형태소 분석을 이용한 문자인식 에러의 검출

o

김윤호, 이종국, 김항준, 이상조

경북대학교 컴퓨터공학과

A Method of Detecting of OCR error using Morphological Analysis

Kim Yun-Ho, Lee Jong-kuk, Kim Hang-Joon, Lee Sang-Jo

Dept. of Computer Engineering, Kyung-Pook National Univ.

요 약

문자인식에 있어서 인식율을 높이기 위한 후처리의 한 방법으로서, 문법 정보를 이용하는 후처리를 제안하고자 한다. 즉, 문자 인식 시스템에 의해 인식된 국어문에 대해서 오인식된 문자를 포함하는 어절을 검출하고, 오인식된 문자의 적절한 후보를 선정하여 그에 따라 자동수정을 행하는 것을 전체 후처리 과정으로 전제한다. 본 논문에서는 형태소 분석을 통해 오인식된 부분을 검출하는 과정을 보임으로써 문자인식에 있어서 문법 정보를 이용하는 후처리의 가능성과 그 유효성을 보이는 것을 목적으로 한다.

I. 서 론

국어문의 문자인식에 있어서 인식율을 향상시키기 위한 노력이 다각적으로 진행되어 왔다. 그러나 대부분의 후처리 방법은 인식모듈 자체에서의 성능 향상에 의존하는 것인데, 한글 문자의 기하학적인 특성과 인쇄종이의 불균일성, 해상도의 한계성 등으로 인하여 인식율의 향상에 근본적인 한계를 갖고 있다. 따라서, 인식 방법 자체에 대한 수정이나 갱신이 아닌 인식된 결과에 대해서 후처리를 적용하여 인식율을 높일 수 있는 방법이 연구되어 왔다[3][6].

본 논문에서는 문자 인식에 있어서 인식율을 높이기 위한 후처리의 한 방법으로서, 국어의 문법 정보를 이용하는 후처리 방법을 제안하고자 한다. 즉, 문자 인식 시스템에 의해 인식된 국어 문장에 대해서 형태소 분석을 행함으로써 오인식된 문자를 포함하는 어절을 검출하고, 그 어절에 대한 엄밀한 재분석을 통하여 오인식된 문자의 적절한 후보를 선정하여 그에 따라 자동수정을 행하는 것을 전체 후처리 과정으로 하는데, 본 논문에서는 오인식된 부분을 검출하는 과정을 보이고 그 부분에 대한 자동수정의 방법들을 고찰함으로써, 문자인식에 있어서 문법 정보를 이용하는 후처리의 가능성과 그 유효성을 보이는 것을 목적으로 한다.

Ⅱ. 후처리 시스템의 개요

본 논문에서 제시하는 후처리 시스템은 OCR로 인식된 국어문에 대하여 음운 성분을 분석하는 부분, 형태소를 분석하는 부분, 어휘 정보를 갖고 있는 사전 등으로 구성된다 (그림 1).

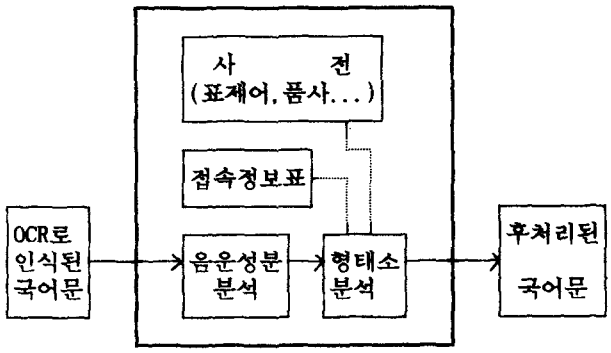

그림 1. 후처리 시스템 구성도.

오인식 문자의 검출을 위한 문법 정보로서 형태론적 정보(품사, 형태소 접속 정보)를 이용한다. 즉, 오인식된 문자를 포함하는 어절은 형태소 분석에서 실패할 것이므로, 형태론적 정보를 이용한 형태소 분석을 통하여 그러한 어절의 검출이 가능하다. 그런데, 오인식된 문자를 포함하는 어절이 형태론적으로 분석이 가능한 경우가 있는데, 이것은 통사적 또는 의미적 분석을 행함으로써 해결될 수 있는 것으로서, 이용하는 문법 정보를 통사정보, 의미정보 등으로 확장함으로써 보완이 가능하다.

