어휘정보를 이용한 형태적 모호성 축소

김 재 훈†

요 약

가능한 모든 형태소 해석을 찾아내는 한국어 형태소 해석기는 필요 이상으로 많은 수의 형태소 해석 결과를 생성하므로 자연언어 처리 시스템의 상위 과정, 즉 구문해석, 의미해석 등에 큰 도움이 되지 않고 있다. 이 문제를 해결하기 위해, 이 논문은 어휘화된 배열규칙과 형태적 포섭관계와 같은 언어지식을 이용하여 형태적 모호성을 줄이는 방법을 제안한다.

실험을 통해 어휘화된 배열규칙과 형태적 포섭관계는 형태적 모호성을 줄이는 매우 유용한 정보임을 알 수 있었으며, 이들 두 방법을 동시에 적용했을 때, 형태적 모호성의 68%가 감소되었다.

Lexical Information Based Morphological Ambiguity Reduction

Jae-Hoon Kim†

ABSTRACT

Most of Korean morphological analyzers generate more morphological analyses than we need. This is one of the reasons to make difficult for Korean processing systems such as a part-of-speech tagging system and a syntactic parsing system. To reduce the morphological analyses, we use two kinds of lexical information as linguistic knowledge; one is called the lexicalized morphotactics and the other is called the subsumption relation on morphological analyses.

Our experiment shows that the lexicalized morphotactics and the subsumption relation are very useful linguistic information to reduce the morphological ambiguity. When using two informations in Korean morphological analysis, the reduction rate of the morphological structures is 68%.

1. 서 론

한국어 형태소 해석의 목적은 주어진 한국어 어절에 대한 가능한 모든 형태소 해석을 찾아내는 데 있다. 이 경우, 한국어 어절 '소나무라고?'의 형태소 해석 수는 무려 51[1]개나 된다. 아래에는 그 해석의 일부를 보여 주고 있으며, 올바른 결과는 두 번째 것이다.[2]

† 정 회 원 : 한국해양대학교, 컴퓨터공학과
논문접수 : 1997년 3월 26일, 심사완료 :

1. [1]에서 사용하는 한국어 형태소 해석기의 결과 수이다.
2. 이하에서 사용되는 품사태그는 〈부록 1〉에 실었다. 참고로 "소나무/nc"는 형태소 '소나무'이고, 품사가 nc(보통명사)임을 의미한다. 또, "소나무/nc +이/jcp +라고/jca"에서 +는 형태소의 부분해석이 연이어짐을 표현하는 연산자(concatenator)이다. 이 논문에서 형태소의 해석결과는 선형구조(linear structure)를 가지는 것으로 가정한다.

소나무/nc +라고/jca +?/s.
소나무/nc + 이/jcp + 라고/ef + ?/s.
소/nc +나무/nc +라고/jca +?/s.
소/nc +나무/nc +이/jcp +라고/ef +?/s.
소/nc +나/nc +무/nc +라고/jca +?/s.
소/nc +나/nc +무/nc +이/jcp +라고/ef +?/s.
....

따라서 한 어절의 모든 형태소 결합 관계를 분석하여 제시하는 방법은 자연언어 처리의 상위 과정(품사 태깅, 구문 해석 등)에 큰 도움이 되지 못한다. 더구나, 이 결과는 자연언어의 모호성을 제거하려고 하는 자연언어 처리 시스템의 결과와는 상충된다. 따라서, 자연언어 처리의 첫 번째 단계인 형태소 해석에서 이와 같은 모호성을 축소 혹은 해소시키지 않으면 상위 과정에서의 해석이 거의 불가능하게 된다. 이 논문의 목적은 가능한 한 주어진 문장에 적합하지 않는 형태소 해석을 출력하지 않도록 하는 것이다. 이를 형태적 모호성 축소(morphological ambiguity reduction)라고 한다. 일반적으로 형태소 해석에서 사용하는 언어정보, 즉, 형태소 배열규칙은 품사에 관한 이진 관계를 이용하거나[2], 품사를 기본 요소로 하는 유한 상태 전이망(finite-state transition network)을 이용한다[3]. 이와 같이 품사들 사이의 관계를 형태소 배열규칙으로 이용하면 형태적 과잉해석(morphological over-analysis)의 원인이 된다. 이 논문은 형태적 과잉해석으로 발생된 형태적 모호성을 축소하기 위해 언어정보로서 어휘화된 (형태소) 배열규칙(lexicalized morphotatics)과 단어의 형성과정을 모형화한 (형태적) 포섭관계(morphological subsumption relation)를 이용한다.

이 논문의 구성은 다음과 같다. 2장에서 관련 연구로서 한국어 형태소 해석과 한국어 단어 형성에 관해서 기술한다. 3장과 4장에서 각각 어휘화된 배열규칙과 포섭관계를 이용한 형태적 모호성 축소 방법에 관해서 기술한다. 5장에서 실험 및 평가에 대해서 기술하고, 6장에서 관련연구와 비교하고, 7장에서 결론을 맺는다.

