論文97-34C-5-9

문자 인식 후처리를 위한 형태소 분석기와 문자 교정기의 구현

(Implementation of Morphological Analyzer and Spelling Corrector for Character Recognition Post-Processing)

李 英 和 *, 金 桂 成 *, 金 永 勳 **, 李 相 祚 *

(Young Hwa Lee, Kye Sung Kim, Young Hun Kim, and Sang Jo Lee)

요 약

본 논문은 형태소 분석기와 문자 교정기를 이용하여 문자 인식기에서 발생하는 오인식 문자의 교정을 위한 후처리 방안을 제시한다. 후처리 과정은 먼저 뭉치어구에 대한 불필요한 분석을 피하고 문자 교정에 필요한 정보를 출력하는 형태소 분석기가 오인식 어절과 오인식 문자의 위치를 추정하여 결과를 출력한다. 다음으로 추정된 오인식 문자에 대해 문자 대치 테이블과 자소 대치/분리 테이블을 이용하여 후보 문자를 생성하고 오인식 어절로 대치시킨 후 형태소 분석을 재시도한다. 끝으로 형태소 분석에 성공한 후보 문자들의 신뢰도를 분석하여 우선순위를 정하고 적절한 최종 후보를 선택한다. 교정 테이블은 문서 인식기로 인식된 말뭉치(corpus)에서 오인식 문자를 조사하여 구성하였고, 신뢰도 분석에서는 우리말 역순 사전의 등록어 빈도수와 말뭉치에서 임의 추출된 약 10만 어절에 대한 조건부 발생 확률을 사용하였다. 어느 인식 시스템을 가지고 문서를 인식한 결과 인식률이 93%인 문서에 대하여 본 시스템으로 교정한 결과 97%로 인식률이 향상되었다.

Abstract

In this paper, we propose post-processing method that corrects a misrecognized character by generated a character recognizer using morphological analyzer and spelling corrector. The proposed post-processing consists of three phases : First, our method pass through morphological analyzer which only outputted necessary information for spelling correcting, doesn't analyze a bundle of phrases, and detects the location of misrecognized character. Second, tagging the generated candidate character using the information of character substitution table and grapheme substitution/separating table. Then we retry analysis after the misrecognition character has been substituted. Finally we select the final candidate that analysis reliability for producing candidate characters. For making correction table, we investigate misrecognized characters in CORPUS. Reliability analysis used to frequency of registration word of the backward dictionary and conditional production probability that has been randomly selected about 100,000 words in CORPUS. A Korean character recognizer demonstrates 93% correction rate without a post-processing. The entire recognition rate of our system with a post-processing exceeds 97% correction rate.

* 正會員, 慶北大學敎 컴퓨터工學科

(Dept. of Computer Engineering, Kyungpook National University)

** 正會員, 安東專門大學 電算科

(Dept. of Computer Science, Andong Junior College)

接受日字:1996年11月14日, 수정완료일:1997年4月21日

Ⅰ. 서 론

사회의 정보화가 진척되고 산업의 다양화와 전문화로 인하여 사람이 다루어야 하는 정보의 양이 급증하게 되었다. 이로 인하여, 막대한 양의 정보를 자동으로 입력하기 위한 문서 인식 시스템이 개발되어 널리 활용되기 시작하였다.

한글 문서 인식 시스템은 크게 전처리, 문자 인식,

후처리의 세 단계로 구성되며, 문자 인식 단계에서 발생하는 오인식 문자를 교정하기 위하여 철자 교정기의 기능을 수행하는 후처리 단계가 필요하다. 후처리는 개별 문자의 인식은 물론이고 어절 내 문자간의 접속 정보와 문장의 구조, 의미 등의 다양한 문맥적 지식 등을 이용하여 오인식 문자를 교정하는 단계이며 인간의 문서 인식 기능에 근접한 시스템을 구현하기 위한 문서 인식의 최종 처리 과정이다.

지금까지의 오인식 문자 교정 방법은 문자 인식 분야의 학자들에 의해서, 혹은 자연어 처리 분야의 학자들에 의해서 각각 조금씩 다른 접근으로 행해지고 있다[1,2,3]. 문자 인식 분야의 접근은 문자 사이의 통계적 분포를 문맥 정보로 이용하는 통계적 방법이며, 후자에 의한 접근은 구조적 방법으로 단어 사전, 문장 구조 그리고 의미의 구조적 표현에 기반한 방법이다. 이외에도 두 가지 방법을 복합하여 사용하기도 한다.

통계적 방법에 의한 교정은 확률적 수치에 의존하여 교정 후보 문자를 선택할 경우, 실제로 자연어에서 사용 빈도나 자소 간의 발생 확률이 낮은 후보 문자가 문맥상 의미를 고려하여 빈도나 확률이 높은 후보 문자보다 더 적절한 교정 후보로 선택되어야 하는 경우가 있다. 그리고 문맥적 의미를 고려한 구조적 방법에 의한 후처리는 인식된 문장에 대하여 구문 분석과 의미 분석의 단계를 거쳐야만 정확한 교정이 가능하며 교정률을 보다 향상시킬 수 있다. 그러나 실제로 복잡한 구문 구조를 가진 한국어 문장의 분석은 아직까지 처리가 어려우며 계속 연구중에 있다.

문자 교정의 구조적 방법과 다소 차이는 있지만 한글 맞춤법 교정기도 맞춤법이 맞지 않는 문자들을 교정한다[4,5]. 하지만 이러한 맞춤법 교정기와는 달리 문자 인식 후처리에서 사용하는 오인식 문자 교정기는 입력 문서가 사람이 아닌 컴퓨터 프로그램의 수행 결과라는 사실을 고려하여, 교정 후보 문자의 선택시에 한글 자소의 결합 원칙과 문자 사이의 의존 관계에 대한 특성 외에도 인식기에 의해 발생하는 오류를 감안해야 한다. 또한 특정 문자에 대한 반복적인 오인식이 발생하므로 이를 먼저 학습시켜 교정에 반영함으로써 문서 인식률을 높일 수 있다. 따라서, 한글의 오인식 문자 교정을 위해서는 한글의 특성에 대한 연구가 선행되어야 하고, 입력된 문서가 사람에 의한 키보드 입력이 아니라 문자 인식기를 통한 컴퓨터 처리 결과임을 감안하여야 한다. 또한 문자의 오류가 인식기의 특성에 따라 차이를 보이므로, 교정 후보를 선택하는 과정에서도 이러한 인식기의 특성을 반영해야 한다.