형태론적 분석은 어절 단위의 음운 성분 분석과 형태소 분석의 두단계로 나뉘어 행해진다. 즉, 주어진 어절에 대하여 국어에서 가능한 음운 성분들을 분석하고 이것들에 대해서만 형태소 분석을 행함으로써, 국어에서 존재하지 않는 음운 성분들에 대하여 형태소 분석을 하지 않도록 한다. 또한 음운 성분 분석과 형태소 분석의 기법으로서는 활성 차트를 이용한 좌-단방향 차트 파싱[4]을 이용하며, 이용되는 형태론적

정보중에서 품사는 사전에 등재되고 접속정보[1][5]는 표로 구성되어 형태소 분석기 내에서 절차적으로 처리된다.

입력문에 사용되는 문자의 종류로는 한글과 문장부호만을 허용하며, 한글코드로는 2-바이트 조합형을 사용한다. 형태론적으로는 단일층위(표층)를 전제로하여 불규칙 활용의 경우 활용된 형태를 사전에 등재하는 방식을 채택하여 사전의 잉여성을 감수하는 반면에 형태소 분석을 위한 알고리즘의 간단성을 취한다. 비통사적 복합어[2]는 사전에 등재되어 있고, 입력문에 대한 모든 형태소들이 사전에 등재되어 있는 것을 가정한다.

문자 인식의 결과로서의 입력문을 분석하는 전체적인 절차는 다음과 같다.

(1) 국어의 형태소 분석은 어절 단위로 수행되므로, 입력문을 어절 단위로 분리하여 처리한다.
(예) 영희가 가방을 본다 => 영희가/가방을/본다

(2) 분리된 각 어절을 낱자 단위로 분리한다.
(예) 영희가 => 영/희/가

(3) 국어에서 한 음소가 한 형태소가 되는 경우가 있으므로 음소 단위로 분리된 형태에서 형태소 분석이 진행되어야 한다. 따라서 낱자들을 음소 단위로 최종 분리한다. 이때 위치정보(초,중,종성)를 보존한다.
(예) 영 => ㅇ(초성)/ ㅕ(중성) / ㅇ(종성)

(4) 최종 분리된 음소들에 대해서 주어진 어절의 음운 성분들을 얻기 위한 분석을 한다.
(예) 영희가 => ㅇ(초성)/ ㅕ(중성)/여/ ㅇ(종성)/영/ㅎ(초성)/ㅢ(중성)/희/ ㅇ희/
영희/ㄱ(초성)/ ㅏ(중성)/가/희가/ ㅇ희가/영희가

(5) 분석된 음운 성분들에 대해서 형태소 분석을 한다.
(예) 영희가 => 영희+가

Ⅲ. 활성 차트를 이용한 단방향 차트 파싱

음운 성분 분석과 형태소 분석을 위한 자료 구조로서 INPUT, PEND-EDGES, 그리고 CHART를 이용한다. INPUT은 음운 또는 형태소 성분에 노드 정보를 추가하여 에지로 만든 다음, 이들을 리스트 형태로 저장하고 있는 자료구조이며, PEND-EDGES는 분석을 위해 대기 중인 활성 에지를 저장하기 위한 자료 구조이고, CHART는 이미 분석이 종

료된 비활성 에지를 저장하는 자료구조이다.

에지는 에지 번호, 노드번호, 그리고 주어진 어절의 부분적 음운값으로 구성된다. 예를 들어, '영'에 대한 음소분리의 결과는 그림 2와 같이 표시된다.

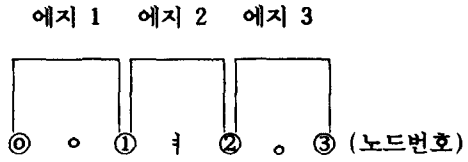

그림 2. '영'에 대한 에지와 노드.

분석이 가해지는 에지를 '현재 에지'라 하며, 현재 에지의 시작 노드번호를 끝 노드번호로 하는 에지를 '좌인접 에지'라 한다.

파싱 기제는 자료 구조로서 차트[7][8]를 이용하는데, 현재 에지에서 좌측의 한 에지(좌인접 에지)만을 바라보고 결합의 가능성을 체크하는 좌-단방향의 차트 파싱[4]을 한다. 그 동작을 간략하게 기술하면 다음과 같다.

