

## 순차적 블록 모델을 이용한 필기체 수식 인식

이택현<sup>o</sup>, 김진형, 김기응

한국과학기술원 전자전산학부 전산학전공

three@ai.kaist.ac.kr, jkim@cs.kaist.ac.kr, kekim@cs.kaist.ac.kr

### Handwritten Mathematical Expression Recognition by Sequential Block Framework

Taik Heon Rhee, Jin Hyung Kim, Kee-Eung Kim

Division of Computer Science, School of Electrical Engineering and Computer Science,  
Korea Advanced Institute of Science and Technology

#### 1. 서론

본 연구에서는 '순차적 블록 모델(Sequential Block Model: SBM)'을 이용한 필기체 수식 인식 기법을 제안한다. 순차적 블록 모델은 구문론적 분석 없이 심볼의 위치 관계만으로 수식을 표현한다. 주어진 필기 입력으로 특정한 수식이 표현될 가능성은 심볼 간의 위치 관계의 조합으로 측정될 수 있으며, 심볼을 순차적으로 추가할 때마다 추가한 심볼과 이미 추가된 심볼간의 관계만을 분석하여 전체 수식의 구조를 쉽게 갱신할 수 있다. 제안 모델을 이용하여 심볼 별로 분할 및 인식이 수행된 필기 입력으로부터 심볼을 하나씩 순차적으로 추가해가면서 휴리스틱 탐색 트리 [10]을 생성함으로써 수식 인식이 이루어진다.

제안 방법의 장점은 다음과 같이 세 가지로 요약할 수 있다. 첫째, 미완성된 수식에 심볼을 하나씩 순차적으로 추가하며 수식을 완성시키는 방법을 사용하므로, 두 개 이상의 하위식 단위로 병합하는 구문론적 분석 기법에 비해 더 좁은 트리 탐색이 가능하여 효율적으로 탐색 공간이 축소된다. 둘째, 제안 모델을 통해 미완성된 수식에 추가된 심볼들로부터 아직 추가되지 않은 심볼 각각의 위치를 한정시킬 수 있으므로, 이를 통해 허용가능(admissible)하면서 빠른 탐색을 가능하게 하는 휴리스틱을 적용할 수 있다. 셋째, 입력에 대한 모든 가능한 형태의 수식을 전역적으로 탐색하므로 변이가 심한 필기체 수식에 대해서도 안정적인 인식이 가능하다.

#### 2. 본론

순차적 블록 모델, 즉 SBM은 직접적인 구문론적 분석 대신 수식 내 심볼 간의 위치 관계만 설명하는 모델이다. SBM 내에서 수식은 수평으로 나열된 하나 이상의 블록(block)의 조합으로 표현된다. 각 블록은 하나의 중심심볼(key-symbol)과 이에 종속된 하위식의 집합으로 이루어지고, 각 하위식은 그 자체로서 하나의 수식으로 간주되어 그 내부 구조는 다시 SBM으로 재귀적으로 표현된다. 그림 1은 SBM의 구성에 대한 예로서, SBM은 그림 1(a)와 같은 수식에 대해 그림 1(b)와 같이 표현한다. SBM에서 임의의 두 심볼 간의 위치 관계는 다음 네 가지 타입 중에 하나로 설명할 수 있다. 그림 1(b)의 화살표는 수식 내에서 각 타입 별 위치 관계의 분류 기준을 나타낸다.

Type 1: 인접한 블록 내의 두 중심심볼의 관계 (파선 화살표: row\_right).

Type 2: 서로 다른 블록에 속한 두 심볼 간의 관계 (굵은 화살표: right).

Type 3: 한 블록 내에 중심심볼과 이에 종속된 하위식에 포함된 심볼과의 관계(실선 화살표: (sub, sup, over, under, inside)).

Type 4: 한 블록 내에서 공통된 중심심볼로부터 종속된 서로 다른 하위식에 위치한 두 심볼 간의 관계 (점선 화살표: below).