이러한 점을 고려하여 본 논문에서는 한글 문서 인식기의 오인식 문자를 교정하는 후처리 시스템을 구현하였다. 본 시스템의 후처리 과정은 세 단계로 나누어진다. 첫 단계는 오인식 문자 검출을 위한 형태소 분석 과정으로, 여기서 형태소 분석기는 오인식 어절을 검출하고 그 위치를 추정하여 문자 교정기에 필요한 정보를 넘겨 준다. 다음 단계에서는 문자 인식기의 특성을 고려한 유사 구조 문자 정보와 실제 사용 빈도수에 따른 확률 정보를 기반으로 구성한 자소 및 문자 대치 테이블, 자소 분리 테이블을 이용하여 교정 후보 문자를 선정하고 이 후보 문자를 오인식 어절로 대치하여 형태소 분석을 재시도한다. 마지막으로 전단계에서 형태소 분석에 성공한 후보 문자들 중에서 최종 후보를 결정하기 위해 신뢰도 분석을 행한다. 이 때 실용성을 고려하여 초등학교 교과서 및 동아 일보 사설 등의 말뭉치(corpus)[1]로부터 얻어진 사용 빈도수와 자소 발생 확률을 사용한다. 이러한 처리 과정을 통해 가장 적절한 후보를 선정하고 그에 따른 자동 수정을 행한다.

본 논문의 구성은 서론에 이어 2장에서는 후처리를 위한 형태소 분석기의 필요성과 형태소 분석기에서 사용하는 사전의 구조 및 사전의 종류, 형태소 분석 알고리즘을 소개하고 이를 통한 오인식 어절의 검출 방법에 대해 설명한다. 3장에서는 교정 후보 문자를 선정하기 위해 사용된 여러 정보들과 후보 문자들의 신뢰도 분석 방법을 제안한다. 끝으로, 4장에서는 전체 시스템의 구성과 실험 결과를 평가하고 5장에서는 결론을 맺는다.

Ⅱ. 형태소 분석기

1. 후처리를 위한 형태소 분석기의 필요성

형태소 분석기의 기능은 응용 분야와 사용 목적에

1) 말뭉치(corpus)란 통계적 가치를 가질 수 있는 언어 자료를 효용이 있도록 접근이 쉬운 형태로 모아 놓은 것을 말한다.

따라 달라진다[6]. 예를 들어, 한국어 구문 분석의 전단계로 쓰이는 형태소 분석기는 어절을 구성하고 있는 모든 형태소에 대하여 가능한 모든 분석 결과를 출력해야 하며, 자동 인덱싱(automatic indexing)이나 정보 검색 시스템에 사용될 형태소 분석기는 명사구에 대한 분석 결과만을 출력하면 된다. 그러나, 철자 검사에 사용될 형태소 분석기는 입력된 각 어절들을 모두 분석하되, 형태소 분석의 목적이 구문 분석이나 의미 분석을 위한 정보 취득에 있는 것이 아니라 어절 내의 오인식 문자를 검출하기 위한 것이므로 한국어 분석에 사용된 형태소 분석기와는 그 기능에서 다소 차이를 보인다.

첫째로 모호성이 내포된 단어나 어절에 대한 모든 분석이 반드시 필요하지 않다. 예를 들어 "나는", "감기는" 등의 어절이 전체 문장에서 어떤 의미로 분석되어야 옳은가를 고려하여 "감(동사)+기+는", "감기(명사)+는", "감기(동사)+는"의 모든 분석 결과를 출력하기 보다는 한 어절을 구성하고 있는 문자들 간의 결합이 어색하지 않고 합당한가를 분석해야 한다.

둘째로 한국어의 조사는 그 형태가 고정되어 있고 거의 모든 체언과 결합이 가능하며 격 정보를 가진다. 조사는 앞 형태소와 결합할 때 앞 형태소의 종성 여부에 따라 같은 격 정보를 가진 다른 형태가 결합되어야 하는 데 예를 들면, "은/는", "이/가" 혹은 "을/를" 등이 있다. 하지만 대부분의 한국어 분석을 위한 형태소 분석기에서는 조사와 결합된 어절이 구문 분석 단계에서 적절한 격 정보를 가지게 하기 위하여 결합된 체언에 이들의 격 정보만을 부여해 준다. 따라서, 어절 "영희을"을 분석하여 "영희(명사)+을(목적격 조사)"로 출력하더라도 구문 분석에서 어떠한 장애도 일으키지 않지만 문자 교정을 위한 형태소 분석에서는 이를 오류로 판정하여 교정을 해야 한다.

특히 본 논문에서 구현한 형태소 분석기는 후처리 시스템의 효율을 고려하여 뭉치어구에 대한 형태소 분석 시간을 줄였다. 예를 들어서, "-에서부터라도", "그럼에도 불구하고", "-만에 하나" 등과 같은 부사 상당어구 혹은 전문 용어들의 결합은 그 형태가 고정되어 크게 변화하지 않는다. 이러한 형태의 어절이나 구를 뭉치어구로 정의하여 확장된 어절 형태를 그대로 사전에 등재하면 불필요한 형태소 분석을 줄일 수 있다. "그럼에도 불구하고"를 형태소 분석하면, "그럼+에도"와 "불구하+고"의 두 어절을 따로 분석해야 하는데, 이를 분석하더라도 결국 이 두 어절은 떨어져서는 별 의미가 없으며 늘 결합된 형태 그대로 사용된다. 그러므로 이러한 종류의 어절은 사전에 그대로 등재해 두었다가 형태소 분석 단계에서 사전의 뭉치어구 정보를 참조하여 한 어절로 취급한다면, 어휘나 조사, 어미 사전을 여러번 탐색해야 하는 번거로움을 피할 수 있고, 형태소 분석을 수행할 어절의 수가 줄어듦으로써 수행 속도도 높일 수 있다. 후처리에 걸리는 시간은 형태소 분석 단계와 생성된 후보 문자의 어절 수에 상당한 영향을 받으므로 형태소 분석 단계에서 입력 문장에 대한 분석 어절을 줄인다면 전체 시스템 수행에 효과를 줄 수 있다.

따라서 본 논문에서는 위에서 언급한 문제들을 고려하여, 한국어의 특성을 반영하면서 실제 예문을 통하여 얻어진 고정된 형태의 뭉치어구를 사전에 등재시켜 불필요한 분석을 줄였고, 문자 교정에 필요한 정보를 얻을 수 있는 문자 인식 후처리를 위한 형태소 분석기를 구현하였다.

2. 형태소 분석기

본 절에서는 형태소 분석기에 사용된 사전의 종류와 각 사전에 수록된 정보 및 형태소 분석 알고리즘을 소개하고, 문자 교정기에 넘겨 줄 오인식 어절의 출력 정보와 오인식 문자의 검출 방법을 기술하고자 한다.