(1) PEND-EDGES가 비어 있으면, INPUT이 null일 때까지 INPUT으로부터 하나의 에지를 가져와 PEND-EDGES에 넣는다.

(2) PEND-EDGES가 비어 있지 않으면, 여기서 하나의 에지를 가져오고 이 에지에 대한 좌인접 에지를 찾는다.

(3) 현재 에지와 좌인접 에지의 결합을 시도한다.

(4) 만일 성공적으로 결합되면, 결합된 에지는 후속 분석을 위하여 PEND-EDGES에 첨가된다.

(5) 분석이 끝난 에지를 CHART에 등록한다.

(6) 이러한 과정을 PEND-EDGES가 null일 때까지 계속한 다음, INPUT이 null이 아닌 동안 INPUT에서 한 에지를 가져와 계속한다.

Ⅳ. 음운 성분의 분석

차트 파싱을 하게 되면 계산적으로 가능한 모든 음운 성분들이 분석되는데, 이중에는 국어에서 비현실적인 음운 성분들을 포함하게 되며, 또한 음운적으로 중복되는 음운 성분들을 생성하게 된다. 따라서 국어에서 현실적으로 존재하는 음운 성분만을 생성하게 하고 또한 음운 성분들을 중복 생성하지 않게 함으로써, 불필요한 형태소 분석의 부담을 줄여준다.

본 논문에서 채택한 파싱 기법은 현재에지를 중심으로 좌인접에지에 대하여 결합 가능성을 체크하는데, 음운 성분의 중복 생성을 방지하기 위하여 현재 에지가 자소나 낱자를 음운으로 가지는 경우에만 결합 가능성을 체크한다. 즉, 예를 들어 '철수'란 음운 성분은 '철+수'로부터 또는 '처+ㄹ 수'로부터 생성될 수가 있는데 어떤 조합에 의해서 생성되었는가는 음운 성분 분석에 있어서는 무의미하며, 다음 단계인 형태소 분석시 같은 음운 성분에 대해서 중복 분석을 하게 할 뿐이다. 또한, 좌인접 에지가 어절인 경우에 그 어절의 마지막 낱자의 음운 형태만이 음운 성분의 생성시에 의미를 가지므로 음운 결합을 할 때에 좌인접 에지는 낱자와 어절의 마지막 낱자를 보고 음운 성분의 생성 여부를 판단한다. 따라서, 좌인접 에지와 현재 에지가 가질 수 있는 음운 성분의 형태는 다음의 표 1과 같다.

표1. 좌인접에지와 현재에지의 음운 성분 형태.

좌인접 에지	현재 에지
자소 낱자*	자소 낱자

(낱자*: 낱자 또는 어절의 마지막 낱자)

또한, 입력 문장을 음소 단위로 분리할 때에 각 음소의 위치정보(초성,중성,종성)를 보존함으로써, 음운 성분을 생성할 때에 예를 들어 뒷낱자의 초성이 앞낱자의 초성으로 결합되는 것과 같은 경우를 방지한다. 결과적으로, 국어에서 현실적인 음운 성분의 생성을 위하여 결합이 가능한 경우는 다음의 표들(표 2-5, X는 결합 불가를 나타냄)과 같다.

