한국어 낱말 묶기와 그 응용

은광희·홍정하·유석훈·이기용·최재웅*†

고려대학교 언어정보 연구소

Koaunghi Un, Jungha Hong, Seok-Hoon You, Kiyong Lee and Jae-Woong Choe. 2005. Chunking Korean and an Application. *Language and Information 9.2*, 49–68. Application of chunking to English and some other European languages has shown that it is a viable parsing mechanism for natural languages. Although a small number of attempts have been made to apply chunking to the analysis of the Korean language, it still is not clear enough what criteria there are to identify appropriate units of chunking, and how efficient and valid the chunking algorithms would be when applied to some authentic Korean texts. The purpose of this research is to provide an alternative set of algorithms for chunking Korean, and to implement them, and to test them against some English-Korean parallel corpora, which is English and Korean bibles matched sentence by sentence. It is shown in the paper that aligning related texts and identifying matched phrases between the two languages can be achieved through appropriate chunking and matching algorithms defined on the morphologically-tagged parallel corpus. Chunking and matching processes are based on the content words rather than the function words, and the matching itself is done in terms of the transfer dictionary. The implementation is done in C and XML, and can be accessed through the Internet. (**Research Institute of Language and Information, Korea University**)

Key words: 영한 정렬 (English-Korean alignment), 낱말 묶음 (chunk), 낱말 묶기 (chunking), 형태소 (morpheme), 병렬 코퍼스 (parallel corpus)

1. 서론

과학의 탐구에 있어서 단위 설정의 중요성은 아무리 강조해도 지나치지 않다. 어떤 주장이든지 단위가 정해져 있지 않으면 그 주장은 객관성을 잃고 어떤 비교도 그 근거를

* 136-701, 서울특별시 성북구 안암동 5가 고려대학교 언어정보 연구소. Email: koaunghi@kornet.net, kleist@korea.ac.kr, syou@korea.ac.kr, klee@korea.ac.kr, jchoe@korea.ac.kr

† 영한 낱말 묶음 정렬 프로젝트(프로젝트 책임자: 최재웅)를 지원한 재단법인 언어교육에 감사드린다. 논문을 세밀하게 검토하여 주신 심사자들께도 감사드린다.

상실한다. 언어 과학의 탐구에 있어서도 단위의 설정이 중요하며 일반적으로 언어의 최소 의미 단위는 형태소로 알려져 있다.

형태소는 음운의 연쇄체로 구성되며 형태소들이 모여 좀더 커다란 언어 단위를 형성한다. 이 단위는 문법이라는 제약에 기반을 두는 구구조 문법의 구문 단위가 될 수도 있고 의사 소통의 핵심인 정보에 기반을 두는 Abney (1991)에서 제안된 chunk, 즉 낱말 묶음이 될 수도 있다. 본 논문에서는 후자의 낱말 묶음을 한국어에 적용하기 위한 한국어 낱말 묶기를 구현 중심으로 다룬다.

구문 정보는 문장 이해에 중요한 단서를 제공하므로, 의사 소통의 핵심이라 할 수 있다. 전통적으로 구문 처리는 완전한 구문 분석(full parsing)을 의미한다. 그러나 완전한 구문 분석은 계산의 복잡도가 높고, 응용 시스템에 따라 필요 이상의 정보를 제공하기도 하며, 높은 처리 비용에도 불구하고 안정된 성능을 발휘하지 못하는 실정이다. 이에 대한 대안으로 문장의 부분 구조만을 더 빠르고, 정확하게 처리할 수 있는 부분 구문 분석(partial parsing)이 1990년대 이후 많은 과제에 적용되고 있다.