1) 사전의 구조와 종류

본 시스템에서는 사전 검색을 용이하게 하기 위해 사전을 트라이(TRIE) 구조로 읽어 들인다. 트라이 구조는 다양한 가변 길이 자료들을 쉽게 처리하고, 사전의 검색 속도가 인덱스를 탐색하는 속도만큼 신속한 유용한 자료 구조이다. 특히, 단어를 좌에서 우로 먼저 분석하는 알고리즘에서는 분석 후보를 따로 생성하지 않으며 입력 단어와 사전을 형태론적 변형 규칙에 따라 일치시켜 나갈 수 있으므로 트라이 구조 사전을 사용하는 것이 좋다[6].

트라이 구조 사전에서 얻을 수 있는 또 다른 잇점은, 임의의 단어가 다른 단어에 내포될 때 최장 단어만 사전에 등재해 두고 내포된 단어의 끝 자소에 독립 단어로 쓰일 수 있다는 표시만 해 두면 하나의

단어 링크를 탐색하는 경로로 다른 단어까지 탐색할 수 있다. 또한, 첫글자의 초성이나 문자가 같은 "감, 감각, 감각, 감동, ..." 등의 경우, 같은 링크를 이용하므로 기억 공간의 효율을 높일 수 있다.

그림 1은 시스템에서 사용하고 있는 사전 중에서 어휘 사전을 트라이 구조로 나타낸 한 예로 기호 "#" 은 종성이 없는 문자의 종성 표시이고, "∧"은 다른 단어에 내포된 단어의 끝을 알리기 위한 종료 기호이다. 그림 1에서는 어휘 "가"와 특히, "형태소"의 경로상에 "형"과 "형태"가 내포되어 있음을 보여주고 있다. 또한, 뭉치어구가 두 어절 이상이 결합된 경우(띄어 쓰기에 의한 공백이 있는 경우)에도 사용하지 않는 자소 링크에 그에 대한 정보를 넣어 처리를 가능하게 하였다.

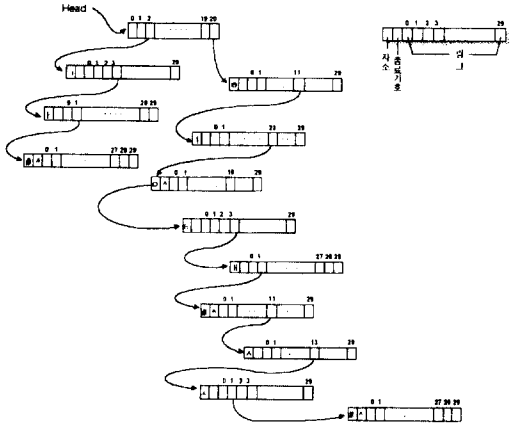


그림 1. 트라이 사전 구조
Fig. 1. TRIE dictionary structure.

형태소 분석기에서 사용하는 사전을 어휘 사전, 조사 사전, 어미 사전 그리고 접미사 사전으로 구분하였다. 어휘 사전은 일반적인 체언, 용언류 외에도 형태소 분석의 효율을 높이기 위한 고정된 형태의 어절들 즉 부사 상당어구, 복합어 등의 뭉치어구를 포함한 확장된 의미의 어휘 사전이다. 조사는 체언류와 결합하며 조사 사전에는 결합할 체언의 마지막 음절에 대한 종성 여부만을 등재하고 접미사 사전에는 약 80여개의 접미사가 등재되어 있다. 그리고 어미는 선어말 어미와 그외의 어미로 구분하여 사전을 구성하였으며, 보조 용언의 처리는 보조 용언 테이블을 이용하였다. 특히, 2.2.2절에서 소개할 형태소 분석 알고리즘을 위하여 조사와 어미 사전은 역순으로 재구성하였다. 그림 2에서는 사전과 등재된 구성 정보들을 보여 준다.

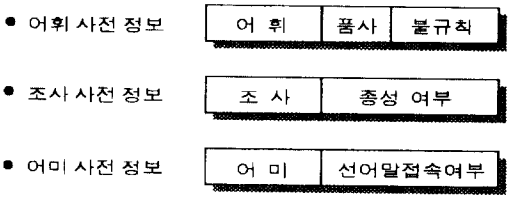


그림 2. 어휘, 조사, 어미 사전 정보
Fig. 2. Information of Lexical, Postposition, Verb-ending dictionary

2) 형태소 분석 알고리즘

우리나라에서 형태소 분석은 임의의 단어들이 결합하여 문법에 맞는 어절을 구성하는지에 대한 검사를 한다[9]. 본 시스템에서는 인식된 입력 문장에 대해 형태소 분석을 이용하여 오인식 문자를 포함한 어절을 검출하고, 다음 단계에서 생성된 후보 문자들로 재구성된 어절을 다시 형태소 분석하여 구조적으로 쓰임이 불가능한 후보 문자를 제거한다. 형태소 분석 알고리즘은 그림 3과 같다.

```
Algorithm Morph_Analy(buffer)
{
     체언/용언 분리 루틴();
     조사/어미 분리 루틴();
     잔류 어절 처리 루틴();
}

체언/용언 분리 루틴()
     {
     어휘사전 탐색 (LP 저장)
     if(탐색한 자소의 끝정보=='∧'){
        if(어휘가 용언의 어간) break;
        else if(탐색성공 어휘의 link[0] != NULL)
          체언/용언분리 루틴(); // 뭉치어구 //
        else return;
} }

조사/어미분리 루틴()
{
     if(분석된 어휘== 용언){
          어미사전 탐색(RP 저장);
          if(LP+1== RP) return SUCCESS;
          else return;
     }
     else{
          조사사전 탐색(RP 저장);
          if(LP+1 == RP) {
               if(체언의 종성 == 조사 종성 정보)
                 return SUCCESS;
               else return(FAIL_1);
```

```
            }
            else{
                어미사전 탐색(LP 저장);
                if(LP + 1 == RP) return SUCCESS;
            }
    } }
    잔류 어절 처리 루틴()
    {
        PROCESS();   // 미분석 잔류 어절 처리 단계 //
        if(LP +1 == RP) return SUCCESS;
        else return(FAIL);
    }
```

그림 3. 형태소 분석 알고리즘
Fig. 3. Morphological Analysis Algorithm.

먼저 형태소 분석은 입력 어절에 대해 왼쪽에서 오른쪽으로 분석하면서 체언과 용언의 어간을 분리하는 순방향 분석을 행하고 다음으로 조사와 어미의 분리를 위해 오른쪽에서 왼쪽으로 분석하는 역방향 분석을 행한다. 순방향 분석을 먼저 하는 이유는 사전에 등재된 정보를 이용해 어간, 어미를 보다 쉽게 분리하고자 하는 의도 외에도 어절의 분석이 큰 의미가 없는 뭉치어구를 형태소 분석을 하지 않고 사전 탐색만으로 사전에 등재된 어절과 입력 어절을 최장 일치시키기 위해서이다. 뭉치어구는 한 어절뿐만 아니라 여러 어절로 구성된 구로까지 확장시켜 처리할 수 있다. 즉 트라이 링크 중 어절 연결 링크가 널(null)이 아닐 경우에 다음 어절을 입력 버퍼로 읽어와서 사전과 뭉치어구를 최장 일치시킨다.