표2. 자소(좌:좌인접 에지)와 자소(우:현재 에지)의 경우

좌 \ 우	초성	중성	종성
초성	X	결합	X
중성	X	X	X
종성	X	X	X

표3. 자소(좌:좌인접 에지)와 낱자(우:현재 에지)의 경우

좌 \ 우	낱자(초+중성)	낱자(초+중+종성)
초성	X	X
중성	X	X
종성	결합	결합

표4. 낱자*(좌:좌인접 에지)와 자소(우:현재 에지)의 경우

좌 \ 우	초성	중성	종성
낱자* (초+중성)	X	X	결합
낱자*(초+ 중+종성)	X	X	X

표5. 낱자*(좌:좌인접 에지)와 낱자(우:현재 에지)의 경우

좌 \ 우	낱자(초+중성)	낱자(초+중+종성)
낱자* (초+중성)	결합	결합
낱자*(초+ 중+종성)	결합	결합

음운 성분의 분석을 위한 알고리즘을 개략적으로 기술하면 알고리즘 1과 같다.

```
음운성분분석알고리즘()
{
   한 어절을 음소단위로 분리;
   분리된 음소들로 INPUT 리스트를 초기화;
   PEND-EDGES, CHART 리스트를 null로 초기화;
   반복(INPUT != null)
   {  INPUT에서 PEND-EDGES로 하나의 에지를 가져옴;
      반복(PEND-EDGES != null)
      {  PEND-EDGES에서 하나의 에지를 가져옴; /* 현재 에지 */
         현재 에지에 대해서 좌인접 에지들을 구함;
         만일 좌인접 에지와 현재 에지의 두 음운성분들이 결합가능하면,
         { 두 음운들을 결합;
           후속 분석을 위해 PEND-EDGES에 첨가; }
      }
      분석이 종료된 현재 에지를 CHART에 등록;
   }
}/**/
```

알고리즘 1. 음운 성분의 분석을 위한 알고리즘

V. 형태소 분석과 오인식의 검출

음운 성분 분석을 하게 되면 주어진 어절에 대한 가능한 음운 성분들이 결과로 남게 된다. 이것들에 대해서 사전을 검색하여 사전에 등재되어 있는 것들에 대해서 형태소 분석을 행한다. 즉, 사전에 존재하지 않는 음운 성분은 음운 구조적으로는 가능하나 국어에서 형태소로는 존재하지 않는 음운 성분이므로 이러한 음운 성분들에 대해서는 형태소 분석을 하지 않음으로써 형태소 분석의 부담을 경감시킨다.

분석된 음운 성분들에 대한 형태소 분석은 접속정보표와 표제어와 같이 등재되어 있는 품사정보를 이용해서 결합가능성을 판별한다. 예를 들어, '철수(고유명사)'와 '가(주격조사)'의 결합 가능성이 체크된다면, 접속정보표에서 격조사는 그 왼편에 올 수있는 품사로서 고유명사를 갖고 있으므로, 두 음운 성분은 결합되어 '철수가'라는 더 큰 단위의 음운 성분을 생성하게 되는데, 본 논문에서는 좌-단방향 차트 파싱을 하므로 좌/우 접속정보 중에서 좌접속정보만을 필요로 한다 (그림 3).

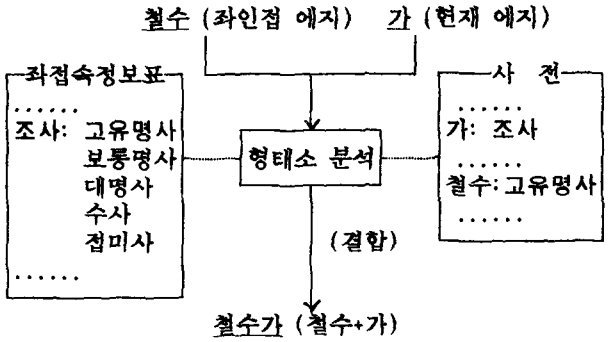

그림 3. '철수가'의 형태소 분석

주어진 어절에 대한 형태소 분석의 결과 성공하면 그 어절을 그대로 출력하고, 실패하면 그 어절은 오인식된 문자를 포함하고 있는 것이므로 표시를 하여 출력한다. 예를 들어, 원래의 문서에서는 '단위로'인 것을 '탄위로'로 인식하였다면 형태소 분석에 실패하므로 오인식 어절임을 표시해서 {탄위로}를 출력한다 (그림 4).