위와 같이 SBM에서는 모든 심볼 쌍의 관계를 표현할 수 있으므로, 주어진 필기 입력으로 특정한 수식이 표현될 가능성은 모든 심볼 쌍의 위치 관계의 조합으로 계산할 수 있고, 이를 구조적 우도(structural likelihood)로 정의한다. SBM에서 사용하는 위치 관계는 지배심볼(dominant symbol)을 기준으로 피지배심볼(dominated symbol)의 상대적인 위치에 따라 결정되며, 본 연구에서 8가지 위치 관계를 정의하여 사용하였다.

SBM을 사용한 휴리스틱 탐색에 의한 필기체 수식의 인식 과정은 다음과 같다.

- 0) 필기 입력에 대해 먼저 심볼 분할 및 인식 과정을 통해 심볼의 집합을 생성한다. (전처리 과정)
- 1) 각 심볼 별로 심볼 하나를 이루어진 수식을 생성하고, 우선순위큐(priority queue)에 삽입(push)한다. 이 때, 우선순위큐 각 항목은 평가함수 (evaluation function)에 의해 올림차순으로 정렬한다. 평가함수는 위에서 설명한 구조적 우도와 휴리스틱의 조합으로 계산된다.
- 2) 우선순위큐의 처음에 위치한 수식을 추출(pop)하고,  $E_{first}$  라 칭한다.
- 3)  $E_{first}$  가 입력에 사용된 모든 심볼을 포함하는 경우, 이를 인식 결과로 사용하고 전체 과정을 종료한다. 그렇지 않을 경우 4)로 이동한다.
- 4)  $E_{first}$  에 추가되지 않은 심볼에 대해 각 심볼마다 가능한 모든 방법으로  $E_{first}$  에 추가하면서 새로운 수식을 만든다. 이 때, 추가되는

심볼은  $E_{first}$  내의 모든 심볼로부터 피지배심볼이 되도록 입력의 방법을 제한한다.

5) 4)에서 생성한 모든 수식을 1)에 정의된 우선순위에 삽입한다. 그 후, 2)로 이동하여 과정을 반복한다.

이 중, (4)에서의 새로 추가되는 심볼이 피지배심볼이 되어야 한다는 조건은 두 가지의 중요한 의미를 지닌다. 첫째, 탐색 과정에서 어떤 순서로 심볼이 입력되는지 무관하게 위의 조건으로 인해 새로 생성되는 수식은 우선순위에 따라 모든 수식과 모두 다르다는 것을 보장할 수 있다. 이를 통해 새로운 수식을 우선순위에 삽입할 때마다 우선순위에 내에 동일한 수식이 존재하는지 확인할 필요가 없어 탐색 속도를 빠르게 할 수 있다. 둘째, 특정한 수식에 추가되지 않은 모든 심볼은 향후 어떤 순서로 수식에 추가되는지 관계없이 해당 수식 내 모든 심볼의 피지배심볼이 되므로, 이를 이용해 허용가능한 휴리스틱을 사용할 수 있다. 본 연구에서 제안한 허용가능한 휴리스틱은 불완전한 수식 내의 심볼들과 이에 추가되지 않은 심볼들과의 관계를 실제로 추가하기 전에 예측할 수 있다는 점에서 상당한 정보력(informedness)을 가지고 있다.

제안 방법을 통해 빠른 탐색으로 수식 인식을 수행할 수 있음을 보이기 위해, 온라인으로 입력된 필기체 수식으로부터 탐색의 반복(iteration) 횟수, 탐색 시 생성한 수식의 수, 탐색 진행 시간 등을 측정하였다. 4명으로부터 수집된 20개의 수식을 대상으로 한 결과는 그림 2와 같다. 그림 2는 본 논문에서 제안한 휴리스틱 탐색과 최상 우선 탐색 간의 탐색 반복 횟수 및 탐색 시간을 비교한 그래프로써, 제안한 휴리스틱을 사용 시 탐색 공간 및 시간이 줄어듦을 확인할 수 있다. 특히, 최대 20개의 심볼을 사용하는 경우에도 제안 방법은 최장 300밀리초 이내에 인식 결과를 내어주므로, 휴리스틱 탐색이 본질적으로 지수적 시간(exponential time)으로 풀리에도 불구하고, 실용적인 적용의 가능성을 보인다. 인식의 정확도 관점에서도 20개의 수식 중 3개만이 오인식되고, 오인식된 수식의 형태도 정답과 크게 다르지 않아 정확한 인식이 이루어짐을 확인하였다. 그림 3은 정인식 및 오인식 된 결과의 예를 보여준다.