2. 관련 연구

2.1 한국어 형태소 해석

한국어 형태소 해석 방법은 최장일치법, head-tail 분리 기억법[4], tabular parsing을 이용한 방법[2], 두 단계 모델을 이용한 방법[5], 음절정보를 이용한 방법[6] 등 여러 가지 방법들이 제시되었다. 이들 연구의 대부분은 어떻게 한국어 형태소를 해석할 것인가에 초점을 맞추어 왔다. 최근에 몇몇 연구는 효율적인 해석 기법을 제안하였다[7, 8]. 형태소 해석의 모호성 축소에 관한 연구는 거의 이루어지지 않았으나, 최근에 음소, 음절, 그리고, 문자열의 배제정보를 이용한 연구가 있었다[9]. 이들 배제정보는 경험적인 정보로써 이를 구축하는 데 많은 시간이 소요될 뿐 아니라, 형태소 해석 알고리즘과 밀접한 관계를 가지게 된다.

이 논문이 사용하는 한국어 형태소 해석은 챠트 파싱 기법을 이용하며, 크게 네 가지의 과정을 거친다[1]. 첫째, 가능한 형태소를 찾아주는 형태소 분리(morpheme segmenting) 과정, 둘째, 두 개의 형태소가 결합할 때 발생하는 여러 불규칙 현상 및 음운 현상으로 변형된 형태소를 복원하는 불규칙 현상 처리(base form recovering) 과정, 셋째, 찾아진 형태소들이 올바른 한국어 어절을 형성하는지의 유무를 검사하는 형태소 배열규칙 검사(morphotactics checking) 과정, 넷째, 사전에 등록되지 않은 단어를 처리하는 미등록어 처리(unknown word processing) 과정이다. 지금까지의 한국어 형태소 해석의 대부분은 주어진 한국어 어절에 대한 가능한 모든 형태소 해석을 찾아내는 데 그 목적이 있었다. 이와 같은 방법의 문제점 중 하나는 형태소에 대한 과잉해석이다. 과잉해석의 원인은 아래와 같이 요약된다.

- 간단한 형태소 배열규칙을 사용함으로써 필요 이상으로 많은 해석을 생성한다. (1ㄱ)은 올바른 형태소 해석결과이나, (1ㄴ)은 잘못된 형태소 해석결과이다. 왜냐하면, 접미사 '-씨'는 주로 고유명사와 결합되기 때문이다. 이를 해결하기 위해서 접미사를 더 세분할 수 있으나, 품사를 세분하는 문제는 또 다른 연구 과제이다[10, 11]. 왜냐하면, 어떤 문제를 해결하기 위해서 특정 품사를 세분하면, 또 다른 문제를 해결하기 위해서는 그 세분된 품사가 걸림돌이 되기도 하기 때문이다. 따라서 가능하면, 표준화된 품사집합[10]을 사용하고, 문제에 따라 추가적인 정보를 사용하는 것이 바람직하다.

(1) ㄱ. 꽃씨/명사
ㄴ. * 꽃/명사 +씨/접미사

• 불규칙 현상을 처리할 때, 제약조건의 부족으로 과잉해석을 유발시킨다. 예를 들면, 한국어에서 두 형태소가 결합될 때, 모음이 중복될 경우에는 이를 하나로 축약한다. 이와 같은 현상을 모음축약이라고 하는데, 이는 불규칙 현상 처리 과정에서 축약된 모음들이 복원된다. 예를 들면, 문장 (2ㄱ)과 같은 문장에서 '가'는 '가 +아'가 축약되어 표현된 것으로 (2ㄴ)과 같이 복원되어야 한다.

(2) ㄱ. 학교에 가 보았다.
ㄴ. 가/동사 +아/보조적 연결어미

또한, (3ㄱ)과 같은 문장에서 '나와'는 (3ㄴ)과 같이 해석되어야 한다. 그러나, (2ㄱ)의 '가'를 처리하는 규칙에 의해서 (3ㄷ)과 같은 해석도 가능하다.

(3) ㄱ. 그는 학생으로 나와 현실과 서로 다른 역할을 ...
ㄴ. 나오/동사 +아/보조적 연결어미
ㄷ. 나/동사 +아/보조적 연결어미 +오/보조용언 +아/연결어미

2.2 한국어의 단어 형성

단어의 형성은 새로운 단어를 만들어 내는 것이다. 한국어에서는 어근과 어근의 결합(복합법), 어근과 접사의 결합(파생법), 어근의 창조(어근 창조법) 등의 방법으로 새로운 단어를 만들어 낸다[12]. 이 절에서는 이 논문과 직접 관계가 있는 합성법과 파생법에 관해서 살펴본다.

2.2.1 복합법

한국어 단어 형성에서 복합법은 독립적으로 사용할 수 있는 두 단어가 하나의 단어를 형성하는 것을 말한다. 이 방법은 예 (4)에서 보는 바와 같이 매우 다양한 유형으로 나타난다[12].

(4) ㄱ. 돌/명사 +다리/명사 → 돌다리/명사
ㄴ. 작은/관형사형 +집/명사 → 작은집/명사
ㄷ. 지나/동사 +아/보조적 연결어미 +가/보조동사 → 지나가/동사

이 방법으로 형성된 단어들도 매우 생산적이기는 하나, 이들 복합어는 많은 형태소 해석기에서 하나의 형태소(단일어)로 간주하고 있다. 그 이유는 접사와는 다르게 두 단어의 결합이 상대적으로 단어 자체에 매우 의존되고, 매우 복잡한 형태의 의미적 구조를 가지고 있기 때문이다[13]. 예를 들면, (4ㄱ)에서 모든 명사가 '다리'라고 하는 단어와 결합해서 새로운 단어를 형성할 수 있는 것은 아니다. 또한, 복합어 '글다리'와 '구름다리'는 '돌다리'와 전혀 다른 의미적 관계가 있다.