기본적으로 부분 구문 분석은 문장에서 두드러진 부분 구문 구조만을 분석하는 과정으로, 좀더 복잡한 구조를 단계적으로 분석하여 최종적으로 완전한 구문 분석을 수행하기 위한 전처리 과정에 해당된다. 그러나 부분 구문 분석의 기본적 목적 외에도 간단한 수준의 구문 정보만으로도 충분히 응용 분야에 적용할 수 있다. 예를 들면, 품사 표지 부착, 명사구 인식, 문장 분리, 정보 검색, 정보 추출, 질의 응답, 코퍼스 분석 도구, 개체명 인식과 같은 특정 표현 인식, 문법 검사, 기계 번역 등에서 부분 구문 분석이 활발하게 적용되고 있다. (이공주·김재훈 (2003), 황영숙 (2002) 참조)

부분 구문 분석 단위의 본격적인 정의는 Abney (1991)에서 시도되었다. 낱말 묶음[1]을 내용어(content word) 중심의 문법 단위로 정의하고, 통사·음운적으로 밀접하게 연관되어 있는 어휘들의 집합체인 낱말 묶음으로 문장 분석이 가능하며, 이를 이용한 구문 분석의 활용에 대해 논의하였다. 낱말 묶음에 대한 출발점은 문장 읽기 단위에 관한 언어 화자의 운율적 직관이다. 다음의 문장 (1a)은 (1b)과 같이 운율적 문장 읽기 단위로 분리할 수 있다.

(1) a. I begin with an intuition: When I read a sentence, I read it a chunk at a time

b. [I begin] [with an intuition]: [When I read] [a sentence], [I read it] [a chunk] [at a time]

c. [I begin] [with an intuition]: ∨ [When I read] [a sentence], ∨ [I read it] [a chunk] [at a time]

[1] Miller (1956)에서는 정보 처리의 개념을 기억과 사고의 연구에 도입하여, 인간이 정보를 단기적으로 처리 또는 기억 가능한 용량은 "7±2" 낱말 묶음 (chunk)으로 제한되어 있다고 주장하면서 낱말 묶음에 대해 처음 소개하였다. 이후 낱말 묶음은 심리학, 언어학, 전산언어학, 정보학 등의 분야에서 기억·정보·언어 단위 등으로 다양하게 정의되어 사용되고 있다.

(1c)의 운율적 강세(음영 부분)는 각 문장 읽기 단위 내의 내용어에 위치하며, 문장 읽기 휴지(pause: ∨ 부분)는 문장 읽기 단위 경계 사이가 가장 적당하다. 이러한 운율적 단위로서 문장 읽기 단위는 낱말 묶음에 해당하며, 일정한 형태의 문법적 단위를 구성한다. 일반적으로 하나의 낱말 묶음은 하나의 내용어와 이를 중심으로 통사·음운적으로 밀접하게 관련된 기능어들로 구성된다.

이러한 Abney (1991)의 연구는 전산언어학에서 낱말 묶음 분석(chunk parsing)에 대한 연구를 촉진하였다.[2] Brill (1993)에서는 변형 기반 학습(Transformation based learning)을 이용하여 고품질 품사 표지 부착(part-of-speech tagging) 프로그램을 구현하였다. 이러한 Brill (1993)의 변형 기반 학습을 적용하여 Ramshaw and Marcus (1995)에서는 품사 표지가 부착된 텍스트에 기본 명사구 낱말 묶음을 표상하여 품사 표지 부착의 문제점을 해결하였다. 이렇게 변형 기반 학습, 즉 기계 학습을 통해 자동적으로 추출된 기본 명사구 낱말 묶음 처리 모형이 품사 표지가 부착된 텍스트 해석의 후속 단계로 적절함을 제시하였다. 또한 Buchholz, Veenstra, and Daelemans (1999)에서는 학습 데이터를 모두 기억하고 기억된 학습 데이터 중에서 가장 유사한 항목으로부터 부류(class)를 추정하여 추출하는 기억 기반 학습(memory-based learning)을 통해 낱말 묶음을 처리하였다. 이 과정은 크게 두 단계로 구분되는데, 첫 번째는 몇 가지 유형의 낱말 묶음, 즉 NP, VP, ADJP, ADVP, PP를 분류해서, 이 낱말 묶음에 위치, 시간 등과 같은 부사적 기능을 부착하였다. 두 번째 단계에서는 첫 번째 단계에서 분류된 낱말 묶음을 동사와 관련하여 주어, 목적어, 처소어와 같은 문법 관계를 할당하였다. 이러한 텍스트 낱말 묶음 분석 연구는 전산언어학의 중요한 분야로 자리 잡았고, CoNLL-2000 (Conference on Computational Natural Language Learning)에서는 공동 과제로 진행되어 낱말 묶음 분석에 대한 실제적 기준과 기법 등을 제시하였다.[3]

