최장일치를 이용한 구문 분석용 형태소 분석기

송연정, 이근용, 이용석
전북대학교 컴퓨터과학과 언어정보공학실
{yjsong, cypher}@cypher.chonbuk.ac.kr, yslee@moak.chonbuk.ac.kr

(Morphological Analyzer using Longest Match Method for Syntactic Analysis)

Y. J. Song, K. Y. Lee, Y. S. Lee
Dept. of Computer Science, Chonbuk National University

요 약

형태소 분석 단계는 자연어 처리 과정의 첫 번째 단계로써 주어진 입력 어절들에 대한 형태소들의 조합을 추출하는 일을 한다. 형태소 분석 시스템의 기본적인 기능은 매우 중요하여 적용되는 형태소 분석 알고리즘에 따라 형태소 분석 시스템의 성능에 영향을 미친다. 그러나 형태소 분석 시스템, 구문 분석 시스템 및 의미 분석 시스템이 연계되어 하나의 자연어 처리 시스템이 구축되는 관점에서는 구문분석 시스템의 부담을 줄여 전체 시스템의 효율을 향상시키기 위하여 구문 분석 시스템의 입력에 적합한 형태소 분석 결과를 생성해주는 일 또한 형태소 분석 시스템의 중요한 역할이라 할 수 있다. 본 시스템은 최장일치법을 이용한 형태소 분석 방법으로 입력 어절에 대한 형태소 분석을 수행하는 동안 분석 후보의 개수를 줄이고 사전 탐색 시간을 줄여준다. 또한 구문분석 시스템의 입력에 적절한 형태소 분석 결과를 생성하여 전체 응용 시스템의 효율성을 향상시킨다.

1. 서론

자연어 처리 시스템은 크게 형태소 분석, 구문 분석, 의미 분석의 세 단계로 구분되어 있다. 이 중 형태소 분석은 구문 분석과 의미 분석의 전단계로서 입력 어절들에 대한 단위 형태소들의 조합을 추출하는 일을 한다. 형태소 분석의 결과는 그 자체만으로도 이용되지만 일반적으로 기계 번역이나 정보 검색 등 모든 자연어 처리 분야에 필수적 이용되고 응용되는 분야에 따라 형태소 분석의 결과는 조금씩 달라지게 된다.

본 시스템은 주로 기계 번역과 정보 검색

분야에 응용을 목적으로 하고 있기 때문에 형태소 분석 시스템 자체에서의 효율성 뿐만 아니라 구문분석 시스템과의 연계과정에서의 효율성을 고려하여 두 개의 분석 모듈로 구성되었다. 기본 형태소 분석 모듈에서는 최장일치법을 사용하여 준말 처리, 불규칙 용언의 원형 복원 등의 일과 형태소 분석 후보의 과 생성 문제를 해결하고 사전 탐색 시간을 줄이기 위한 일 등의 다른 일반적인 형태소 분석 시스템에서 하는 역할을 담당하고 있으며 구문 형태소 모듈에서는 구문분석 시스템의 부담을 줄이기 위하여 기본 형태소 분석 모듈에서 생성한 형태소 분석 후보에 대해 여러 형태소가 결합하여 하나의 구문적 단위나 의미적 단위로 묶일 수 있는 경우 이들을 묶어 하나의 구문 형태소로 출력하여 구문분석 시스템의 입력에 적합한 형태소 분석 결과의 생성을 담당하고 있다. 즉, 구문 형태소 분석기는 형태소 분석 시스템의 후처리나 구문 분석 시스템의 전 처리에 해당한다. 본 형태소 분석 시스템은 응용 분야에 따라 기본 형태소 분석기의 결과를 즉시 이용할 수 도 있고 이 형태소 분석 결과를 구문 형태소 분석기를 거쳐 나온 결과를 이용할 수 도 있게 구현되어 있다.

본 논문에서는 형태소 분석 시스템의 기본 형태소 분석기의 기능에 대하여 주로 기술한다. 특히, 이 모듈에서 분석 후보의 과 생성 문제가 어떤 방법론을 사용하여 어느 정도 해결을 하고 있는지에 대하여 기술하고 구문 형태소 분석기에 대한 내용으로는 구문분석 시스템의 입력으로 어떤 결과를 제공하고 있는지에 대하여 기술한다.