2.1.2 파생법

한국어 단어 형성에서 파생법은 주로 접사로 새로운 단어를 형성하는 것을 말한다. (5ㄱ)은 접두사에 의해서 새로운 단어가 형성되었고, (5ㄴ)은 접미사에 의해서 생성되었다.

(5) ㄱ. 맨/접두사 +주먹/명사 → 맨주먹/명사
ㄴ. 공부/명사 +하/접미사 → 공부하/동사

파생법은 매우 생산적으로 단어를 형성한다. 따라서, 한국어 형태소 해석기의 대부분은 접사를 그 처리대상으로 삼는다. 이 경우의 장점은 많은 미등록어에 대한 처리부담이 줄어든다는 것이다. 단점은 많은 경우에 접사의 의미가 모호하므로 복합어의 의미해석에 많은 어려움을 가져온다는 것이다. 최근에는 이들의 단점을 보완하는 입장에서 매우 생산적이면서 모호성을 가지지 않은 일부의 접사(예를 들면, '-하', '-되', 등)에 대해서만 접사로 처리하고, 그렇지 않은 접사에 대해서는 파생된 하나의 단어로 간주하여 기본적인 형태소(단일어, simple word)로 처리한다[1].

3. 어휘화된 배열규칙을 통한 형태적 모호성 축소

많은 한국어 형태소 해석기는 (형태소) 배열규칙(morphotatics)으로 (형태소) 접속정보를 사용한다[1, 2, 8]. 2.1절에서 언급했듯이 비교적 단순한 형태소 배열규칙인 접속정보는 형태적 과잉해석(morphological over-analysis)의 원인이 된다. 일반적으로 명사 접미

사(xn)는 거의 모든 명사류와 결합이 가능하다. 따라서, 접속정보에는 〈n*, xn〉[3]과 같은 관계를 가지고 있다. 그러나, 명사 접미사 '-씨'는 대부분의 명사와 결합 가능하지 않고, 주로 고유명사와 결합한다. 이와 같이 접미어나 기능어의 개별적인 성질에 따라서, 결합 가능한 품사나 단어가 제한된다. 품사만으로 구성된 접속정보는 이와 같은 단어의 개별적인 성질을 표현할 수 없다. 이 문제를 해결하기 위해서 일반적으로 세분된 품사를 사용한다[2, 8]. 그러나, 품사를 세분하면 또 다른 문제를 유발한다. 즉, 어느 정도로 많은 품사를 사용해야 하는지, 세분된 품사는 다른 응용 분야에서도 적합한지 등의 문제가 유발된다. 이와 같은 문제점 때문에, 배열규칙의 정보 부족 문제를 해결하기 위해 품사를 더 세분하는 방법은 그다지 좋은 방법이 아니다. 이 논문은 이 문제를 해결하기 위해 어휘정보를 사용한다. 어휘 자체를 배열규칙에 사용할 때, 해결되어야 하는 문제는 어떻게 어휘를 배열규칙에 포함시킬 것인가 하는 것과 배열규칙에 포함될 어휘를 어떻게 찾을 것인가 하는 것이다. 아래에서 이 두 가지 문제에 대해서 구체적으로 설명하겠다.

3.1 어휘화된 형태소 배열규칙

일반적인 형태소 접속정보는 〈표 1〉과 같다[1]. 이 접속정보는 품사들의 쌍으로 정의되는 이진관계(binary relation)이다. 이 논문은 〈표 1〉과 같은 형태소 접속정보에 접미어나 기능어를 첨가시킴으로써 형태적 모호성을 축소한다. 이 접속정보를 어휘화된 (형태소) 배열규칙 혹은 어휘화된 (형태소) 접속정보라고 하며, 〈표 2〉와 같은 구조를 갖는다. 〈표 2〉에서 $*/t_i$는 품사 t_i를 갖는 모든 형태소를 의미한다. 어휘화된

〈표 1〉 어휘화되지 않은 한국어 형태소 배열규칙
〈Table 1〉 Korean non-lexicalized morphotactics

〈왼쪽 품사,	오른쪽 품사〉	〈왼쪽 품사,	오른쪽 품사〉
〈a,	jca〉	〈xpa,	exa〉
〈a,	jcm〉	〈xpa,	exm〉
〈a,	jcp〉	〈xpa,	exn〉
〈a,	jx〉	〈xpv,	ecq〉
〈...,	....〉	〈xpv,	exn〉

3. 여기서, n*의 의미는 n으로 시작하는 모든 품사 태그를 의미한다.

〈표 2〉 어휘화된 한국어 형태소 배열규칙
〈Table 2〉 Korean lexicalized morphotactics

〈왼쪽 품사,	오른쪽 품사〉	〈왼쪽 품사,	오른쪽 품사〉
〈*/a,	*/s′〉	〈화/xn,	는/jx〉
〈*/a,	*/s,〉	〈화/xn,	도/jx〉
〈*/a,	*/s.〉	〈화/xn,	되/xpv〉
〈*/a,	*/s`〉	〈화/xn,	로/jca〉
〈...,	....〉	〈히/xa,	들/jx〉

접속정보는 어휘화되지 않은 접속정보와 마찬가지로 이진순서관계(binary ordered relation)로 표현될 수 있다. 따라서 기존의 형태소 해석 알고리즘은 수정할 필요가 전혀 없다.