순방향 분석에서 체언은 사전을 탐색하면서 종료 기호를 가진 문자열을 찾아 최장 일치시키고, 최장 일치된 위치를 저장하여 조사, 어미 사전의 탐색에 이용한다. 불규칙 용언은 어미와 결합할 때 어간과 어미의 형태가 변화할 수 있으므로 입력 문자열에 대하여 사전 탐색을 시행한 결과 종료 기호를 만나지 못하는 경우가 발생한다. 이러한 경우에는 현재까지 탐색한 위치, 즉 최장 일치되는 버퍼의 자소 위치를 LP(왼쪽 위치 포인터)에 저장하고, 역방향 분석을 실시한다. 역방향 분석은 조사, 어미의 역순 사전 탐색으로 이루어지며 이는 선어말 어미의 처리와 활용된 용언의 어간, 어미의 처리를 용이하게 한다.

조사의 역방향 분석 후 조사 사전에서 최장 일치된 RP(오른쪽 위치 포인터)와 순방향 최장 일치된 LP가 RP = LP + 1 의 조건을 만족하면 조사의 종성 정보와 체언의 종성 여부를 조사하고 이 조건이 만족되면 분석을 종료한다. 어미의 역방향 분석은 불규칙 용언의 경우가 문제시 되는데, 이때는 어미 사전과 입력 어절을 역방향으로 최장 일치 시킨 위치 포인터 RP와 용언 분석에서 얻어진 LP가 다음의 관계를 만족하는지를 조사하고 분석을 하지 못한 버퍼의 중간 문자열에 대해서 아래의 과정을 거친다. (그림 3의 알고리즘에서 *PROCESS()* 루틴에 해당한다.)

정의 1. RP - LP > 1 인 경우, 다음 문자열 Y를 미분석 잔류 어절이라 정의하고, 아래의 순서 로 분석을 시도한다.

buffer [1:n] 내의 문자열 X 에서

$$X = X_1X_2 \cdots X_i\ X_{i+1} \cdots X_{j-1}\ X_j \cdots X_n \quad (1 \le i < j \le n)$$

$LP(=i)$ $RP(=j)$

$$Y = X_{i+1} \cdots X_{j-1}$$

1. 준말 처리 루틴 (준말에 의해 변형된 중성을 복구하여 어간, 어미 분리)
2. 불규칙 처리 루틴 (음절 정보를 이용해 불규칙 유형 추정, 어간과 어미를 원형 복구,분리)
3. 선어말 처리 루틴 (선어말 어미 존재 여부에 따른 어간, 어미 분리)
4. 복합 명사 처리 루틴
5. 보조 용언 처리 루틴 (보조 용언 테이블 이용)

위의 과정을 수행한 후에도 미분석 잔류 문자열이 남아 있으면, 현재 분석 중인 어절을 오인식 어절로 판정한다. 오인식 어절이 발견되면 현재 *LP*와 *RP*의 위치를 저장한 후 인접한 다음 문자의 초성 자소로 포인터를 옮겨 분석을 재시도한다. 이는 오인식 문자의 발생 범위를 탐색에 실패한 *LP* 혹은 *RP*의 인접부로 추정하여 후보 문자의 생성을 효과적로 하기 위함이다. 오인식 어절을 발견하면 오인식 어절과 *LP, RP*값을 이용한 오인식 추정 위치(*EL, ER*)을 출력하는데 이는 교정 후보 문자의 생성시에 불필요한 후보를 최대한 줄일 수 있다. 분석에 실패한 어절 *X*는 오인식 추정 위치 *EL, ER*에서부터 교정을 시도한다. 자소 대치 후에는 형태소 분석을 재시

도한다.

정의 2. LP와 RP가 다음과 같을 때, 오인식 자소의 추정 위치를 EL, ER이라 한다.

$$X = X_1X_2 \cdots X_l\ X_{l+1} \cdots X_{m-1}\ X_m \cdots X_n \quad (1 \le l < m \le n)$$
$$LP(=l) \qquad\qquad RP(=m)$$

$$Y = X_{l+1} \cdots X_{m-1}$$
$$EL = X_{l+1}\ (=LP+1) \quad ER = X_{m-1}\ (=RP-1)$$

EL이나 ER위치의 자소를 대치한 후에도 형태소 분석이 여전히 실패할 경우에는 이미 사전 탐색이 성공한 음절에서 오류가 있다고 판단하고 $(EL-1)$이나 $(ER+1)$로 교정 위치를 확대시켜 처리를 계속한다.

다음 장에서는 형태소 분석기에서 검출한 오인식 어절에 대한 교정 후보들의 생성 방법과 최적 후보 선택 과정을 설명하겠다.

Ⅲ. 문자 교정기

이 장에서는 2장의 형태소 분석기에서 출력된 정보들을 이용하여 후보 문자를 생성하는 방법과 최적 후보를 선택에 이용하는 신뢰도 분석 방법에 대하여 기술하고자 한다. 본 논문에서는 오인식 문자의 교정을 위하여 초등학교 교과서, 동아 일보 사설 등의 말뭉치(약 67만 어절)를 현재 상용중인 문서 인식기를 통하여 인식시켰다. 인식 결과 화일에서 문자의 인식이 완전히 실패한 경우를 제외한 약 20만 어절로부터 오인식 문자의 교정을 위한 자소 대치 테이블을 구성하였으며, 말뭉치와 유재원의 역순 사전[11]을 이용해 후보 문자의 선택시에 사용하는 가중치 정보를 확률 테이블로 구성하였다.

1. 문자 대치 테이블을 이용한 교정 후보 생성 방법

문자 인식 시스템은 어떤 인식 알고리즘이 적용되었고, 어떠한 특징을 가졌는지에 따라 조금의 차이를 가지지만 대부분 비슷한 오인식 결과를 나타낸다. 한글 맞춤법 교정에서의 오류는 사람의 타이핑에 의한 철자 오류인데 반해 문자 인식에서 발생하는 오류는 객관적인 구조적 유사 문자에 의한 오인식이 많다는 것을 알 수 있다. 따라서 문자의 구조적 유사성에 따른 교정은 자소 및 문자 대치 테이블을 이용하여 교정을 시도하며 여기서 이용하는 자소 및 문자 대치 테이블은 인식된 말뭉치에서 조사된 오인식 어절들을 이용해 구성하였다.