한편, 한국어의 경우는 영어에 비해 낱말 묶음 분석이 매우 용이하다. 한국어에서는 조사나 어미와 같은 기능어가 내용어와 함께 하나의 어절을 이루므로, 조사나 어미를 이용하여 쉽게 낱말 묶음을 분석할 수 있다. 구문 분석의 방법론으로 신효필 (1999)에서 문법 이론 대신 한국어의 형태·통사적 특성에 따른 명사구 및 동사구 낱말 묶음의 분할 및 분석 규칙을 제안하였다. 또한 응용 분야에 따라 이공주·김재훈 (2003), 황영숙 (2002), 김재훈 (2000), 박성배·장병탁·김영택 (2000), 김미영·강신재·이종혁 (2000), 김창제 외 (1995), 박상규 외 (1995), 안동언 (1987)은 부분 구문 분석 단위를 다양하게 정의하여 사용하였다.

---

[2] 낱말 묶음은 말덩이(이공주·김재훈, 2003), 기본구(황영숙, 2002), 성분 분할 단위(신효필, 1999) 등의 용어로 사용되었다. 낱말 묶음 분석(chunk parsing)은 간단하게 청킹(chunking)으로 불려지기도 하며, 구문 분석에서 완전한 구문 분석이 아닌, 부분 구문 분석이라는 차원에서 partial parsing, light parsing 등의 용어로 사용되기도 하며, 텍스트 단위화로 사용되기도 한다. 본 논문에서는 청킹을 낱말 묶기라는 용어로 사용한다.

[3] 학습 및 실험 데이터로 PennTreebank II의 Wall Street Journal을 이용하였으며, 네델란드 Tilburg 대학교 Sabine Buchholz가 개발한 텍스트 낱말 묶음 분석기를 이용하였다.

본 논문에서는 영한 병렬 코퍼스의 정렬 단위를 낱말 묶음으로 설정하고 영어 낱말 묶음에 한국어 낱말 묶음을 정렬하기 위한 한국어 낱말 묶기의 기준을 설정하여 이를 구현하고자 한다. 낱말 묶기의 기준은 개별 언어의 차원에서 벗어나 한영 병렬 코퍼스와 같이 문법이 서로 다른 언어의 차이점을 고려함으로써 좀더 보편 언어적인 차원에 접근할 수 있다. 또한 내용어 중심으로 낱말 묶음을 정렬함으로써 문법의 형식적인 기능에 의존하지 않고, 의사를 소통하기 위한 의미적인 기능에 중점을 두게 된다. 이같은 입장에서 한국어 낱말 묶기를 위한 기준을 설정하면 한영 정렬과 같은 과제를 수행하는 데에 있어서 효율적임을 주장한다.

2절에서는 한국어 낱말 묶기 기준에 대한 연구를 개관한다. 위에서 소개한 논문의 대부분은 전산학의 입장에서 기술되었으며 한국어 낱말 묶음을 이용한 응용을 주제로 한다. 그러나 한국어 낱말 묶기에 대한 이론적인 연구나 실용적인 설명이 부족하여 본 논문에 유용한 논문으로는 두 편만이 고려될 수 있었다. 3절에서는 한국어 낱말 묶기의 기준을 마련하고 구현한다. 4절에서는 구현된 한국어 낱말 묶기 모듈의 응용성과 보완점을 살펴본다.

2. 기존 연구

한국어 낱말 묶음에 대한 기존 연구는 그리 활발하지 않았다. 대부분의 연구가 전산학 분야에서 한국어의 낱말 묶음을 소개하는 수준에 그쳤고 한국어 낱말 묶기의 이론적 근거를 자세히 밝히지 않았다. 이 절에서는 한국어 낱말 묶기에 대한 기준을 다룬 두 개의 논문을 소개한다.

2.1 부분 구문 분석 (김재훈, 2000)

김재훈 (2000)에서는 다양한 응용 분야에 두루 적용할 수 있도록, 한국어 낱말 묶음에 대해 인터넷 신문(중앙일보, 조선일보) 데이터를 이용하여 체계적인 기술을 시도하였다. 이 논문에서는 한국어 낱말 묶음을 기술하기 위한 한국어의 대략적인 특성을 다음과 같이 기술한다.

