

# 온라인 필기체 수식 인식에서의 순차적인 구조 분석

이도화<sup>1)</sup>, 정선화, 김수형  
전남대학교 전산학과

## Sequential Structure Analysis in On-line Handwritten Formulas Recognition

Do-Hwa Lee, Seon-Hwa Jeong, Soo-Hyung Kim  
Dept. of Computer Science, Chonnam National University

### 요 약

본 논문에서는 온라인 필기체 수식 인식을 위한 순차적인 구조 분석 방법을 제안한다. 제안된 방법은 태블릿 상에서 필기된 수식에 대한 심볼 인식 결과와 각 심볼의 Bounding Box의 좌표를 입력받아서 필기 순서를 기반으로 순차적으로 수식의 구조를 해석한다. 그래프 내의 이웃하는 두 노드 사이의 관계를 결정하기 위해서 심볼의 사용에 관한 표기 정보와 6 단계 관계 결정 규칙을 사용하여 노드들 사이에 생성될 수 있는 에지의 수를 최소화하고 BackTracking을 피했다. 제안 방법의 성능을 평가하기 위해 100개의 테스트 샘플에 대해 구조 분석 실험을 수행하였다.

### 1. 서 론

자연과학분야에서 사용되고 있는 많은 문서들은 다양한 수식을 포함하고 있지만 컴퓨터에 수식을 입력하는 것은 쉽지 않다. 특히 복잡한 수식의 경우 키보드보다는 필기 인터페이스가 적당할 것이다. 수식의 인식은 심볼 인식과 구조 분석이라는 두 개의 프로세스로 나누어진다. 심볼 인식은 의미를 해석하지 않고 단순히 개개의 심볼을 인식하는 것이다. 구조 분석은 심볼의 상대적인 크기와 위치 정보를 가지고 인식된 심볼의 의미를 해석하는 것이다. 본 논문에서는 수식의 구조 분석에 대해 논한다.

기존 수식의 구조 분석에 관한 연구로는 구문론적인 방법, 투영을 이용한 방법, Graph Rewriting, 순차적으로 코드화된 수학 구문이 있다[1]. 구문론적인 방법은 이차원 문법을 기반으로 수학적인 표기 구조를 파악한다. Anderson[2]은 구문론상의 목표를 가지고 top-down 파싱을 심볼에 적용한 반면, Belaid[3]은 top-down 파서와 bottom-up 파서 모두를 사용했다. Chang[4]은 수식의 구조를 탐색하기 위해 구조 명세도를 제안했다. 투영을 이용한 방법은 다양한 문서 이미지 분석에 사용되었던 방법으로, 수식의 구조를 분석하기 위해 투영 정보를 이용한 분할을 시도한다. Okamoto[5]는 RPPC (Recursive Projection Profile Cutting)를 적용하여 부분적인 구조 분석을 시도했고, Faure와 Wang[6]은 투영 방법이 실패했을 경우에 대한 처리방법으로 mask-removal 연산을 정의했다. Graph Rewriting 방법은 수식 구조를 속성 그래프로 표현하고 Backtracking을 피하는 bottom-up 방식으로서, Grbavec[7]은 구조 분석 모델

로 'Build-Constrain-Rank-Incorporate'을 제시했다. 순차적으로 코드화된 수식 구문론은 투영을 통해 분할을 한후 심볼들을 그룹핑하는데 빠른 수행시간을 가지는 반면 공간적, 논리적 관계를 파악하기가 어렵다.

본 논문에서는 2차원 공간상에 쓰여진 수식으로부터 심볼단위 인식이 완벽하게 되었다고 가정하여 수식의 구조를 해석하는 순차적인 구조 분석 방법을 제안한다. 2장에서는 제안된 시스템 구조, 수학 함수명을 탐색하는 방법, 순차적인 구조 분석 방법을 소개하고 3장에서는 실험 및 결과에 대해 기술한다. 4장에서는 결론을 기술한다.