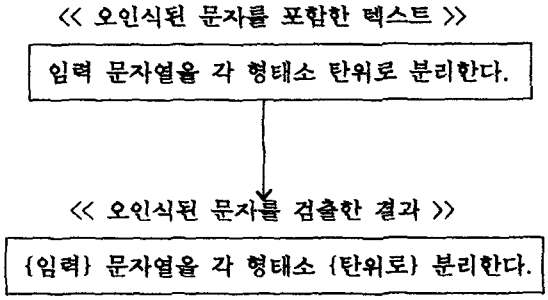

그림 4. 텍스트 처리의 예

형태소 분석을 위한 알고리즘을 개략적으로 기술하면 알고리즘 2와 같다.

```
형태소분석알고리즘()
{
   분석된 음운성분들로 INPUT 리스트를 초기화;
   PEND-EDGES, CHART 리스트를 null로 초기화;
   반복(INPUT != null)
   {  INPUT에서 PEND-EDGES로 하나의 에지를 가져옴;
      반복(PEND-EDGES != null)
      {  PEND-EDGES에서 하나의 에지를 가져옴;  /* 현재 에지 */
         현재 에지에 대해서 좌인접 에지들을 구함;
         만약 좌인접에지와 현재에지가 사전에 등재되어 있으면,
         {
            만약 좌접속정보표에서 현재 에지의 좌접속값이
                 좌인접 에지의 품사값과 같으면,
            { 두 형태소들을 결합;
              후속 분석을 위해 PEND-EDGES에 첨가; }
         }
      }
      분석이 종료된 현재 에지를 CHART에 등록;
   }
   주어진 어절에 대한 형태소 분석의 성공/실패 여부를 출력;
}/**/
```

알고리즘 2. 형태소 분석을 위한 알고리즘

VI. 앞으로의 연구과제

오인식 문자를 포함한 어절을 검출하였을 때, 그것에 대한 적절한 후보를 선정하는 방법을 모색하고, 이에 따른 자동수정을 행함으로써, 전체 후처리 시스템을 구현하는 것이다. 후보를 선정하는 방법으로서 첫째, 오인식된 어절의 각 음소에 대해서 그 위치(초성,중성,종성)에 상응하는 모든 음소를 대입하는 방식과 둘째, 오인식된 어절의 음운 성분에 대해서 사전에 등재되어 있는 같은 길이의 모든 표제어와 비교해서 음소 단위로 차이가 가장 적은 단어를 사전으로부터 찾아서, 그것들을 후보로 선정하는 방식과 셋째, 인식 시스템의 인식 오류 경향을 도입하여, 인식 오류 경향을 보이는 음소들이 그 어절에 포함되어 있는지를 보고 포함되어 있으면, 그 음소에 대해서 자신을 제외한 인식 오류 경향 음소들을 대입하여 후보를 선정하는 방식 등을 상정해 볼 수가 있겠는데, 계속 연구중에 있다. 또한, 후보들이 선정되었을 때, 자동수정을 행하는 방법으로서 첫째, 후보가 하나인 경우에는 자동수정을 행하고, 둘째로 후보가 둘 이상인 경우에는, 후보들을 모두 출력하여 사용자가 그 후보들 중에서 올바른 것을 선택하도록 하는 방식과 후보들 중에서 가장 우월한 후보를 선택해서 사용자의 개입없이 자동수정을 행하는 방식을 고려하고 있는데, 이 역시 더 연구되어야

할 사항들이다.

Ⅶ. 결 론

본 논문에서는 문자인식의 인식율을 높이기 위한 후처리의 방법에 대하여 고찰하였다. 즉, 문자인식에 의한 국어문을 입력으로 하여 형태소 분석을 행함으로써 오인식 문자를 포함한 어절을 검출한다. 음운 분석과 형태소 분석시에 국어에서 비현실적인 음운 성분 및 형태소들을 억제함으로써 분석의 효율성을 높였으며, 형태소 분석을 통하여 오인식을 포함한 어절이 검출됨을 보였다. 그러나, 형태소 분석만으로는 오인식을 검출할 수가 없는 경우가 있는데, 이것은 통사적 또는 의미적 처리를 가함으로써 해결이 가능할 것으로 보여진다.

참 고 문 헌

[1] 김성용, "Tabular Parsing방법과 접속 정보를 이용한 한국어 형태소 분석기", 한국과학기술원 석사학위 논문, 1987.

[2] 남기심, 고영근, 표준 국어 문법론, 탑출판사, 1985.

[3] 서영훈, "의미 정보를 이용한 중심어 주도의 한국어 파싱", 서울대 박사학위 논문, 1991.

[4] 민병우, 이성환, "문자 인식을 위한 오인식 수정 기술", 한국정보과학회지, 제9권 제1호, 1991.

[5] 이상조, 한국어 자연어 인터페이스를 위한 사전 구성에 관한 연구(II), 경북대 전자기술연구소, 한국전자통신연구소 위탁과제, 1992.

[6] Itoh and Maruyama, "A Method of Detecting and Correcting Errors in the Results of Japanese OCR", Tokyo Research Laboratory, IBM Japan Ltd., 1991

[7] Varile, "Charts: A Data Structure for Parsing", *Parsing Natural Language*, ed. M. King, 1983

[8] T. Winograd, "Language as a Cognitive Process, Volume 1: Syntax", 1983.