- 문장에서 구문적으로 동일한 기능을 수행하는 내용어가 하나의 낱말 묶음으로 분석될 수 있다.
- 낱말 묶음의 의미적 머리어가 별도로 표시될 수 있다. 의미적 머리어는 일반적으로 제일 마지막에 위치하고 "먹고 있다"와 같은 보조 동사 구문에서는 의미적 머리어 "먹다"가 앞에 나온다.
- 기능어 낱말 묶음은 내용어 낱말 묶음으로부터 분리될 수 있다.

- 기능어 낱말 묶음은 의미적 머리어를 가지지 않고, 제일 마지막에 있는 어휘가 의미적 머리어로 간주될 수 있다.
- 병렬 구문은 의미적 머리어가 여러 개가 존재하므로 낱말 묶음으로 분리될 수 있다.
- 모든 형태소는 하나의 낱말 묶음에 속할 수 있다.

이러한 특성을 기반으로 한국어 낱말 묶음을 (2)와 같이 구분한다. 여기서 낱말 묶음 표지가 두 자로 구성된 경우는 내용어 낱말 묶음에 해당하고, 세 자로 구성된 경우는 기능어 낱말 묶음에 해당한다.

(2) 한국어 낱말 묶음의 유형 및 표지 (김재훈, 2000)

유형	표지	유형	표지
명사구 낱말 묶음	NX	용언구 낱말 묶음	PX
부사구 낱말 묶음	AX	관형사구 낱말 묶음	MX
독립어구 낱말 묶음	IX	격조사구 낱말 묶음	JCX
호격조사 낱말 묶음	JVX	접속격조사 낱말 묶음	JCX
관형격조사 낱말 묶음	JMX	보조사구 낱말 묶음	JXX
연결어미 낱말 묶음	ECX	종결어미 낱말 묶음	EFX
전성어미 낱말 묶음	ETX	문장 부호 낱말 묶음	SYX

이 논문에서는 비교적 상세하게 낱말 묶음 표지를 정의한다. (2)에서 보듯, 낱말 묶음 표지의 분류는 형태소 품사 표지 분류 체계와 흡사하다. 낱말 묶음은 형태소가 모여 형성하는, 형태소보다는 더 큰 덩어리라는 점을 고려할 때, 낱말 묶음 표지의 분류 체계는 형태소 품사 표지 분류 체계와 다른 차이점을 보일 것이다. 또한 이 논문은 이론적인 연구에 치중하여 실용적인 측면에서 의문을 야기하는 점이 있다. 예를 들어 격조사구 낱말 묶음이 영한 정렬과 같은 응용 분야에서 어떠한 역할을 하게 될 것인지에 대해 의문이 든다.

2.2 가중적 확률 결합 모델 (황영숙, 2002)

황영숙 (2002)에서는 낱말 묶음을 기본구(base phrase)로 정의하고, 그 구조가 다른 하위 구조를 포함하지 않으면서 머리어가 최대로 투사되어 형성된 구로 정의한다. 또한 기본구는 논항을 포함하지 않으며, 머리어의 전치수식은 동일한 기본구에 포함시킨다.

한국어 문장을 구성하는 기본구는 명사구(NP), 동사구(VP), 부사구(ADVP), 독립어구(IP), 형용사구(ADJP)로 분류하며, 다음은 한국어 기본구 정의 시 고려 사항이다.

- 기본 단위는 형태소이며 실질 형태소와 형식 형태소로 구분한다. 실질 형태소는 머리어 분석에 사용되고 형식 형태소는 구문 관계 분석에 사용된다.
- 명사형 어미 "-음, -기", 용언화 접미사 "-하, -되"에 의한 파생어는 실질 형태소로 취급한다.
- 관용적 표현으로 굳어진 보조 용언구는 하나의 동사구로 처리한다.
- 관형격 조사 "-의"를 포함하는 명사구는 독립된 명사구로 처리한다.
- 명사를 수식하는 관형어는 명사구에 포함한다.
- 관형절의 수식을 받는 명사구는 관형절과 분리한다.
- 조사, 연결 어미, 종결 어미, 기호는 기본구에 속하지 않는다.