본 논문의 구성은 2장에서 최장일치법을 사용하는 형태소 분석 시스템의 구조와 알고리즘을 설명하고 구문 분석 시스템의 부담을 줄이기 위한 구문 형태소 분석기의 기능을 설명한다.

3장에서는 MATEC의 테스트 어절에 대한 결과 파일에 대하여 본 논문의 형태소 분석기를 이용하여 분석했을 때의 차이점을 태그셋과 표제어 선정의 기본원칙을 중심으로 비교하고 설명하며, 본 논문의 형태소 분석기의 표제어 선정 관점으로 실험 및 평가를 한다.

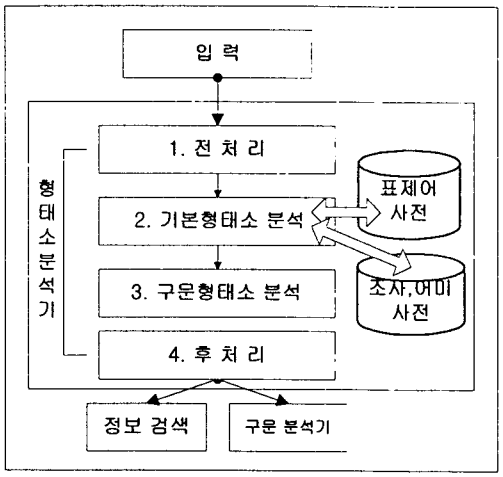
마지막으로 결론에서는 우리 형태소 분석기를 객관적으로 평가한 결과를 바탕으로 앞으로 개선해야 할 사항을 제시한다

2 시스템 구성 및 특성

2.1 시스템의 구성

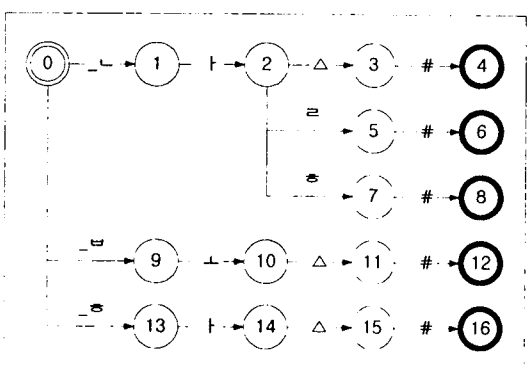
본 논문의 형태소 분석 시스템은 입력 어절들에 대한 단위 형태소들의 조합을 추출하는 기본 형태소 분석기와 구문분석 단계의 입력에 알맞은 형태로 결과를 생성해주는 구문 형태소 분석기 그리고 형태소 분석을 위한 전자 사전으로 구성되어 있다. 형태소 분석기의 출력은 정보 검색기와 구문 분석기와 같은 응용 프로그램의 입력으로 사용된다.

형태소 분석기를 이루는 구성 요소는 <그림 1>과 같이 영문자, 숫자, 특수기호와 같은 것들을 분리해 주는 전처리, 한국어 어절들을 분석하여 형태소들의 조합을 추출해내는 형태소 분석, 응용 분야에 따라 형태소 분석의 결과를 적당한 형식으로 변형해 주는 후처리로 나누어 볼 수 있다. 후처리는 형태소 분석기의 응용 범위에 따라 구문 분석의 입력을 생성하는 부분과 정보 검색의 입력을 생성하는 부분으로 구성되어 있다.



<그림 1> 시스템 구성

표제어를 구성하는 전자사전은 트라이 구조[김철수 98]를 사용하고 있으며, 표제의 선정은 "동아 새 국어 사전"[동아 94]의 표제어를 중심으로 구성되어 있다. 표제어 선정이나 품사 태그에 대한 부분은 2.5절에서 다루기로 한다. <그림 2>는 사전의 구축 형태를 나타낸 것이다.



<그림 2> 사전의 표현 형태

<그림 2>에서 초성과 종성의 형태가 같으므로, 초성은 앞 부분에 '_'를 붙여 구분하였다. 사전은 <그림 2>와 같은 상태의 전이로 표현된다. 예를 들어서, 상태 '0'에서 '_ㄴ'이 입력으로 주어지면, 상태 '1'로 전이한다고 볼 수 있다. 따라서, stack 구조나, 병렬처리 기법을 사용한다면, 한번 사전을 탐색하는 동안에 여러 경로에 대해서 탐색이 가능하다 [이근용 95]. 따라서, 전자 사전은 다음과 같이 정의한다.