### 2. 제안 방법

제안된 구조 분석 방법은 그림 1에서와 같이 초기 그래프 생성, 이웃하는 두 노드 사이의 위치관계를 고려한 에지 생성, 혼란의 수식 형태의 변환 등 세 가지 모듈로 구성된다. 초기 그래프 생성 모듈은 입력 데이터로부터 수식의 구조를 표현할 수 있는 그래프를 출력한다. 이웃하는 두 노드 사이의 위치관계를 고려한 에지 생성 모듈은 수식의 필기순서에 근거하여 노드 사이에 생성 가능한 에지의 수를 줄이고 수학에서 사용되는 심볼의 표기 정보와 위치 정보를 이용하여 노드 사이의 관계를 결정한다. 혼란의 수식 형태의 변환 모듈에서는 노드가 표현하는 심볼과 에지의 정보를 가지고 수식이 표현하는 의미를 혼란의 수식 형태로 출력한다. 본 논문에서 수식은 왼쪽에서 오른쪽으로 필기된다고 가정하고 그 수식으로부터 얻어진 입력 데이터는 필기순으로 정렬된 심볼 인식결과 리스트, 심볼의 Bounding

Box 좌표로 이루어진다. 그리고 출력 데이터는 한글의 수식형태이다.

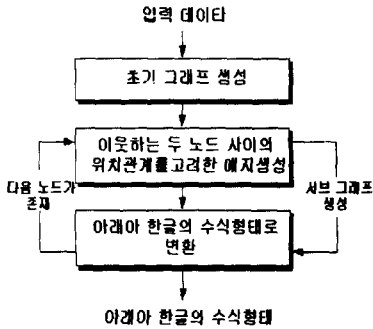


그림 1. 제안된 구조 분석 방법의 구성도

2.1 수학 함수명 탐색

주어진 입력 데이터로부터 구조 분석을 하기 전에 미리 적당하게 심볼들을 그룹핑할 필요가 있다. 예를 들면 sin, cos, tan와 같은 함수명은 하나의 단위로 취급할 수 있기 때문이다. 본 논문에서는 미리 정의된 수학 함수명을 그림2 같이 trie 구조로 저장하는데 이는 탐색을 편리하게 하기 위한 것이다.

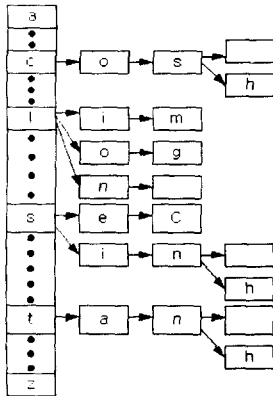


그림 2. 수학 함수명의 계층적인 구조

예를들어, 그림 3과 같은 수식으로부터 얻어진 심볼 인식 결과 리스트가 'Σ', 'x', '=', '1', 'n', 'sin', 'x', '2'이라면 수학 함수명 탐색은 다음과 같이 수행된다 먼저, 첫 번째 unigram에서 심볼의 존재를 탐색한다. 심볼 's'가 발견되면 다음으로 심볼 'i'를 두 번째 unigram에서 찾고 심볼 'n'을 세 번째 unigram에서 찾으면 심볼 's'와 'i' 그리고 'n'을 수학 함수명 'sin'의 의미를 갖는 단위로 그룹시킬 수 있다.

$$\sum_{x=1}^n \sin x^2$$

그림 3. 수식의 예

수학 함수명을 하나로 그룹핑한 후 초기 그래프를 생성한다. 그래프의 노드는 하나의 심볼을 표현하고, 노드의 속성으로 인식된 결과와 Bounding Box 좌표를 가진다. 예지는 노드와 노드 사이의 관계를 표현하는데 초기 그래프에는 예지가 존재하지 않는다. 그림 4는 그룹핑된 심볼들을 그래프로 표현한 것이다.

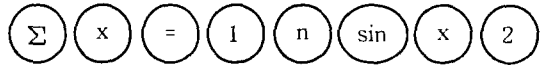


그림 4. 초기 그래프

2.2 순차적 구조 분석

주어진 심볼 S<sub>1</sub>, ..., S<sub>n</sub>을 순차적으로 구조 분석하기 위해서는 수학에서 사용되는 심볼의 표기에 관한 정보와 관계 결정 규칙을 사용한다. 심볼의 표기에 관한 정보는 다음과 같이 기준 심볼의 중심으로부터 이웃하는 심볼들이 나타날 수 있는 상대적인 위치 정보를 말한다.