3.2 형태소 배열규칙에 포함될 형태소의 선정

어휘화된 형태소 배열규칙을 구성하는 데 있어서 가장 큰 문제는 어떤 형태소를 배열규칙에 포함시킬 것인가 하는 것과 선택된 형태소를 이용하여 어떻게 배열규칙을 어떻게 만들 것인가 하는 것이다. 이 논문에서는 비교적 간단한 방법으로 이 문제를 해결한다. 첫번째 문제는 어휘의 수가 제한적인 일부의 기능어는 모두 포함시켰다. 이 논문에서 선택된 기능어는 크게 어미와 조사 그리고 접사들로서 아래와 같다.

ecq, ecs, ecx, ef, efp, exa, exm, exn
jc, jca, jcm jcp, jcv, jj, jx,
xa, xn, xpa, xpv

두 번째 문제는 품사 태깅 말뭉치로부터 모든 형태소/품사 쌍을 구한다. 그리고 나서, 기능어나 접미어가 아닌 형태소/품사(m_i/t_i)에 대해서 MSC 일반화 방법[14, pp. 188-196]을 이용하여 일반화시킨다. 이와 같은 접근방법의 가장 큰 문제는 접속정보 내에 포함되지 않은 관계가 문장에서 나타나는 경우이다. 이는 미등록어 처리부에서 배열규칙의 제약조건을 완화함으로써 해결될 수 있다.

3.3 어휘화된 접속정보를 이용한 형태적 모호성 축소

어휘화되지 않은 접속정보를 사용할 경우의 형태

소 해석 방법과 어휘화된 접속정보를 사용한 형태소 해석 방법은 거의 같다. 단지, 접속정보를 검사하는 부분에서 어휘화된 관계에 해당하는 형태소가 속하는지를 확인하면 된다. 그리고 미등록어 처리부에서 어휘화된 접속정보에 의해서 실패되었을 경우에는 이를 완화하여(어휘화 부분을 검사하지 않음) 처리할 수 있도록 하는 기능만 추가하면 된다.

4. 형태적 포섭관계을 이용한 형태적 모호성 축소

많은 한국어 단어는 단어의 형성 과정을 통해서 만들어진 복합어들이다. 복합어의 일부는 형태소 해석의 처리대상에 포함되고, 또 다른 일부는 하나의 단어(단일어, simple word)로 간주된다. 복합어를 단일어로 간주할 경우에도 형태소 해석기는 복합어에 포함된 어근들의 결합으로 해석한다. 이 논문은 이와 같이 두 형태소의 해석 결과들 사이의 복합어 관계를 찾아서 어근들의 결합으로 해석된 결과를 최종적인 형태소 해석 결과로부터 제거하고자 한다[15, 16] 복합어에 대한 형태소 해석 결과를 구체적인 형태소 해석 결과라고 하고, 어근들의 결합에 의한 해석 결과를 일반적인 형태소 해석 결과라고 한다. 언어학적으로 일반적인 형태소 해석 결과의 의미구조는 매우 복잡하나[17, 18], 이들 중 하나는 구체적인 형태소 해석 결과의 의미구조를 포함한다.

4.1 포섭관계의 정의 및 성질

(정의 1) 어절 Ξ에 대한 형태소 해석 결과 $A=\{A_1, A_2, ..., A_n\}$가 있다고 가정하자. 이때 해석 결과 A_i가 해석 결과 A_j보다 더 일반적인(more general) 해석이라면, A_i가 A_j를 포섭한다(subsume)라고 하며, $A_i \subseteq A_j$로 표기한다. 여기서, A_i를 포섭하는 해석(subsuming analysis)이라고 말하고, A_j를 포섭되는 해석(subsumed analysis)이라고 말한다. ■

예를 들면, 어절 '날아오다'에 대한 형태소 해석 결과는 예 (6)과 같으며, (6ㄱ)은 복합어이다. 정의에 따라서, (6ㄴ)은 (6ㄱ)을 포섭한다. 즉, (6ㄴ)이 (6ㄱ)보다 더 일반적인 형태소 해석 결과이다.

(6) ㄱ. 날아오/pv +다/ef
ㄴ. 날/pv +아/ecx +오/px +다/ef

(정의 2) 어절 Ξ에 대한 형태소 해석 결과를 $A=\{A_1, A_2, ..., A_n\}$라고 하자. 형태소 해석 결과 A에 관한 포섭관계(subsumption relation) $S=(A, \subseteq)$는 식 (1)와 같이 정의된다.

$$S=\{(A_i, A_j) | A_i \subseteq A_j\} \quad (1)$$ ■

(성질 1) 포섭관계는 부분순서 관계(partial ordered relation)이다. ■

(7) ㄱ. 소나무/명사
ㄴ. 소/명사 +나무/명사
ㄷ. 소/명사 +나/명사 +무/명사

예를 들면, 어절 '소나무'에 대한 형태소 해석 결과인 예 (7)로부터 포섭관계 S(〈표 3〉)를 얻을 수 있다.

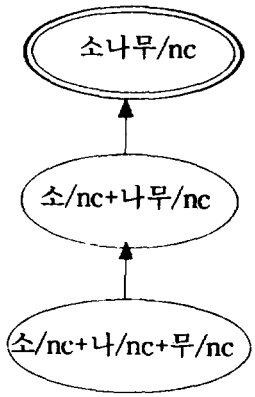
〈표 3〉 어절 '소나무'의 형태소 해석 결과에 대한 포섭관계.
〈Table 3〉 Subsumption relation on the morphological structures of eojeol 'sonamu'.