전자 사전 = <Initial State, Final State, Input Set, State Set>

Initial State = 0

Final State={x:x=단어가 완성된 상태의 집합}

Input Set={한글 초성, 중성, 종성의 자소, #:EndMark}

State Set = {0,1,2,.....n}

형태소 분석을 수행하는 동안에 대량의 데이터를 전자 사전으로부터 얻어오게 되는데, 이를 유지하여 후처리에 전달해 주기 위해서 자료 구조는 {실질 형태소, 실질 형태소의 사전 정보, 형식 형태소, 형식 형태소의 정보}를 갖추고 있으며, 여러 개의 결과가 존재할 수 있으므로, 전체 형태소 분석의 결과는 최종적으로 트리 구조의 형태를 가지고 후처리의 입력으로 주어지게 된다.

2.2 전처리

형태소 분석기에서 전처리의 역할은 입력 어절로부터 한글을 제외한 나머지 부분을 처리하는 것이다.

예1> 1. 나는 하늘을 보았다.

예1>과 같은 문장이 입력으로 주어졌다고 하면, '1','.'과 '.'의 한글이 아닌 부분이 포함되어 있다. 이런, 한글이 아닌 부분에 대해서 미리 분석을 수행한다. 입력 어절에 포함되

어 있는 숫자나 심볼 뿐만 아니라, 영문자, 한자, 특수 문자들도 전처리 과정에서 미리 처리한다.

2.3 형태소 분석

형태소 분석에서 가장 문제가 되는 부분은 형태소의 원형을 복원하는 과정다. 우리가 구현한 형태소 분석기는 크게 세가지 특징을 갖는다.

첫째, 형태소 분석의 원형 복원에서 발생하는 오버헤드를 최소화하기 위해서, 트라이 형태로 구축된 전자 사전을 탐색하는 동안에 어절에 포함된 모든 원형을 추출할 수 있도록 하여, 원형을 파악하기 위해서 전자사전을 참조해야 하는 횟수를 줄였다. 이를 통해 전체적인 형태소 분석의 시간적 효율을 높였다.

둘째, 구문 해석이나 의미 해석 전에 하나의 단위로 결합하여 여러 응용 시스템에 효율적으로 이용될 수 있도록 구문 형태소 단위의 결합을 시도하였다. 형태소 분석의 단위를 한 어절이 아닌 다음 어절과의 관계를 고려하여 구문 단위 형태소로 결합하여 구문 해석의 입력 단위로 이용함으로써 구문 해석 과정을 간략화 시킬 수 있었다.

셋째, 복합 명사처리를 확장하여, "명사+명사" 구조나, "명사+용언"의 구조도 모두 복합 명사와 같은 방법으로 분석할 수 있도록 하였다. 복합 명사처럼 분석하는 두 번째 경우인 "명사+용언"의 형태는 "밥먹다"와 같은 경우인데, 흔히 띄어쓰기 오류를 범하는 형태로서, 띄어쓰기 오류에 대해서 부분적으로 대응할 수 있도록 하였다.

사전 탐색 동안에 원형 복원을 수행하는 구체적인 예를 들어 보면, <예1>의 첫번째 어절인 '나는'에 대해서 일반적으로 나타나는 분석후보는 다음과 같다.

1 나/N + 는/J
2 나/V + 는/E
3 날/V + 는/E
4 낳/V + 는/E
5 나느/V + ㄴ/E
6 나늘/V + ㄴ/E
7 나늫/V + ㄴ/E

위에서 나타나는 7개의 분석 후보에서 실제 형태소 분석 결과로 나타나는 것은 1,2와 3의 세가지 밖에 없다. 물론 사전 탐색 횟수를 줄이기 위한 여러 방법론이 있을 수 있겠지만, 위의 7가지의 분할이 이루어져 있다면, 사전 등록 여부를 확인하기 위해서 7번의 사전 탐색을 필요로 한다. 그러나 우리가 만든 형태소 분석기는 단 한번의 사전 탐색으로 위에서 정답이라 제시한 분석 후보를 추출해 낼 수 있다[이근용 95].다음은 <그림 2>에 제시한 사전 구성 예를 이용하여 후보 추출 과정을 도시하고 있다.