- 위(90° 방향), 아래(- 90° 방향)
  - Σ, Π, -, lim(아래만 존재), ...
- 위첨자(45° 방향), 아래첨자(- 45° 방향)
  - J, 변수, 상수(위 첨자만 존재), ...
- 평행 (0° 방향)
  - 위의 두경우에 소속된 심볼들, +, -, ×, =, ...

심볼의 Bounding Box 좌표와 심볼의 중심좌표를 이용한 관계 결정 규칙은 다음과 같은 6단계로 구성된다. 이때 초기 기준심볼은 S<sub>1</sub>으로 설정한다.

- 단계1: 기준심볼 S와 심볼 S<sub>i</sub>, i = 2, ..., n의 평행관계 파악
- 단계2: 평행관계인 경우, 심볼 S<sub>i</sub>를 기준심볼로 대체
- 단계3: 평행관계가 아닌 경우, 위첨자(위) 관계인지, 아래첨자(아래) 관계인지를 파악
- 단계4: 위첨자(위) 관계인 경우, 심볼 S<sub>i</sub>가 기준심볼의 첫 번째 위첨자(위) 심볼이면 기준심볼의 노드 하나를 갖는 위첨자(위) 서브 그래프를 생성한다. 이때 심볼 S<sub>i</sub>를 노드에 대응시킨다. 이때 S<sub>i</sub>가 서브 그래프에서 초기 기준심볼 역할을 하게 된다. 그렇지 않은 경우, 서브 그래프의 노드를 하나 증가시키고 심볼 S<sub>i</sub>를 대응시킨다. 그런 후 기준심볼과 심볼 S<sub>i</sub>의 관계를 단계1-단계 5까지를 적용시켜 파악한다.
- 단계5: 아래첨자(아래) 관계인 경우는 위첨자(위) 서브 그래프를 생성하는 대신 아래첨자(아래) 서브 그래프를 생성하고 나머지는 단계4와 동일하게 처리된다.
- 단계6: i 를 1 증가하고, 단계1로 이동

근 심볼( $\sqrt{\quad}$ )이 기준심볼이 되는 경우는 심볼  $S_i$ 가 근 심볼에 포함되는지를 파악하고, 포함되면 단계4나 단계5와 같이 근 심볼의 서브 그래프를 생성하고 동일하게 처리한다.

평행, 위첨자, 아래첨자, 위, 아래 관계를 파악하기 위한 규칙은 다음과 같이 적용된다. 기준심볼과 심볼  $S_i$ 의 Bounding box의 중심좌표  $(x_s, y_s)$ ,  $(x_i, y_i)$ 가 주어졌을 때

- $|y_s - y_i| \leq \alpha$  이면 평행 관계,
- $|y_s - y_i| > \alpha$  and  $y_s < y_i$ , 이면 위첨자(위) 관계,
- $|y_s - y_i| > \alpha$  and  $y_s > y_i$ , 이면 아래첨자(아래)관계

로 결정한다.

### 3. 실험결과 및 분석

고등학교 수학과과서상의 수식을 필기한 100개의 테스트 샘플을 사용해서 제안된 방법을 실험했다. 그림 5에는 구조분석에 성공한 수식이 예시되었으며, a)는 근심볼을 포함하면서 심볼들 사이의 평행 관계만 존재하는 경우에 제안된 방법의 성공 사례이고, b)는 분수기호와 분모, 분자와의 위, 아래 관계를 파악하는 예로 성공적으로 수행되었다.

그림 6은 구조분석에 실패한 수식을 보여주고 있다. 구조분석의 실패는 닫힘기호 ')'가 기준심볼일 때 닫힘괄호 ')'의 위첨자 서브그래프내에서 분수기호 '/'와 심볼 'x'의 관계파악을 수행하면서 이루어졌다. 실패한 이유는 평행 관계 범위내에 명확하지 않은 위첨자 관계가 포함되므로 중심 좌표의 비교만으로 해결되지 않기 때문이다.