한글 문자는 자음과 모음의 이차원적 조합으로 생성되어지므로[10], 자소의 형태를 분석할 때 일차원 구조의 언어들에 비해(영어...) 훨씬 복잡하다. 그러므로 한글은 초성, 중성, 종성의 한 자소에 대한 형태 분석 뿐만 아니라 이들의 조합에 의해 발생하는 형태의 분석도 이루어져야한다. 따라서, 본 논문은 이를 고려하여 한 문자에 대한 자소 대치를 6가지 조합으로 처리하였으며 문자 단위의 오인식을 조사하여 구성한 문자 대치 테이블을 별도로 이용한다. 이들 중 우선 순위가 가장 높은 문자 대치 테이블은 표 1과 같다.

표 1. 문자 대치 테이블
Table 1. Character substitution table.

들 ⇔ 풀	골 ⇔ 끌	경 ⇔ 정	계 ⇔ 패	학 ⇔ 확
졌 ⇔ 겼	비 ⇔ 네	커 ⇔ 귀	끗 ⇔ 곳	레 ⇔ 컷
겼 ⇔ 젔	띠 ⇔ 며	키 ⇔ 거	깔 ⇔ 간	끝 ⇔ 곧

다음으로 자소 대치 테이블을 살펴 보자. 자소 대치 교정에서는 한글 문자의 조합, 즉 초성, 중성, 종성, 초성+중성, 초성+종성, 중성+종성의 경우를 고려하여 서로 다른 대치 후보를 생성하여야 한다. 이 6가지의 경우 중에서 표 2에서 표 4는 초성, 종성, 중성에 대한 자소 대치 테이블의 일부를 나타내고 있다.

표 2. 초성 대치 테이블
Table 2. First consonant substitution table.

	ㄱ	ㄴ	ㄷ	ㄹ	ㅁ	ㅂ	ㅅ	ㅇ	ㅈ	ㅊ	.....
1	ㅈ	ㄹ	ㄴ	ㄷ	ㅂ	ㅁ	ㅈ	ㅁ	ㄱ	ㅈ	
2	ㄲ	ㄱ	ㄹ	ㅂ	ㅇ	ㄸ	ㄴ	ㄷ	ㅊ	ㄷ	
3	ㅉ	ㄷ	ㅍ	ㅁ	ㄹ	ㅎ	ㅉ	ㄹ		ㅉ	
4	ㅅ	ㄲ	ㅎ	ㅌ	ㅌ	ㄹ	ㅂ	ㅎ			
5	ㅍ	ㅎ		ㄴ		ㅇ	ㅃ				
6	ㅎ						ㅊ				

표 3. 종성 대치 테이블
Table 3. Final consonant substitution table.

	.....	ㅅ	ㅇ	ㅈ	ㅊ	ㅌ	ㅍ	ㅎ	ㅆ	ㄵ
1		ㅆ	*	ㄱ	ㅈ	ㄴ	ㄹ	ㅍ	ㄲ	ㄹ
2		ㄱ	ㄹ	ㅅ		ㄷ	ㅂ	*	ㅂ	
3				ㄴ					ㅅ	
4									*	
5									ㄴ	
6										

표 4. 중성 대치 테이블
Table 4. Medial consonant substitution table.

	ㅏ	ㅑ	ㅓ	ㅕ	....	ㅡ	ㅣ	ㅐ	ㅔ	....	ㅘ	ㅙ	ㅚ	ㅝ
1	ㅑ	ㅏ	ㅣ	ㅓ		ㅗ	ㅕ	ㅖ	ㅐ		ㅙ	ㅚ	ㅘ	ㅞ
2	ㅐ	ㅣ	ㅏ	ㅣ		ㅜ	ㅏ	ㅖ	ㅓ		ㅚ		ㅙ	ㅜ
3	ㅣ	ㅓ	ㅕ	ㅜ		ㅛ	ㅐ	ㅚ	ㅖ				ㅏ	ㅐ
4	ㅘ	ㅘ	ㅜ			ㅏ	ㅓ	ㅗ	ㅣ				ㅓ	
5			ㅚ				ㅖ							
6														

다음으로 자소 분리 테이블은 두 자소의 조합이 한 자소로 오인식 되는 경우를 조사하여 아래와 같이 구성하였다.

표 5. 자소 분리 테이블
Table 5. Grapheme separating table.

오인식	자소 분리
ㅈ	ㄱ + ㅖ
ㅎ	ㅂ + ㅕ
ㄸ	ㅎ + ㅐ
ㅃ	ㅅ + ㅕ
...	...

오인식	자소 분리
ㅣ	ㅣ + ㄱ
ㅏ	ㅏ + ㄱ
ㅑ	ㅏ + ㄱ
ㅖ	ㅐ + ㄴ
...	...

오인식	자소 분리
ㅗ	ㅇ + ㄴ
ㄹ	ㄱ + ㄴ
ㅂ	ㅁ + ㄴ
ㄸ	ㅋ + ㄷ
...	...

위에서 언급한 테이블들을 이용하여 오인식 문자의 교정 후보를 생성하면 많은 수의 후보가 생성된다. 본 시스템에서는 생성된 후보 문자를 오인식 어절에 대치시켜 형태소 분석을 재시도하는데 이때 후보 문자의 수는 후처리 전체 시스템의 처리 시간에 상당한 영향을 주게 되므로, 실제 거의 사용되지 않거나 문자 구성상 결합이 불가능한 후보 문자의 태깅을 위하여 한글 오토마타를 사용한다. 후보들 중에서 태깅 정보가 없는 문자들만을 재분석하여 처리 시간을 단축하고자 한다. 하지만 오인식 어절을 후보 문자로 대치하여 재분석하더라도 선택되는 후보 문자의 수는 대부분 하나 이상이므로 이들 사이에 우선 순위를 두어 가장 적절한 교정 후보 문자를 선택해야 한다. 따라서 선택된 후보들에 대한 신뢰도 분석이 필요하다.

2. 생성 후보의 신뢰도 분석

한글의 문자 구성 특성 중의 하나인 자소들의 결합 즉, 초성, 중성, 종성 간의 결합 상태는 각 자소들끼리 연관성을 가지고 있다. 이러한 관계를 이용하여 한 자소와 나머지 자소들과의 결합 정도를 조건부 확률로 표현할 수 있다. 또한, 실제 사용되고 있는 문자들을 살펴볼 때, 이러한 확률값은 한 문자에서 그 문자를 구성하고 있는 각 자소들의 신뢰도를 분석하는데 유용하게 사용될 수 있다. 즉 자소 대치에 의한 교정 후보의 생성에 있어 구조적 유사성을 바탕으로 한 대치가 우선적이지만 문자는 자소들의 결합으로 이루어지므로 자소의 발생 환경을 고려하여야 한다. 즉 초성을 대치하는 경우에 이미 문자를 구성하고 있는 중성과 종성에 의해 영향을 받게 되고, 중성은 초성과 종성에 의해, 그리고 종성은 초성과 중성에 의해 영향을 받기 때문에 각각 대치 가능한 자소가 제한될 수 있으며, 각 자소들의 확률치 합이 클수록 사용 빈도수에 의한 경험치가 많으므로 우선 순위를 높게 설정할 수 있다.