입력이 '나는'일 때, 자소로 나누어 보면, 'ㅣㄴ ㅏ ㅣㄴ ㅡ ㄴ'이 된다. 첫 입력인 'ㅣㄴ'이 입력되면, 사전에서 상태는 0→1로 전이 되고 다음 입력으로 'ㅏ'가 입력되면 상태는 2로 전이된다. 계속해서 위의 입력으로 전이되는 과정을 되풀이 하면, 0→1→2→3→4로 전이 되어 '나'가 인식됨을 알 수 있다. 한국어 원형의 굴절은 중성 또는 종성의 위치에서 일어난다. 따라서, 상태 전이를 위한 입력을 구할 때 주어진 입력 뿐만 아니라 중성에

서 나타날 수 있는 입력, 종성에서 나타날 수 있는 입력을 구하여 각각을 입력으로 주어 또 다른 상태로 진행시킬 수 있다. 즉, 상태 2에서 ' '를 입력으로 받아 3의 상태로 전이시키기 전에, 다른 가능한 입력을 구하는 것이다. 이 부분에서 원형 복원 규칙을 적용하여 'ㄹ'과 'ㅎ'을 다른 입력으로 얻어 낼 수 있다. 따라서

$\delta(2, \ ) = 6$
$\delta(2, ㄹ) = 5$
$\delta(2, ㅎ) = 7$

의 상태로 각각을 전이 시킬 수 있다. 또한 단어를 이루는 지를 알아 보기 위해서 #(endmark)를 입력으로 주어 Final State에 도달하는 지를 살펴본다.

$\delta(3, \#) = 4$
$\delta(5, \#) = 6$
$\delta(7, \#) = 8$

이와 같이 세개의 상태가 Final State에 도달하는데, $\delta(7, \#) = 8$의 경우는 사전정보의 불규칙 정보를 확인하여 삭제되는 후보이다. 분석 후보의 5, 6과 7의 경우는 3번 상태에서 입력으로 '_ㄴ'를 받아들이고 전이한 상태에서 중성 'ㅡ'를 받아 들여 전이를 시키는 '_ㄴ'로의 전이는 존재하지만, 중성 'ㅡ'에 대한 전이가 존재하지 않으므로 5, 6과 7의 분석 후보는 생성하지 않는다. 위의 예시에서 살펴 본 바와 같이 사전 탐색과 입력 어절의 원형을 찾는 과정이 동시에 이루어짐으로 한 어절에 대해서 한 번의 사전 검색으로 모든 분석 후보를 찾아 낼 수가 있다. 따라서, 형태소 분할을 미리 하여 사전을 탐색하는 경우 최소 7번의 사전 탐색이 필요하지만, 우리가 수행하는 알고리즘의 경우 단 1번의 사전 탐색으로 모든 후보를 찾아 낼 수 있어, 사전 탐색에 소요되는 시간을 줄일 수 있다. 위의 사전 탐색에서 병렬적인 부분은 stack을 이용하여 수행하고 있지만, 병렬 프로그램 기법을 이용하여 해결한다면, 현재 보다 좀 더 빠른 원형 복원을 할 수 있을 것이다.

한 어절 중심이 아닌 인접 어절과의 관계를 고려하여 분석함으로써 얻을 수 있는 장점은 형태소 분석 단계에서 발생하는 과생성 문제를 해결할 수 있다는 것이다. 여러 기능 형태소가 결합하여 하나의 의미나 문법 단위를 이루어 선행하는 용언이나 체언과 결합하는 형태소의 나열을 구문 형태소로 청의하여, 두개 이상의 어절이 하나의 단위로 묶일 수 있다면, 두개 이상의 어절에서 나타나는 형태소 분석 결과들 중에서 어절 사이에 서로 관련 있는 분석 결과만 남기고 나머지는 삭제할 수 있으므로, 다음 분석 단계로의 입력의 숫자를 줄일 수 있다.

이와 같은 방법은 일반적으로 태거[김재훈 96],[임희석 97]가 수행하는 방식과 유사하게 보일 수도 있지만, 태거는 품사모호성이 있는 단어에 대해 통계정보 등을 이용하여 단어에 대한 가장 적절한 품사를 선택하여 모호성을 제거하며, 우리는 두 어절 이상을 하나의 형태소 열로 결합하는 것이다. 인접 어절과의 관계를 고려하여 하나로 묶어 줄 수 있는 구문 형태소의 예로는 복합동사와 의사조사가 있다.