포섭하는 해석 결과	포섭되는 해석 결과
(소/nc + 나무/nc,	소나무/nc)
(소/nc +나/nc +무/nc,	소나무/nc)
(소/nc +나/nc +무/nc,	소/nc +나무/nc)

일반적으로 부분순서 관계는 Hasse 다이그램으로 표현될 수 있으며[19], 포섭관계 S의 Hasse 다이그램은 (그림 1)과 같다. Hasse 다이그램에서는 이행관계(transitive relation)는 나타나지 않는다.

(정의 3) 어절 Ξ에 대한 형태소 해석 결과 A에 대한 포섭관계 $(A, \subseteq)$가 부분순서 관계라고 하면, 어떤 해석 A_j도 포섭하지 않는 해석 A_i를 최대해석(maximal analysis)이라고 한다. ■

(그림 1)에서 최대해석은 '소나무/nc'이고, 복선으로 표현되었다.



(그림 1) 어절 '소나무'에 대한 포섭관계 S의 Hasse 다이그램.

(Fig. 1) Hasse diagram of subsumption relation S on the morphological structures of eojeol 'sonamu'.

(성질 2) 어절 Ξ에 대한 형태소 해석 결과 A에 대한 포섭관계 $(A, \subseteq)$가 부분순서 관계일 때, Ξ에 대한 형태소 해석 결과로는 최대해석들만 있으면 충분하다. ■

최대해석을 포섭하는 모든 해석들은 최대해석보다 더 일반적인 해석이기 때문에 최종적으로는 최대해석만 이용하면 된다.

좀더 구체적인 예를 통해서 포섭관계를 살펴보자. 두 형태소 해석결과 (7ㄱ)과 (7ㄴ)은 형태적으로는 물론이고, 구문적으로도 올바른 결과라고 말할 수 있다. 그러나, 의미론적인 입장에서 해석결과 (7ㄱ)과 (7ㄴ)을 살펴보면, (7ㄴ)은 여러 가지의 의미로 해석될 수 있으며, 그 중에 하나는 (7ㄱ)의 의미해석을 포함한다.[4] 이와 같은 의미에서, (7ㄱ)은 (7ㄴ)에 비해 훨씬 더 구체적인 해석이라고 할 수 있다. 이 논문은 이와 같이 어느 하나의 해석이 다른 해석의 구체적인 해석이 될 수 있는 조건을 찾아서, 더 구체적인 해석을 올바른 형태소 해석의 결과로 삼는데, 한 어절 내에서 가장 일반적인 해석이 바로 최대해석이 된다.

4.2 포섭조건

두 형태소 해석 결과가 포섭관계에 포함되는지를 검사하기 위한 조건이 필요하다. 이 조건을 이 논문에서는 포섭조건(subsumption condition)이라고 한다. 포섭조건은 아래 항목들의 부울 표현식(Boolean expression)으로 표현된다.

1. 어절 Ξ
2. 어절 Ξ의 형태소 해석 결과, $\{A_1, A_2, ..., A_n\}$
3. 문장 상에서 어절 Ξ의 위치, $loc(\Xi)$
4. 예외적인 해석, $exception(\Xi_{sub})$

여기서, Ξ_{sub}는 Ξ의 부분 문자열에 의해서 만들어진 정합 패턴이다. 〈표 4〉에서는 포섭조건을 만족하는 포섭관계에 대한 예를 보여 주고 있다. 표에서 T_i와 T_j는 각각 포섭하는 해석 A_i와 포섭되는 해석 A_j의 품사열이다. 대부분의 포섭조건은 품사열에 의해서 구성된다. 그러나, 다섯번째 조건의 경우에는 주어진 어절의 위치가 추가되어 있다. 구체적인 예를 보면, 형태소 해석 '따라서/ajs'는 문장의 시작위치(문두)에 있을 경우에만 형태소 해석 '따르/pa +아서/ecs'에 비해 더 구체적인 해석이 될 수 있다.

〈표 4〉 포섭관계의 예

〈Table 4〉 Examples of subsumption relations

번호	포섭하는 해석(A_i)	포섭되는 해석 (A_j)	포섭조건		
			T_i	T_j	loc(W)
1	(메일/nc, 우유/nc)	(메일우유/nq)	(nc, nc)	(nq)	
2	(서울/nq, 우유/nc)	(서울우유/nq)	(nq, nc)	(nq)	
3	(호텔/nc, 신라/nq)	(호텔신라/nq)	(nc, nq)	(nq)	
4	(소/nc, 나무/nc)	(소나무/nc)	(nc, nc)	(nc)	
5	(따르/pv, 아서/ecs)	(따라서/ajs)	(pv, ecs)	(ajs)	문두
6	(날/pv, 아/ecx, 오/px)	(날아오/pv)	(pv, ecx, px)	(pv)	
7	(즐겁/pa, 어/ecx, 지/px)	(즐거워지/pv)	(pa, ecx, px)	(pv)	
8	(동시/nc, 에/jca)	(동시에/a)	(nc, jca)	(a)	
9	(이/npd, 대로/jca)	(이대로/ad)	(npd, jca)	(ad)	

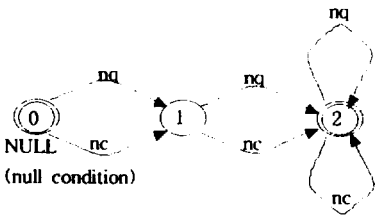
포섭조건은 포섭되는 해석에 따라 서로 구별되어 질 수 있다. 따라서, 각 포섭되는 해석에 해당하는 포섭조건들은 하나의 오토마타에 의해서 표현될 수 있다.