위의 기준에 따라 분석된 낱말 묶음의 예는 (3)과 같다.

(3) [ADVP 여기서] [NP 도그마]의 [NP 수정 변경]이 [ADJP 가능하]다는 [NP 사실], [ADVP 곧] [NP 비평 기준]의 [NP 상대성]을 [VP 볼 수 있]다.

이 논문은 낱말 묶음을 이용하여 가중적 확률 결합 모델을 자연 언어 처리에 적용하였다. 낱말 묶음의 분류 체계는 실용적인 측면이 적극 반영되었으나 실제 낱말 묶기는 한국 과학 기술원에서 배포한 국어 정보 베이스에 포함된 구문 구조 부착 말뭉치를 낱말 묶음 표지 부착 말뭉치로 변형한 것이다. 이 논문의 초점이 한국어 낱말 묶기가 아니고 가중적 확률 결합 모델의 응용 사례를 보이는 데에 맞춰져 있어서 한국어 낱말 묶기에 대해서는 소개의 수준에 그쳤다.

3. 한국어 낱말 묶기 모듈의 구현

한국어 낱말 묶기는 영한 정렬 과제의 일부로 수행되었다. 영한 정렬에서는 형식 형태소의 역할을 통한, 한 문장 내에서의 언어 단위의 상호 관계가 중요한 기능을 제공하지 못한다. 영한 정렬이 두 언어 사이의 형식적인 문장 성분 관계을 정렬하는 것이 아니기 때문이다. 영한 정렬은 각 언어의 문장이 담고 있는 정보를 실질 형태소를 통해서 비교하는 기능을 요구한다. 예를 들어 (4)를 보자.

(4) a. [NC John] [VC was clothed] [PC with camel's hair].

a′. [NC 요한은] [NC 낙타털옷을] [VC 입었다].

형식적인 측면에서 (4a)에서 명사 낱말 묶음(NC) 표지를 갖는 "John"은 (4a′)에서 명사 낱말 묶음 표지를 갖는 "요한은"과 "낙타털옷을"에 정렬될 수 있고 (4a)에서 동사

낱말 묶음(VC) 표지를 갖는 "was clothed"는 (4a′)에서 동사 낱말 묶음 표지를 갖는 "입었다"에 정렬될 수 있다. 그러나 (4a)에서 전치사 낱말 묶음(PC) 표지를 갖는 "with camel's hair"는 (4a′)에서 형식적으로 대응될 낱말 묶음이 없다. 문장 성분의 형식적인 관계에 있어서 (4a)에서 주어 역할을 하는 "John"은 (4a′)에서 주격 조사의 품사 표지를 갖는 "요한은"과 정렬될 수 있기 때문에 정렬되지 않은 채로 남아있는 낱말 묶음 "with camel's hair"와 "낙타털옷을"은 비롯 형식적인 문장 성분 관계에 있어서 수용하기 힘들다고 해도 서로 정렬되는 낱말 묶음이라고 추측할 수 있다.

(4′) a. [$_{\text{John}}$ John] [$_{\text{clothe}}$ was clothed] [$_{\text{camel,hair}}$ with camel's hair].

a′. [$_{\text{요한}}$ 요한은] [$_{\text{낙타,털,옷}}$ 낙타털옷을] [$_{\text{입다}}$ 입었다].

반면에 낱말 묶음 표지를 내용어로 대체한 (4′)에서는 형식적인 정렬에서 보이는 문제점이 쉽게 해결된다. 각각의 낱말 묶음에서 실질 형태소를 정렬의 기준으로 삼아서 "John"과 "요한"이, "clothe"와 "입다"가, "camel,hair"와 "낙타,털,옷"이 낱말 묶음을 정렬하는 열쇠가 된다.[4]