이러한 문제를 해결하기 위해서는 수식을 쓸 때 심볼 사이의 간격을 적당히 두어 필기하거나 심볼의 크기에 따른 조합을 고려하여 임계치를 조정하고 심볼들 사이의 관계와 관련된 더 많은 특징들을 구하여 심볼 사이의 관계를 결정하는 규칙을 개선하는 것이다.

입력 :  $\sqrt{x + \sqrt{x+2}} = 1$

출력 :  $\text{sqrt}\{ x + \text{sqrt}\{ x + 2 \} \} = 1$

a)

$$\frac{\begin{matrix} (x-a) \\ (x-b) \\ (x-c) \end{matrix}}{(x-d)}$$

입력 :

출력 : { LEFT( x - a RIGHT ) } OVER { LEFT( x - bRIGHT T ) } OVER { LEFT( x - c RIGHT ) }

b)

그림 5. 구조분석 성공 예

입력 :  $\lim_{x \rightarrow 0} (1 - e^{-\frac{1}{x}})$

출력 :  $\lim_{x \rightarrow 0} \text{LEFT}( 1 - e^{-\frac{1}{x}} \text{RIGHT})$

그림 6. 구조분석 실패 예

### 4. 결론

본 논문에서는 수식의 인식에 필요한 순차적인 구조 분석 방법을 제안했다. 제안된 방법이 평행한 관계, 위, 아래관계에서 효율적으로 작동함을 알 수 있다.

향후 연구로는 보다 다양한 특징을 사용하여 위, 아래첨자를 해결하고 다양한 분야에서 사용되는 수식에 적용하여 관계 결정 규칙을 개선하는 것이다.

### 참고문헌

- [1] D. Blostein and A. Grbavec, "Recognition of Mathematical Notation," in H. Bunke and P.S.P Wang, editors, Handbook of Character Recognition and Document Analysis, pp. 557-582. World Scientific Publishing, 1997.
- [2] R. Anderson, "Two-dimensional mathematical notation," in Syntactic Pattern Recognition, Applications, K. S. Fu ed., pp. 147-177, 1977.
- [3] A. Belaid and J. Haton, "A syntactic approach for handwritten mathematical formula recognition," IEEE Trans. Pattern Analysis and Machine Intell., vo. 6, no. 1, pp. 105-111, 1984.
- [4] S. Chang, "A method for the structural analysis of two-dimensional mathematical expressions," Information Sciences vo. 2, no. 3, pp. 253-272, 1970.
- [5] Masayuki Okamoto, Bin Miao, "Recognition of Mathematical Expressions by Using the Layout Structures of Symbols," ICDAR'91, Saint Malo, France, pp. 242-250, 1991.
- [6] Z. Wang and C. Faure, "Structural analysis of Handwritten Mathematical Expressions," Proc. Ninth Int. Conf. on Pattern Recognition, Rome, Italy, Nov. pp. 32-34, 1988.
- [7] A. Grvavec and D. Blosten, "Mathematics Recognition Using Graph Rewriting," Proc. Third Int. Conf. on Document Analysis and Recognition, pp.417-421, Montreal, Canada, Aug. 1995.
- [8] H. Lee and Me Lee, "Understanding Mathematical expressions using procedure-oriented transformation," Pattern Recognition, vo. 27, no. 3, pp. 447-457, 1994.
- [9] H. Schwenk and M. Milgram, "On-line Character Recognition for Mathematical Expressions," Proceeding French-Korean Workshop Man-Machine-Handwritten Communication, Paris, France, pp. 15-24, 1996.
- [10] Andreas Kosmala and Gerhard Rigoll, "Recognition of On-line Handwritten Formulas," 6th IWFHIR, Taejon, Korea, pp. 219-228, August, 1998.
- [11] K. F. Chan and D. Y. Yeung, "Mathematical Expression Recognition: A Survey," Technical Report HKUST-CS99-04, April, 1999.