4. 단어의 형성 과정에서 '소나무'는 '솔'과 '나무'가 결합하여 하나의 단어로 굳어진 복합어이다.

즉, 정규문법(regular grammar)에 의해서 표현될 수 있다. 포섭조건을 표현하는 정규표현의 단말기호(terminal symbol)는 '*m/t*'와 같은 형태로 표현된다. 여기서 *m*은 형태소이거나 '*'(어떤 형태소와도 정합이 가능함을 의미함)이고, *t*는 품사 태그 중의 하나이다. 시작상태(starting state)에는 *loc*(Ξ)와 *exception*(Ξ_{sub})이 부가되며, 이를 시작조건이라고 한다. (그림 2)는 포섭되는 해석이 고유명사(nq)일 경우의 포섭조건을 표현하는 정규문법과 그에 대응하는 오토마타이다. 시작상태와 종결상태는 각각 0과 2이고, 시작조건은 아무런 제약이 없다.

시작상태: { 0 }
종결상태: { 2 }
시작조건: { }
정규규칙:
0 → ?/nc 1
0 → ?/nq 1
1 → ?/nc 2
1 → ?/nq 2
2 → ?/nc 2
2 → ?/nq 2

ㄱ) 포섭조건을 표현한 정규문법.
a) Regular grammar representing subsumption condition.



ㄴ) 포섭조건을 표현하는 오토마타
b) Automata representing subsumption conditions

(그림 2) 고유명사의 포섭조건을 표현한 정규문법 및 오토마타
(Fig. 2) Regular grammar and its corresponding automata representing the subsumption condition of proper noun.

4.3 포섭관계를 이용한 형태적 모호성 축소

포섭관계를 이용한 형태적 모호성 축소 방법은 형태소 해석 결과에 대한 포섭관계를 찾고, 그 포섭관계로부터 최대해석을 찾는 것이다. 두 형태소 해석 사이의 포섭관계가 있는지 없는지에 관한 결정은 4.2절에서 설명한 포섭조건을 검사함으로써 이루어진다. 포섭조건을 효율적으로 검사하기 위해서, 먼저 두 해석의 상대적인 차를 구한다. 얻어진 상대적인 차에 대해서 포섭조건을 검사한 후, 그 조건을 만족할 경우, 포섭관계가 성립하는 것으로 간주한다. 예를 들면, 어절 '중국의'를 형태소 해석 결과는 예 (8)과 같다. (8ㄱ)과 (8ㄴ)의 상대적인 차는 예 (9)와 같으며, 예 (8)에 대해서 포섭조건을 검사하면, (8ㄴ)이 (8ㄱ)을 포섭하므로 (8ㄱ)은 어절 '중국의'에 대한 형태소 해석의 최대해석 중 하나가 될 수 있다. 결과적으로 (8ㄴ)은 최종적인 해석에서 제거된다. (그림 3)은 이를 알고리즘으로 기술한 것이다.

(8) ㄱ. 중국/nq +의/jcm
ㄴ. 중/nc +국/nc +의/jcm
(9) ㄱ. 중국/nq
ㄴ. 중/nc +국/nc

```
Input. 형태소 해석결과 A_1,...,A_n
Output. 축소된 형태소 해석결과 R_1,...,R_m, m ≤ n
Method.

for i = 1 to n do reduced[i] = false; enddo
for j = 1 to n do
    for i = 1 to n do
            if (j == i) continue;
            if (reduced[i] == true) continue;
            if ( A_i ⊆ A_j) then
                    reduced[i] = true;
            endif
    enddo
enddo

j = 0;
for i = 1 to n do
    if (reduced[i] == true) continue;
    R_j = A_i; j = j + 1;
enddo
```

(그림 3) 포섭관계를 이용한 형태적 모호성 축소 알고리즘
(Fig. 3) Algorithm for morphological ambiguity reduction using subsumption relation

5. 실험 및 평가

5.1 실험 환경

학습과 시험을 위해서 품사 태깅된 말뭉치[20]를 사용하였으며, 학습 및 시험 말뭉치는 각각 131,581개와 41,122개의 어절로 구성되었다. 평가에 기본이 되는 형태소 해석기는 [1]를 사용하였다. 포섭관계를 이용한 방법에서 포섭조건은 자동적으로 추론할 수도 있

고[16], 수동으로 추론할 수도 있으나, 이 논문에서는 수동으로 추론된 포섭조건을 사용하였다.

5.2 성능 평가

〈표 5〉는 어휘화된 형태소 배열규칙과 포섭관계를 동시에 사용했을 경우, 형태소 해석의 모호성 감소를 보이고 있다. 어휘화된 배열규칙만 사용하였을 경우는 약 54%의 형태소 해석을 감소시켰으며, 포섭관계만을 이용했을 경우에는 약 40%의 형태소 해석을 감소시켰다. 어휘화된 형태소 배열규칙과 포섭관계를 동시에 사용할 경우에는 약 68%의 형태소 모호성을 감소시켰다. 이 결과를 통해서 볼 때, 이들 두 정보는 형태적 모호성을 축소시키는 매우 유용한 정보임을 알 수 있었다. 감소된 결과에는 항상 올바른 결과가 포함되어 있었다.