3. 결론

본 연구에서는 순차적 블록 모델과 이를 이용한 허용가능하면서도 효율적인 휴리스틱의 제안을 통해 입력에 대해 가능한 모든 수식 중 최적의 수식을 찾아주는 휴리스틱 탐색을 이용한 필기체 수식 인식 기법을 제안하였다. 제안 방법은 전역적 탐색을 하면서도 실용적으로 사용이 가능한 인식 속도를 내어주어, 번거로워 보이는 필기체 수식 입력 또는 수식의 심볼 종류 및 입력 범위가 넓어도 효과적으로 적용이 가능하다.

본 연구에서는 제안 방법의 설명을 간단히 하고, 실험을 효율적으로 진행하기 위해 필기 입력에 대해 먼저 심볼 분할 및 인식 과정이 수행된다고 가정을 하였으나, 위의 가정이 없이 다량의 심볼 분할 및 인식 후보를 생성하는 경우에도, 휴리스틱과 평가함수의 작은 변형 만으로 충분히 휴리스틱 탐색이 가능하므로, 본 연구를 통해 보다 실용적인 필기체 수식 입력 시스템의 개발이 용이할 것이다.

향후 연구로서, 좀 더 많은 사용자로부터 데이터를 수집하여 정량적인 성능 평가를 수행함과 동시에, 더 복잡한 구조의 수식에 대해서도 빠른 탐색이 가능하도록 구문론적 분석 기법을 도입할 필요가 있다. 수식 내의 괄호의 매칭 및 알파벳과 숫자의 순서 등 문법적인 요소가 도입되면 위치 관계만으로는 우열을 비교하기 힘든 경우에 대해 좀더 효율적인 비교가 가능할 것이다.

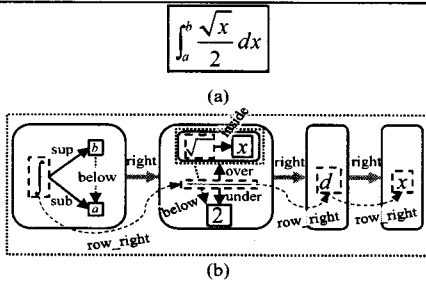


그림 1. 순차적 블록 모델에 의한 수식의 표현

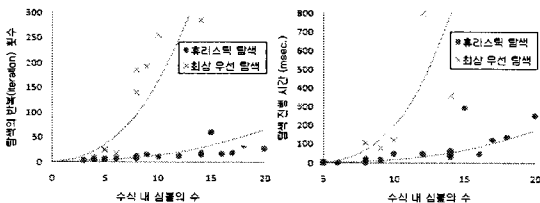


그림 2. 휴리스틱 탐색과 최상 우선 탐색 간 탐색 반복 횟수 및 탐색 시간을 비교한 그래프

$$\int_1^2 \left( \frac{x^2-1}{x^2} \right) e^{x+\frac{1}{x}} dx$$

$$\int \frac{e^x}{e^x+3} dx$$

$$\lim_{x \rightarrow \infty} \int_0^x e^{-y^2} dy = \frac{\sqrt{\pi}}{2}$$

$$\frac{1}{2} x^{-\frac{1}{2}} = \frac{1}{2\sqrt{x}}$$

필기 입력	인식 결과	정답
$b_f p_a$	$b, p_a$	$b, p_a$
$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$	$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$	$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

그림 3. 필기체 수식의 정인식 및 오인식의 예. 정선 사각형 내 수식은 정인식 된 경우, 하단의 표는 오인식의 예를 나타냄.