본 논문에서는 생성 문자에 대한 최종 후보 문자의 선택에 사용할 신뢰도 분석을 위하여 유재원의 역순 사전에 등재된 1,548개의 한글 음절의 빈도수와 박안기의 자소별 사용빈도수 그리고 초등학교 교과서와 동아 일보 사설 등의 말뭉치를 이용하여 한 음절에서의 초성, 중성, 종성의 발생확률을 다음과 같은 조건부 확률로 계산하였다.

초성, 중성, 종성을 표시하는 확률 변수를 X, Y, Z라 정의하고, 이 때 초성 X의 발생확률을 $P(X)$라 하고, 중성 Y의 발생확률을 $P(Y)$, 종성 Z의 발생확률을 $P(Z)$라고 정의하자. 음절 X,Y,Z의 발생확률 $P(X,Y,Z)$는 전체 말뭉치에서 나타난 음절별 빈도수에 의한 확률이며, 초성, 중성, 종성의 발생확률은 $P(X,Y,Z)$를 이용하여 다음과 같은 방법으로 구한다. 같은 방법으로 자소간 결합 확률도 구할 수 있

다.(식(1), (2), (3)에서 a, b, c는 초성 자음, 종성 자음, 중성 모음의 개수를 나타낸다.)

$$\begin{aligned} P(X) &= \sum_{Y=1}^{b}\sum_{Z=1}^{c} P(X,Y,Z) \\ P(Y) &= \sum_{X=1}^{a}\sum_{Z=1}^{c} P(X,Y,Z) \\ P(Z) &= \sum_{X=1}^{a}\sum_{Y=1}^{b} P(X,Y,Z) \end{aligned} \tag{1}$$

$$\begin{aligned} P(X,Y) &= \sum_{Z=1}^{c} P(X,Y,Z) \\ P(X,Z) &= \sum_{Y=1}^{b} P(X,Y,Z) \\ P(Y,Z) &= \sum_{X=1}^{a} P(X,Y,Z) \end{aligned} \tag{2}$$

그리고, 조건부 확률 $P(X|Y,Z)$은 중성과 종성이 Y, Z일 때 초성이 X일 확률, $P(Y|X,Z)$은 초성과 종성이 X, Z일 때 중성이 Y일 확률, $P(Z|X,Y)$은 초성과 중성이 X, Y일 때 종성이 Z일 확률을 나타내며 각각의 조건부 확률은 다음의 식으로 구한다.

$$P(X|Y,Z) = \frac{P(X,Y,Z)}{P(Y,Z)} = \frac{P(X,Y,Z)}{\sum_{X=1}^{a} P(X,Y,Z)}$$

$$P(Y|X,Z) = \frac{P(X,Y,Z)}{P(X,Z)} = \frac{P(X,Y,Z)}{\sum_{Y=1}^{b} P(X,Y,Z)} \tag{3}$$

$$P(Z|X,Y) = \frac{P(X,Y,Z)}{P(X,Y)} = \frac{P(X,Y,Z)}{\sum_{Z=1}^{c} P(X,Y,Z)}$$

위의 식에 의해 계산된 조건부 확률표는 다음 장의 실험 결과에 적용시켜 소개하고자 한다.

마지막으로 음절(X,Y,Z)로 구성된 문자를 C라 할 때, 신뢰도 $Q(C)$는 다음과 같이 구한다.

$$\begin{aligned} Q(C) &= P(X)+P(X|Y,Z)+P(Y)+P(Y|X,Z) \\ &\quad +P(Z)+P(Z|X,Y) \end{aligned} \tag{4}$$

최종 교정 후보는 후보 문자들 중에서 $Q(C)$를 최대로 하는 후보로 선택한다.

Ⅳ. 시스템 구성과 실험 결과

1. 전체 시스템의 구성

본 논문에서 제시한 후처리 시스템은 그림 4와 같다. 영상 화일로 입력된 원문과 인식 결과 그리고 후처리 결과를 비교하여 인식률과 교정률을 측정하도록 하며, 후처리 시스템은 형태소 분석에 의한 오인식 어절 검출과 교정 후보의 생성, 최종 후보 선택의 3단계로 구성된다.

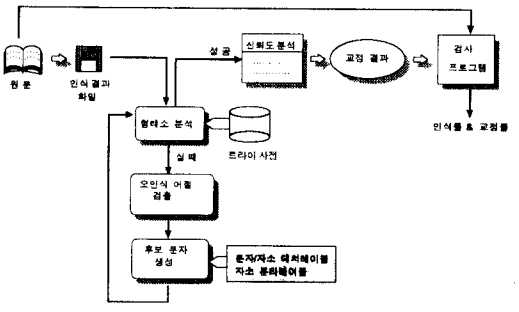


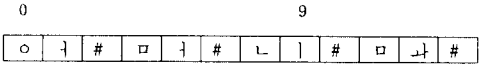
그림 4. 후처리 시스템 구조
Fig. 4. Structure of Post-processing System.

2. 실험 결과

[예문] 사랑하던 어머니와 헤어진 레미는 할아버지와 함께 많은 고생을 한다. 레미는 어리지만 하프를 연주해 주고 돈을 버는 꿋꿋한 소년이었다. 그래서 할아버지는 레미를 무척이나 사랑하셨다. 그리고 레미 자신보다 데리고 다니는 짐승을 항상 먼저 걱정하는 마음씨는 너무나 착하고 아름다웠다.

[인식 결과] 사랑하던 어머니와 헤어진 레미는 할아버지와 함께 많은 고생을 한다. 레미는 어리지만 하프를 연주해 주고 돈을 버는 꿋꿋한 소년이었다. ... 레미 자신보다 데리고 다니는 짐승을 항상 먼저 걱정하는 마음씨은 너무나 착하고 아름다있다.

- 검출된 오인식 어절과 출력 정보의 예
  ① 검출된 오인식 어절
  [어머니와], [함께], [돈을], [짐승을], [번저], [마음씨은], [아름다있다]
  ② 출력 정보의 예
  [어머니와]
  ⇒ buffer 상태



  ⇒ 출력 정보 : RP = 10, ER = 9
  [함께]
  ⇒ RP = 5, ER = 4 // 함(명사) + 께(?) //
  ⇒ LP = 3, EL = 4 // 함께(?) //

[점송을]

⇒ LP = 3, EL = 4

● 한글 오토마타를 거친 후보 문자

[봐] - (봐 0)(와 0)(뵈 0)(막 0)

[꼐] - (꼐 0)(껴 0)(깨 0)

[번저] - (번져 0)(번지 0)(번자 0)(먼저 0)(떤저 0)(빈저 0)(반저 0)(먼지 0)(말 0)(달 0)

(할 0)(랄 0)(일 0)(앨 0)(암 0)(안 0)(돈 0)(등 0)(돔 0)(둠 0)(는 0)(픈 0)(김 0)

(침 0)(짐 0)(잠 0)(감 0)(참 0)(정 0)(종 0)(농 0)(쏭 0)(승 0)(숭 0)(솜 0)

[아름다있다] - 아름다웠다

[마음씨은] - 마음씨는

● 형태소 분석과 신뢰도 분석에 의한 교정 후보의 선택

[봐] - 와

[꼐] - 깨

[마음씨은] - 마음씨는

[아름다있다] - 아름다웠다

[든을] - 등을, 돈을, 돔을

[번저] - 번져, 번지, 먼저, 떤져, 먼지, ...