〈표 5〉 형태적 모호성의 감소율
〈Table 5〉 Reduction rate of morphological ambiguity

형태적 모호성 축소 방법	어휘화된 배열규칙		감소율
	(사용하지 않음) 해석 수	(사용함) 해석 수	
포섭관계(사용하지 않음)	353,903개	162,874개	53.98%
포섭관계(사용함)	211,069개	115,110개	45.46%
감소율	40.36%	29.32%	67.67%

6. 토 의

형태소 해석에서 최장일치법과 포섭관계를 이용한 방법의 차이점에 관해서 살펴본다. 최장일치법의 주된 목적은 사전 접근 회수를 줄이는 데 있으나[21], 이 논문의 주된 목적은 형태적 모호성을 축소하는 데 있다. 이들 두 방법의 비슷한 점은 긴 형태소를 선호한다는 것이다. 그러나, 이 논문은 항상 긴 형태소를 선호하는 것은 아니며, 특정 조건(포섭조건)을 만족하는 경우에만 선호하게 된다. 예를 들어, "제주도 좋은 도시이다."라는 문장에서 '제주도'에 대한 형태소 해석 결과를 살펴보자. 이 문장에서 올바른 해석 결과는 "제주/nq +도/jx"이다. 최장일치법의 경우 아마도 "제주도/nq"만을 출력할 것이다. 그러나, 이 논문에서는 "제주도/nq"와 "제주/nq +도/jx"는 포섭관계에 포함되지 않기 때문에 두 해석 모두 출력된다. 따라서, 이 논문과 최장일치법과의 차이를 한마디로 요약하면 조건을 가진 최장일치법이라고 말할 수 있다. 물론 최장일치법의 경우에도 잘못된 해석이 발생될 경우, 여러 가지의 경험규칙(heuristics)에 의해서 이를 제거하기도 한다[22].

한국어에서 형태소 과잉해석을 줄이는 연구로서 음소, 음절, 그리고 문자열 단위의 배제 정보를 이용한 연구[9]가 있었다. 이 연구는 사전을 검색하기 전에 배제 정보를 이용해서 사전 검색 수를 줄이는데 그 목적으로 두고 있으며, 배제 정보는 수동 및 경험적인 방법에 의해서 만들어 진다. 이 논문은 사전의 검색이 완료되고 부분적인 형태소 격자구조가 완성된 후, 포섭조건을 이용하여 모호성을 줄이고 있으며, 포섭조건은 자동으로 추출하여 사용하고 있다.

정규문법은 자연언어 처리의 여러 분야에서 응용되고 있다. 예를 들면, 음운 및 형태 규칙[23, 24], 대용량 사전의 압축 표현[25], 구절 분리 규칙[26] 등이 있다. 이 논문은 형태적 모호성을 줄이기 위해서 정규문법을 사용하였으며, 정규문법의 단말기호는 품사 뿐 아니라 형태소도 사용되며, 예외정보 및 위치정보를 부가적으로 가질 수 있는 확장된 형태의 정규문법을 사용한다.

7. 결 론

이 논문에서는 형태소 과잉해석으로 발생되는 형태적 모호성을 축소하는 방법에 대해서 기술하였다. 과잉해석은 간단한 형태소 배열규칙과 불규칙 현상 처리에 의해서 발생되는데, 이를 줄이기 위해서 어휘화된 배열규칙과 단어의 형성과정을 모형화한 포섭관계를 이용하였다. 실험을 통해서 이들 두 언어지식는 형태적 모호성을 감소시키기 위한 매우 유용한 정보임을 알 수 있었다. 어휘화된 접속정보는 비교적 간단한 정보이지만, 약 54%의 형태소적 모호성을 감소시킬 수 있었으며, 포섭관계는 약 40%의 형태적 모호성을 줄였다. 두 언어정보를 동시에 사용했을 때, 약 68%의 형태적 모호성을 축소시켰다. 형태적 모호성 축소 방법은 구문 해석이나 자연언어 처리 전반에 응용될 수 있을 것이며, 특히 오류 입력을 요구하지 않는 자연언어 처리 시스템에 매우 유용한 도구로 이

용될 수 있다.

참 고 문 헌

[1] 김재훈, *어류-보정 기법을 이용한 어휘모호성 해소*, 한국과학기술원, 전산학과, 박사학위 논문, 1996.

[2] 김성용, *Tabular Parsing 방법과 접속정보를 이용한 한국어 형태소 해석기*, 한국과학기술원, 전산학과, 석사학위 논문, 1987.

[3] Kim, D.-B, Choi, K.-S., and K.-H. Lee, "Predictive morphological analysis of Korean with dynamic programming," *Korean Journal of Cognitive Science*, vol. 5, no. 1, pp. 145-180, 1994.

[4] 최형석, 이주근, "자연어 처리 앨고리즘," *한국정보과학회 가을 학술대회 발표 논문집*, 제11권, 제2호, pp. 36-43, 1984.

[5] 이성진, *Two-level 한국어 형태소 해석*, 한국과학기술원, 전산학과, 석사학위 논문, 1992.

[6] 강승식, *음절 정보와 복수어 단위 정보를 이용한 한국어 형태소 분석*, 서울대학교 컴퓨터 공학과 박사학위 논문, 1993.

[7] 김덕봉, 예측 중심의 형태소 분석:한국어 어절 인식을 위한 계산 모델, 한국과학기술원, 전산학과, 박사학위 논문, 1996.

[8] 이은철, CYK법에 기반한 한국어 형태소 분석에서의 개선기법, 포항공과대학 대학원, 전자계산학과, 석사학위 논문, 1992.