복합동사는 "먹어 보다"나 "먹어보다"와 같이 띄어쓰기에 관계없이 같은 분석 결과를 만들어 낼 수 있도록 하였다. 즉, "먹어 보다"와 같이 두 어절로 이루어 져 있어도, "먹어

보다"와 같이 한 어절로 이루어진 것과 같은 형태소 분석 결과를 만들어 낸다. 이 때 "나와 있었다"와 같은 표현은 두 어절을 묶어서 하나의 결과로 처리하면, "그는 나와 있었다"와 같이 '나와'의 '나'가 대명사일 때의 정보를 잃어버리게 된다. 이런 문제를 해결하기 위해서 '나와 있었다'와 같이 품사적으로 중의성이 발생하는 것은 두개의 결과를 모두 생성할 수 있도록 하였다[이기오 94].

의사조사 또한 조사가 의존명사에서 파생한 용언이나 일반 용언과 결합하여 문장 내에서 서술어로서의 기능을 수행하지 않고, 서로 띄어 쓰여진 어절이지만 하나로 묶여서 하나의 조사적인 의미를 갖는 경우를 말하는 것으로 "~에 대해서", "~를 위하여"와 같은 경우가 의사조사에 해당한다. 위의 예들은 두개 이상이 어절이 하나의 결과로 묶여 질 수 있는 것으로서, 다음 응용 프로그램에 불필요한 정보를 넘기지 않아도 되므로 구문 분석기의 부담을 줄여준다.

본 전자 사전은 '동아 새 국어 사전'을 기준으로 표제어를 수록하고 있으므로, '동아 새 국어 사전'에 등록되어 있지 않은 단어는 형태소 분석시 미등록어가 되어 분석이 실패한다. 따라서 형태소 분석기가 복합 명사의 범주를 확장하여 미등록된 표제어를 처리하고 있다. 복합 명사 범주에 들지 않는 것은 미등록어 처리를 한다. 다음과 같은 것들도 복합명사의 범주에 들어간다.

예2> 밥먹다 → 밥/N + 먹/V + 다/E

예3> 사회복지위원회
→ 사회복지/N + 위원회/N

예2>의 경우는 명사와 용언의 형태가 일반적으로 띄어 쓰여져야 하지만 흔히 붙여 쓰는 오류를 범할 수 있으므로 복합명사의 범주에 포함시켜 분석하였다. 예3>에서는 '사회', '복지',와 '사회복지'가 모두 사전의 개별 표제어로 등록되어 있지만, 최장일치에 의해서 '사회복지'만을 선택한 것이다.

2.4 후처리

후처리가 담당하는 부분은 형태소 분석의 결과를 받아 들여 다음 응용프로그램이 요구하는 입력 형식을 맞추어 주는 역할을 한다. 따라서, 형태소 분석기의 내용을 바꾸지 않아도 여러 응용 프로그램에 적절한 결과를 생성해 내도록 하는데 용이하다. 우리가 사용하는 시스템에서는 형태소 분석기의 응용범위에 따라, 하나는 구문 분석기의 입력 형식에 맞게 생성하는 부분과, 다른 하나는 정보 검색 시스템의 색인어로서 사용할 형식에 맞게 결과를 생성하는 부분으로 나뉘어져 있다.

이번 MATEC에 대한 결과를 만들기 위해서 태그간의 매핑 또한 이 후처리를 바꾸어서 이루어진 것이다.

2.5 사전의 구성 및 품사 태그의 특성

본 형태소 분석기에서 사용하는 표제어는 동아 새 국어 사전을 기준으로 작성하였다. 품사의 세분류 표는 <표 1>과 같다.

대분류	소분류	태그
명사	동작성 명사	Ncpa
	상태성 명사	Ncps
	비서술성 명사	Ncn
	고유명사	nq

	인칭 대명사	npp
	지시 대명사	npd
	양수사	nnc
	서수사	nno
	비단위성 의존명사	nbn
	단위성 의존명사	nbu
용언	자동사	pvgi
	타동사	pvgt
	지시동사	pvd
	보조용언	px
	성상형용사	paa
	지시형용사	pad
수식언	지시부사	mad
	일반부사	mag
	접속부사	maj
	성상관형사	mma
	지시관형사	mms
	수 관형사	mmc
독립어	감탄사	ii

<표 1> 품사 세분류 표

위에서 제시된 품사 태그 이외에 명사를 용언으로 파생시키는 접미사를 부착할 수 있는가의 여부를 나타내는 태그를 추가적으로 사용한다. 이때 사용하는 태그는 <표2>와 같다.