[점송을] - 짐승을, 정승을, ...

● 자소 발생 확률과 조건부 확률을 이용한 신뢰도 분석 결과의 예

Q(등) = .0619 + .0053 + .0819 + .0060 + .0712 + .0318 = 0.2581

Q(돈) = .0619 + .4814 + .1076 + .6197 + .0612 + .2105 = 1.4423

Q(돔) = .0619 + .5077 + .1076 + .4120 + .0712 + .0021 = 1.1625

위의 계산식에서 보듯이, 오인식 어절 "든을"은 후보 문자들 중에서 확률값이 큰 "돈"을 가장 높은 우선 순위를 갖는 교정 후보로 선택한다.

● 교정 결과

사랑하던 어머니**와** 헤어진 레미는 할아버지와 **함께** 고생을 한다. 레미는 어리지만 하프를 연주해 주고 **돈을(돔을, 등을)** 버는 *곳곳한* 소년이었다. ... 레미 자신보다 데리고 다니는 **짐승을(정승을) 번지(먼저, 번저, 떤저,먼지)** 걱정하는 **마음씨는** 너무나 **아름다웠다.**

표 6은 신뢰도 분석에 사용된 자소 발생 확률표과 조건부 확률표의 일부분이다.

표 6. 자소 발생 확률과 조건부 발생 확률
Table 6. Grapheme Production Probability and Conditional Production Probability.

(A) 초성의 발생확률 $P(X)$

X	ㄱ	ㄲ	ㄴ	ㄷ	ㄸ	...	ㅊ	ㅌ	ㅍ	ㅎ
$P(X)$	.1272	.0387	.0784	.0619	.0233		.0411	.0165	.0235	.0455

(B) 중성의 발생확률 $P(Y)$

Y	ㅏ	ㅐ	ㅑ	...	ㅗ	ㅘ	...	ㅠ	ㅡ	ㅢ	ㅣ
$P(Y)$	.1523	.0576	.0278		.1076	.0213		.0333	.0819	.0021	.1379

(C) 종성의 발생확률 $P(Z)$

Z	공백	ㄱ	ㄲ	...	ㄹㄱ	ㄹㅁ	...	ㅆ	ㅇ	...	ㅎ
$P(Z)$	.5121	.0831	.0196		.0019	.0035		.0051	.0712		.0331

(D) 음절에서 초성의 조건부 확률 $P(X|Y,Z)$

Y	Z \ X	ㄱ	ㄲ	ㄴ	ㄷ	...	ㅅ	ㅇ	..	ㅍ	ㅎ
ㅏ	ㄱ	.0978	.0571	.0317	.1299	...	.0895	.0538	...	.0094	.0254
	ㄲ	.0012	.0210	.0287	.0323		.0054	.0036		.0196	--
	ㄳ	--	--	.0026	--		.0319	--		--	--
	:	:	:	:	:		:	:		:	:
ㅗ	ㄴ	.0198	.0211	.0973	.4814	...	.3174	.0712	...	.0014	.0417
	:	:	:	:	:		:	:		:	:
	ㅇ	.1977	.0162	.0193	.5077		.1680	.1314		.0129	.0180
	ㅈ	.0330	.0252	.0210	.0097		--	.0010		--	--
:	:	:	:	:	:	...	:	:	...	:	:
ㅡ	ㄴ	.3001	.0019	.6813	.0053	...	.0124	.8107		.0014	.0048
	:	:	:	:	:		:	:		:	:
	ㄹ	.0210	.0006	.7216	.1122		.0101	.0960		.0013	.0091
	:	:	:	:	:		:	:	...	:	:
	ㅇ	.0023	.0002	.0965	.5813		.0085	.0056		.0000	.0008
	:	:	:	:	:		:	:		:	:
:	:	:	:	:	:	...	:	:	...	:	:

3. 결과 분석

실험에서 인식 시스템의 인식 결과 120문자 중에서 110문자가 올바로 인식되어 평균적으로 92.8%의 인식률을 보였으며 본 시스템을 이용하여 교정한 결

과 인식률이 97.3%로 향상되었다. 오인식 어절 중에서 몇몇을 제외하고는 본 시스템에 의해 바른 교정이 됨을 볼 수 있다.

특히 후처리를 위한 형태소 분석기를 구현하여 문자 교정시에 형태소 정보를 최대한 이용하였으며 이는 후보 문자의 생성을 제한하여 교정 효율을 높이는데 도움을 주었다. 인식된 입력 문장내에 뭉치어구의 수가 많을수록 형태소 분석에 걸리는 시간은 줄어들며, 체언과 결합한 오인식된 조사의 교정과 오인식된 불규칙 용언의 활용부 교정은 형태소 정보를 이용하여 쉽게 처리할 수 있었다. "마음씨은"과 같은 오인식 어절은 종성 유무에 따른 정보를 이용하여 한 번의 시도로 해결되며, "아름다있"과 같이 불규칙 용언의 경우에도 "아름답"의 "ㅂ불규칙" 정보를 보고 용언의 어간과 어미 결합에서 "오/우"를 생성하여 보다 쉽게 교정 후보를 생성할 수 있다. 또한 자소 대치에 의해 생성된 후보 문자들을 태깅함으로써 불필요한 분석을 피할 수 있다.

하지만, 위의 실험 결과와 같이 오인식된 형태가 사전에 존재하여 형태소 분석에 성공한다면 오인식 단어의 검출이 불가능하다. "곳곳한"의 경우 사전에 있는 등록어이므로 인식이 바르게 된 것으로 인정된다. 다음으로는 "번저"의 경우 "먼저"가 후보 문자에 있긴 하지만 확률에 의해 "번지"가 교정 후보로 선택되는데, 이것은 의미 분석까지 해 보아야 실제 오류임을 알 수 있으므로 여기서는 올바른 교정이 어렵다. 하지만 신뢰도 분석은 후보 문자가 둘 이상일 경우, 가중치를 두어 우선 순위를 정하는 데 의미가 있다. 또 다른 문제로 한 어절내에서 오인식 문자의 위치가 두군데 이상이거나 첫 음절이 오인식된 경우, 예를 들어 "짐승을"이 "점송을"과 같이 인식된 경우는 생성된 후보 문자의 수가 많고 분석의 재시도 횟수가 급증하므로 이에 대한 연구가 계속되어야 한다.