[9] 임희석, 윤보현, 임해창, "배제 정보를 이용한 효율적인 한국어 형태소 분석기," 한국정보과학회 논문지, 제22권, 제6호, pp. 987-964, 1995.

[10] 최기선, 남영준, 김진규, 한영균, 박석문, 김진수, 이춘택, 김덕봉, 김재훈, 최병진, "한국어 정보베이스를 위한 형태. 통사 태그 표준에 관한 연구," *인지과학*, 제7권, 제4호, pp. 43-61, 1996.

[11] 김진규, "자연언어 처리를 위한 한국어 품사 태그의 몇 가지 문제," *한글*, 제233호, pp. 187-208, 1996.

[12] 이주행, *현대국어문법론*, 대한교과서주식회사, 1993.

[13] 정도환, *국어 복합어의 의미 연구*, 서광 학술 자료사, 1993.

[14] Anzai, Y, *Pattern Recognition and Machine Learning*, Academic Press, 1992.

[15] 김재훈, 장병규, 김길창, 서정연, "한국어 형태소 해석의 모호성 축소," 제1회 지능기술 공동 학술대회 발표논문집, 과학기술진흥센터, 서울, pp. 161-121, 1995.

[16] 김재훈, 장병규, 김길창, 서정연, "형태소의 모호성을 축소하기 위한 포섭조건의 자동 추론," 제7회 한글 및 한국어 정보처리 학술대회 발표논문집, 연세대학교, 서울, pp. 175-180, 1995.

[17] Bourigault, D. "An endogeneous corpus-based method for structural noun phrase disambiguation," *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics(EACL-93)*, Utrecht, The Letherlands, pp. 81-86, 1993.

[18] Isabelle, P. "Another look at nominal compounds," *Proceedings of the International Conference of Computational Linguistics(COLING-84)*, Stanford, California, USA, pp. 509-516, 1984.

[19] Liu, C. L., Elements of Discrete Mathematics, McGraw-Hill Inc, 1986.

[20] 김재훈, 김길창, *한국어에서의 품사 부착 말뭉치의 작성 요령:KAIST 말뭉치*, 한국과학기술원, 전산학과, CS-TR-95-99, 1995.

[21] 최재혁, 이상조, "양방향 최장일치법에 의한 한국어 형태소 분석에서의 사전 검색 횟수 감소 방안," *한국정보과학회 논문지*, vol. 20, no. 10, pp. 1497-1507.

[22] Oi, K., Yumura, T., and Nishida, Y., "A method of Japanese morphological analysis using longest matching method," *Proceedings of the 43th Conference on Information Processing*, vol. 3, 119-120, 1991(in Japanese).

[23] Kaplan, R. M. and Kay, M. "Regular models of phonological rule systems," *Computational Linguistics*, vol. 20, no. 3, pp. 331-378, 1994.

[24] Koskenniemi, K. "A general computational model for word-formation recognition and production," *Proceedings of International Conference on Computational Linguistics*, pp. 178-181, 1984.

[25] Mohri, M. "Compact representation by finite-state

transducers," *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pp. 204-209, 1994.

[26] Ejerhed, E. "Finite-state segmentation of discourse into clauses," *Proceedings of the ECAI 96 Workshop*, pp. 24-33, 1996.

〈부록 1〉 이 논문에서 사용된 한국어 품사 태그

1.	s,	쉼표	27.	a	부사
2.	s.	문자의 종결	28.	ad	지시부사
3.	s′	여는 따옴표(괄호)	29.	ajw	단어 접속 조사
4.	s`	닫는 따옴표(괄호)	30.	ajs	문장 접속 조사
5.	s-	이음표	31.	i	감탄사
6.	su	단위	32.	jcm	관형격 조사
7.	sw	화폐단위	33.	jcp	서술격 조사
8.	sy	기타 기호들	34.	jcv	호격 조사
9.	f	외국어	35.	jca	부사격 조사
10.	nca	동작성 명사	36.	jc	격조사
11.	ncs	상태성 명사	37.	jx	보조사
12.	nc	보통 명사	38.	jj	접속조사
13.	nq	고유 명사	39.	ecq	대등적 연결어미
14.	nbu	단위성 의존 명사	40.	ecs	종속적 연결어미
15.	nb	의존 명사	41.	ecx	보조적 연결어미
16.	npp	인칭 대명사	42.	exm	관형사형 전성어미
17.	npd	지시 대명사	43.	exn	명사형 전성어미
18.	nnn	숫자	44.	exa	부사형 전성어미
19.	nn	수사	45.	efp	선어말 어미
20.	pv	동사	46.	ef	어말 어미
21.	pad	지시 형용사	47.	xn	명사 접미사
22.	pa	형용사	48.	xpv	동사 파생 접미사
23.	px	보조용언	49.	xpa	형용사 파생 접미사
24.	md	지시 형용사	50.	xa	부사 파생 접미사
25.	mn	수 관형사	51.	sp	공백
26.	m	관형사			

김 재 훈

1986년 2월 계명대학교 전자계산학과(이학사)

1988년 2월 한국과학기술원 전산학과(공학석사)

1996년 8월 한국과학기술원 전산학과(공학박사)

1988년 2월~1997년 8월 한국전자통신연구원, 선임연구원

1997년 9월~현재 한국해양대학교, 컴퓨터공학과, 전임강사

관심분야: 자연언어처리, 정보검색, 코퍼스 중심 언어처리, 음성언어처리