	설명	태그
명사	'하다'가 붙어 자동사로 파생	hi
	'하다'가 붙어 타동사로 파생	ht
	'하다'가 붙어 형용사로 파생	ha
	'되다'가 붙어 자동사로 파생	di
	'스럽다'가붙어형용사로 파생	sa

<표 2> 명사의 용언 파생 태그

위에서 제시한 태그들은 사전에 표제어로 등록되는 단어에 부착되는 품사 태그들이다.

이외에 실질 형태소와 결합하여 단어와 단어 사이의 관계나 기능을 표시하는 조사 사전과 어미 사전이 있다. 조사 사전과 어미 사전의 품사 분류표는 <표 3>이다.

조사 사전이나 어미 사전에는 앞의 실질 형태소와의 결합 관계를 표현하는 정보를 포함하고 있다.

대분류	소분류	태그
조사	주격 조사	jcs
	부사격 조사	jca
	보격 조사	jcc
	접속격 조사	jcj
	관형격 조사	jcm
	목적격 조사	jco
	공동격 조사	jct
	호격 조사	jcv
	서술격 조사	jp
	통용보조사	jxc
	종결보조사	jxf
	특수보조사	jxt
어미	대등적 연결어미	ecc
	인용적 연결어미	ecq
	종속적 연결어미	ecs
	관형사형 어미	etm
	명사형 어미	etn
	명령형 종결어미	inst
	감탄형 종결어미	exec
	의문형 종결어미	ques
	평서형 종결어미	dec
	청유형 종결어미	let

<표 3> 조사사전과 어미사전의 품사 태그

3. 실험 및 평가

우리가 구현한 형태소 분석기의 성능을 평가하기 위해서, MATEC에서 시험 데이터로 제시한 3만 어절의 말뭉치를 사용하였다. MATEC에서 제시한 정답 파일을 이용하여 재현률과 정확률을 구하기에는 다소 무리가 있었다.

우리의 형태소 분석기는 한국어에서 특히 형태소 과생성을 유발하는 복합 동사구와 의존 명사를 포함하는 어절에 대해 구문 형태소 단위의 처리를 한다. 그러므로, 우리가 개발한 형태소 분석기에서 채택하고 있는 형태소의 분할이 MATEC에서 제시한 형태소의 분할과 상당한 부분이 불일치했다. 이런 형태소 분할의 문제는 실질 형태소와 형식 형태소 모두에서 나타났다.

실질 형태소 분할의 차이는 다음 예들과 같다.(품사 태그는 생략했음)

예4> 질서정연한
MATEC : 질서정연 + 하+ ㄴ
우리결과 : 질서 + 정연하 + ㄴ

예5> 눈깜짝할
MATEC : 눈깜작하 + ㄹ
우리결과 : 눈+깜짝하 + ㄹ

예6> 한치의
MATEC : 한 + 치 + 의
우리결과 : 한치 + 의

예7> 선보이는 :
MATEC : 선 + 보이 + 는
우리결과 : 선보이 + 는

우리의 형태소 분석기에서 예4>와 예5>는 확장한 복합 명사의 범주인 [N+V]에 포함되어 분석된 형태로, MATEC과는 다르게 나타난다. 예6>과 예7>또한 우리의 사전 의 표제어와 MATEC 사전 표제어의 불일치에서 발생한 문제점이다.

형식 형태소의 분할의 차이는 MATEC에서 채택한 조사나 어미의 분할이 우리와 관점이 다르다. 이에 대한 것은 다음 예와 같다. (품사 태그 생략)

예8> 군부대에서는
MATEC : 군부대 + 에서 + 는
우리결과 : 군부대 + 에서는

예9> 기업으로부터
MATEC : 기업 + 으로 + 부터
우리결과 : 기업 + 으로부터

예8>과 예9>에서 볼 수 있듯이 MATEC에서는 형식 형태소를 최소 단위로 나누었으며, 우리 형태소 분석에서는 형식 형태소도 최장일치에 의한 가장 긴 형태의 형식 형태소로 분석한다.