V. 결 론

본 논문에서는 문자 인식 시스템의 전체 인식률을 향상시키기 위하여 오인식 문자 교정을 위한 후처리 시스템을 구현하였다. 먼저 형태소 분석기가 입력 어절에서 오인식 문자를 검출하여 교정에 필요한 오인식 문자의 위치를 추정하고 이 정보를 출력한다. 이때 사용하는 형태소 분석기는 뭉치어구를 처리하여 불필요한 탐색 시간을 줄이고 오인식 문자의 위치를 추정하므로 교정 후보 생성에 효율적이다. 오인식 문자가 발견되면 교정기는 한글 자소의 구조적 유사성을 반영한 자소 대치 테이블, 문자 대치 테이블, 자소 분리 테이블을 이용하여 오인식 문자의 후보를 생성한다. 생성된 후보 문자 중에서 쓰임이 거의 없거나 문자 구조상 생성이 불가능한 문자를 한글 오토마타를 이용하여 구분한 다음, 한국어 어절 구성법상 적절하지 않은 후보를 제거하기 위해 다시 형태소 분석을 행한다. 재분석에 성공한 교정 후보가 유일하지 않을 경우에는 문자 간의 신뢰도를 분석하여 우선 순위를 정하고 최종 교정 후보를 선택한다. 이러한 과정을 통한 오인식 문자의 교정은 구문에 적당한 후보만을 선택하게 되며, 신뢰도 분석으로 그 실용성을 평가받을 수 있다.

본 시스템에서는 한글 문서 인식기에서 발생하는 오인식 문자의 교정을 위해 한글의 문법적 특성을 고려한 새로운 형태소 분석기를 구현하였으며, 말뭉치와 사전을 통해 구해진 자소 간 조건부 발생 확률을 교정 후보 생성시에 이용하였다.

실험에 사용된 문서들은 HP ScanJet 4c 스캐너상에서 입력 받았으며, 상용화된 문서 인식기로 인식시켜 93%의 인식률을 보인 문서를 본 시스템으로 교정한 결과 97%로 인식률이 향상되었다. 이는 문서 인식 시스템의 전체 인식률 향상에 도움이 된다.

하지만 오인식 어절이 이미 사전에 등록된 형태로 나타나거나 형태소 분석에 성공한 경우는 오인식 어절로 검출되지 않아 올바른 교정이 어렵다. 즉 구문 오류가 되는 오인식 어절은 형태소 분석 뿐만 아니라 구문 분석, 의미 분석을 거쳐야 올바른 교정이 가능하므로 이에 대한 연구가 계속되어야 한다.

참 고 문 헌

[1] 민병우, "문자 인식을 위한 후처리 기법의 사례 연구", 충북대학교 전자계산학과 석사학위논문, 1993

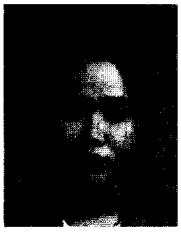
[2] 박진우, "통계적 방법에 의한 후처리", 연세대학교 전산과학과 석사학위논문, 1995

[3] 홍남희, 이원일, 이종혁, 이근배, "어절 정보와 문자열 정보를 이용한 문자 인식에서의 오인식 수정 기법에 관한 연구", 제 1회 문자인식

워크샵 발표 논문집, 충북대학교 컴퓨터 과학 연구소, pp. 109-113, 1993
[4] 이병훈, "말뭉치 분석을 기반으로 한 한국어 철자 교정기 구현", 연세대학교 전산과학과 석사학위논문, 1994
[5] 심철민, "어절 간 연관 관계와 오류 유형 추정 규칙에 기반한 한국어 철자 교정기", 부산대학교 전자계산학과 석사학위논문, 1995
[6] 강승식, "음절 정보와 복수어 단위 정보를 이용한 한국어 형태소 분석", 서울대학교 컴퓨터과학과 박사학위논문, 1993
[7] Thomas-N.Turba, "Checking for Spelling Typographical Errors in Computer Based Text", SIGPLAN Notices, pp. 298-312, 1981.
[8] Charniak, E., Statistical Language Learning, MIT Press, Cambridge, MA, 1993.
[9] 유진희, 이종혁, 이근배, "형태소 분석과 언어 평가를 이용한 문자 인식 후처리", 한국정보과학회 논문지 제22권 제6호, pp. 880-891, 1995
[10] 도영희, 황영섭, 방승양, "한글의 유사 문자", 제 1회 문자인식 워크샵 발표 논문집, 충북대학교 컴퓨터과학연구소, pp. 47-21, 1993
[11] 유재원, 우리말 역순사전, 서울정음사, 1985.
[12] 최재혁, "양방향 최장일치에 의한 형태소 분석기의 구현", 경북대학교 전자공학과 박사학위논문, 1993
[13] 이성환, 문자인식 이론과 실제, 홍릉출판사, 서울, 1994
[14] R.L.Kashyap, B.J.Oommen, "Spelling correction using probabilistic methods", Pattern Recognition Letters, pp. 147-154, 1984.
[15] Yamashina, M. and Obashi, F., "Collocation Analysis in Japanese Text Input", COLING88 (Proceedings of the twelfth International Conference on Computational Linguistics), pp. 770-772, 1988.
[16] Sugimura, T., "Error Correction Method for Character Recognition Based on Confusion Matrix and Morphological Analysis", The Transactions of The Institute of Electronics, Information and Communication Engineers D-II, Vol. J72-D-II, no. 7, pp. 993-1000, 1989. (written in Japanese).
[17] R.W. Cormew, "A statistical method of spelling correction correction", Information and Control, Vol. 12, No. 2, Feb. 1968, pp. 79-93.

저 자 소 개

李 英 和(正會員)
1991년 경북대학교 컴퓨터공학과 졸업. 1993년 경북대학교 대학원 컴퓨터공학과 석사. 1996년 현재 경북대학교 대학원 컴퓨터공학과 박사과정. 관심분야는 자연어 처리, 문자 인식, 인공지능



金 桂 成(正會員)
1996년 부산여자대학교 전자계산학과 졸업. 1997년 현재 경북대학교 대학원 컴퓨터공학과 석사과정. 관심분야는 자연어 처리, 문자 인식

金 永 勳(正會員)
1988년 경북대학교 전자공학과 졸업. 1990년 경북대학교 대학원 컴퓨터공학과 석사. 1996년 현재 경북대학교 대학원 컴퓨터공학과 박사과정, 안동전문대학 전산과 조교수. 관심분야는 자연어 처리, 데이터 베이스, 문자 인식, 인공지능

李 相 祖(正會員) 第 33卷 B編 第 4號 參照
현재 경북대학교 컴퓨터공학과