 질의로 사용하면 두 질의 ‘a+b’, ‘ $\frac{x+1}{x-1}$ ’와 같이 ‘+’가 들어간 수식은 모두 검색된다. 때문에 검색 결과의 재현율보다는 정확률(precision)이 더 의미가 있으며, 사용자는 주로 검색된 상위 결과만을 이용하는 경향이 있으므로 검색 결과 상위 10개의 정확률을 측정하였다.

본 논문은 질의에 적합한 수학적식이 검색되었는지 평가하기 위하여 이미 색인된 수학적식에서 무작위로 선택한 5개의 수학적식을 검색하여 동일한 수학적식이 1위에 랭크되는지 확인하였다. 또한 질의에 의해 검색된 1~10위 결과 가운데 사람의 판단으로 질의에 적합한 수학적식을 선택하여 정확률을 측정하였다. 그리고 제안 모델과 일반적인 검색 방법을 비교하기 위하여 식 (3)의 2-포아송 모델[8]을 사용하여 검색 결과를 비교하였다.

$$\begin{aligned} Sim(D, Q) &= \sum_{i=1}^n (w_{di} \times w_{qi}) && \text{식 (3)} \\ &= \sum_{i=1}^n \left\{ \frac{tf_{di}}{k_1 \left( (1-b) + b \cdot \frac{doc.length}{average.doc.length} \right) + tf_{di}} \right. \\ &\quad \left. \times \log \frac{N - df_i + 0.5}{df_i + 0.5} \times tf_{qi} \right\} \end{aligned}$$

where  $k_1 = 2, b = 0.75$

정확률 측정은 5명의 대학원생에게 질의와 검색 결과를 보여주고 적합한 검색 결과를 선택하도록 하여 검색 결과 상위 10개의 정확률의 평균으로 계산하였다. 제안 모델의  $\alpha$ 값은 0.7을 사용하였으며, 제안 모델과 비교하기 위한 2-포아송 모델을 사용한 수학적식 검색에는 일반적인 색인어 생성 방법을 이용하였다.

표 1. 성능 비교에 사용된 입력 질의

번호	질의 수학적식
1	$\log_a \sqrt{a} - \log_a a^2 - \frac{1}{2} \log_a \frac{1}{a}$
2	$n^2 - 1$
3	$\sum_{k=1}^3 (k+1)$
4	$\lim_{n \rightarrow \infty} \frac{\sqrt{5n^2 + n - 1}}{\sqrt{n^2 + 1}}$
5	$(\sqrt{2})^3, \sqrt[3]{16}$

[표 1]은 성능 비교에 사용된 질의이며, [표 2]에는 제안 모델과 2-포아송 모델의 검색 결과를 비교하였다

표 2. 제안 방법과 2-포아송 모델 비교

입력 질의	precision at 10(%)		정답의 1위 랭킹	
	제안방법	2-포아송	제안방법	2-포아송
1	38.0	16.0	O	O
2	24.0	16.0	O	X
3	52.0	46.0	O	O
4	40.0	44.0	O	X
5	32.0	20.0	O	O

본 논문은 검색 결과의 비교를 위해 이미 색인된 수학적식 5개를 무작위로 선택하여 검색하였기 때문에 정답을 알 수 있다. 따라서 정답이 1위에 랭크되었는지 여부를 [표 1]에 O, X로 표시하였다. 제안 방법의 경우 정답 5개가 모두 1위에 랭크되었지만, 2-포아송 모델을 이용한 일반 검색 방법은 정답 3개만이 1위에 랭크되는 결과를 볼 수 있다. 검색 결과 상위 10개 중 적합한 수학적식의 비율을 측정한 결과 제안 방법이 평균 37.2%의 정확률을 보여 2-포아송 모델의 평균 28.4%보다 8.8% 더 좋은 결과를 보인다. 그러나 이렇게 수학적식 검색의 정확률을 측정하는 방법은 결과의 적합성을 판단하는 사람에 따라 개인차가 발생하기 때문에 절대적인 기준이 될 수는 없다.

## 5. 결론 및 향후 과제

본 논문은 수학적식 검색에 적합한 색인 방법과 랭킹 방법을 제안하였다. 실험을 통해 제안 방법의 성능을 2-포아송 모델과 비교하였으며, 그 결과 제안 방법이 더 우수함을 보였다. 그러나 현재 제안 색인 방법의 한계로 행렬, 선분, 집합 등의 수학적식에 대해서는 후위표현을 사용한 색인을 적용할 수 없었다.

향후 연구는 다음과 같다. 현재의 랭킹 모델은 특수한 함수에 별도의 가중치를 적용하지 않았는데, 특수한 함수 또는 연산자에 가중치를 더 적용하는 방법에 대한 연구가 필요하다. 그리고 공기정보를 사용한 색인어 가중

치와 수학적식에서 의미 있는 자질과 주변 문맥정보를 사용하는 방법을 연구할 필요가 있다 또한 수학적식의 모양이 유사해도 검색이 가능하도록 하는 방법과 텍스트 질의에도 검색이 가능하도록 하여 기존 텍스트 검색 엔진에 수학적식 검색 엔진을 쉽게 추가할 수 있는 방법에 대한 연구가 필요하다. 마지막으로 수학적식 검색의 성능을 측정하는 합리적인 방법의 개발이 필요하며 공개된 수학적식 검색 데이터용 문서 집합이 필요하다

## 참고 문헌

- [1] 수학 정보 검색 - 수식의 표시 및 의미 정보 표현 지침, 표준번호 TTAS.KO-10.0138, 한국정보통신기술협회, 2002.
- [2] Grzegorz Bancerek, Michael Kohlhase, MathFind: a math-aware search engine, SIGIR 2006, pp. 735-735, 2006.
- [3] Abdou S. Youssef, Search of Mathematical Contents: Issues And Methods, IASSE, pp. 100-105, 2005.
- [4] Muhammad Adeel, Hui Siu Cheung, Sikandar Hayat Khiyal, MATH GO! PROTOTYPE OF A CONTENT BASED MATHEMATICAL FORMULA SEARCH ENGINE, JATIT, Vol4, No10, pp. 1002-1012, 2008.
- [5] Jozef Misutka, Leo Galambos, Extending Full Text Search Engine for Mathematical Content, Towards Digital Mathematics Library(DML): DML2008 workshop, pp. 55-67, 2008.
- [6] Abdou S. Youssef, Relevance Ranking and Hit Description in Math Search, Mathematics in Computer Science, pp. 333-353, 2008.
- [7] <http://dmlf.nist.gov/>
- [8] 김지승, 이준호, 이상호, 세 가지 정보 검색 모델의 성능 평가 및 분석, 정보과학회논문지: 데이터베이스, 제 28권, 제2호, pp. 266-278, 2001.