이번 MATEC을 기준으로 평가한 형태소 분석기는 형태소 분석기의 평가 코퍼스로 띄어쓰기를 기준으로, 한 어절 단위의 입력 문장을 기준으로 하였기 때문에 우리 시스템이 장점으로 취하고 있는 다어절을 기준으로 한 분석을 전혀 반영할 수 없었다. 일반적으로, 자연어 처리에서의 입력은 문장 단위로 이루어지고 있다. 평가 코퍼스의 입력이 한 어절 단위였으므로, MATEC 평가 기준에 대한 우리의 형태소 분석기의 재현률과 정확률은 더 떨어지는 결과를 가져오게 되었다.

우리가 구현한 형태소 분석기의 성능을 측

정하기 위해서 사용한 환경과, 우리 형태소 분석기의 구성요소는 다음 <표 4>와 같다.

OS	Windows 2000		
CPU	Pentium III, 450MHZ		
Main Memory	198MB		
사전크기	12만 단어(명사 9만여 개)		
메모리 요구량	최초실행	5776KB	
	최대요구	5784KB	
프로그램 크기	실행파일	152KB	
	DLL (단위:KB)	형태소	384
		MISC	212
		사전	328
	사전	3.83MB	

<표 4> 시스템 사양과 구성 요소

MATEC에서 제시한 정답을 우리의 형태소 분할에 맞게 수정하여 평가한 정확률, 분석에 소요된 시간은 <표 5>와 같다.

시험말뭉치	MATEC 최종 평가에 사용된 33855어절
총분석소요시간	2분 45초
한 어절 분석시간	0.00487초
한 어절당 분석 후보수	1.75개
정확률	0.92

<표 5> 형태소 분석 성능

4. 결론

자연어 시스템을 이루는 첫 단계인 형태소 분석기는 어절의 최소 단위를 찾아내는 형태소 분석 나름대로의 고유 기능이 있다. 그러나, 그 고유 기능에만 충실하여, 다음 단계의 응용 프로그램을 고려하지 않는다면, 결코 좋은 형태소 분석기라 할 수 없을 것이다.

본 논문은 한 어절 중심이 아닌, 어절의 관계를 고려하여 다어절 중심의 형태소 분석을 하여, 형태소 분석의 수를 최소로 줄이는 방법을 사용하는 형태소 분석기를 소개하였다. 우리가 개발한 형태소 분석기가 갖는 가장 큰 특징은 다음과 같다.

첫째, 다어절에 걸쳐서 나타나는 복합용언, 의사 조사를 하나의 형태소로 결합함으로써, 다음 단계로의 입력을 줄였다. 형태소 분석기를 사용하는 응용 프로그램의 부담을 줄여 주었다.

둘째, 복합 명사의 처리에 명사와 명사의 나열 뿐만 아니라, 명사와 용언이 띄어 쓰기 오류로 붙어 쓰여진 경우도 같이 처리하도록 하여 띄어쓰기 오류에 대해서도 대응할 수 있도록 하였다.

셋째, 형태소 분석과 분석 결과를 생성하는 부분(후처리)를 완전하게 분리하여, 후처리 부분만 수정하여 어떤 응용 프로그램으로의 결과로 손쉽게 생성할 수 있도록 하였다.

[참고 문헌]

[김재훈 96] 김재훈, 오류-보정 기법을 이용한 어휘 모호성 해소, 한국과학기술원 전산학과 대학원 박사학위 논문, 1996

[김철수 98] 김철수 "한국어 형태소 분석 환경을 효율적으로 지원하는 사전 구조", 전북대학교 컴퓨터 과학과 박사학위 논문, 1989

[동아 94] 동아 새국어 사전 제 3판, 두산 동아 출판사

[이근용 95] 이근용, 김철수, 이용석, "사전 검색 알고리즘을 이용한 자소단위 한국어 형태

소 분석", 정보과학회 학술 논문지 Vol 22, No 2, 1995

[이기오 94] 이기오, 김기철, 이용석, "형태소 분석 주도의 한국어 복합동사 처리", 제 6회 한글 및 한국어 정보처리 학술대회, pp. 119~127, 1994

[임희석 97] 임희석, 언어 지식과 통계 정보를 이용한 한국어 품사 태깅 모델, 고려대학교 대학원 컴퓨터학과 박사학위 논문, 1997