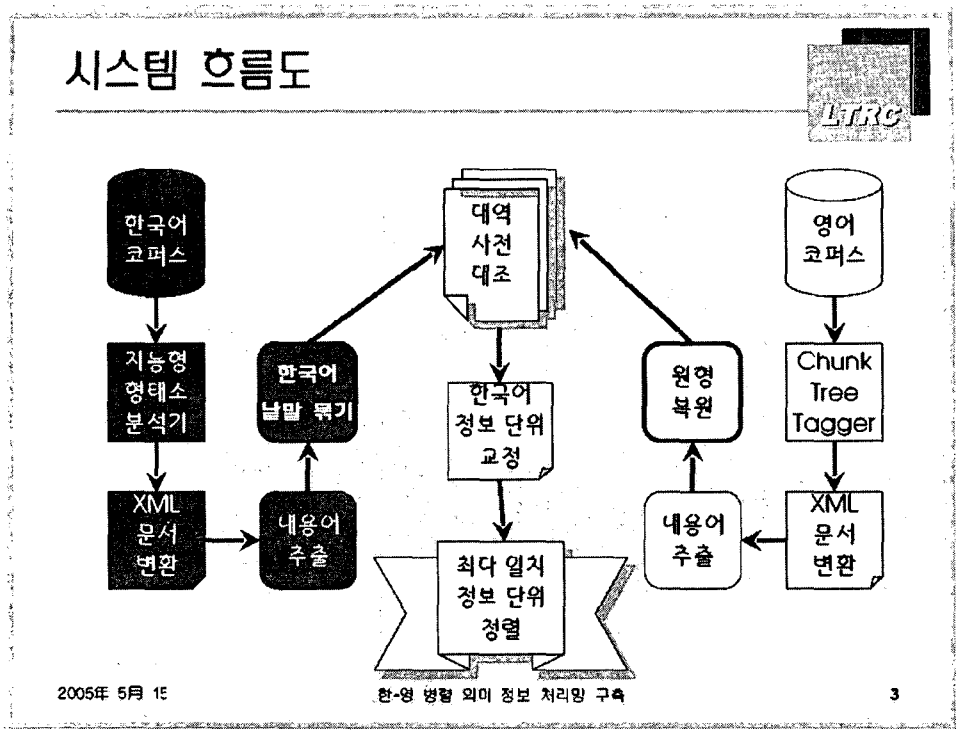
본 논문에서는 위와 같이 형식적인 분석으로 해결하기 어려운 문제점을 내용어 중심의 낱말 묶음 정렬로 보완하기 위해서 한국어 낱말 묶음의 핵심이 되는 두 형태소 부류로 동사와 명사를 선정하고 이 내용어를 중심으로 영한 낱말 묶음 정렬 과제에 응용한다. 이를 위해 영한 대역어 사전이 구축되었다.

한국어 낱말 묶기 모듈은 영한 낱말 묶음 정렬 프로그램의 일부로 구현되었다. 영한 낱말 묶음 정렬 프로그램은 영한 대역 코퍼스와 대역 사전 그리고 영어 기본형 굴절 사전을 이용하여 대역 코퍼스의 대역 문장을 낱말 묶음으로 분할한 후 각 낱말 묶음에서 내용어로 선정한 명사와 동사를 추출하여 이 내용어를 대역 사전을 통해 비교함으로써 대역 문장의 낱말 묶음을 정렬하는 프로그램이다. 영한 낱말 묶음 정렬의 시스템 흐름도는 그림 1과 같다. 본 논문에서 다루는 부분은 두꺼운 글자체로 표시된 "한국어 낱말 묶기"이다.

영어 코퍼스는 독일 Stuttgart 대학교 전산언어학과에서 개발한 Chunk TreeTagger를[5] 이용하여 형태소 분석과 동시에 낱말 묶기를 하고 분석 결과를 XML 문서로 저장한다. 한국어 코퍼스는 21세기 세종 계획의 결과물 중에서 지능형 형태소 분석기를 이용하여 형태소 품사 표지를 부착한 후에 품사 표지가 부착된 결과를 XML 문서로 저장한다. 영한 낱말 묶음 정렬 프로그램은 지정된 문장 식별 번호에 해당하는 영어 문장을 영어 코퍼스에서 검색하여 형태소 품사 표지를 기준으로 내용어를 추출하고 내용어의

[4] 한국어 문장에서 사용된 낱말 "옷"에 대한 영어 문장에서의 대역어는 "clothe" 혹은 "camel's hair"의 일부분으로 볼 수 있다. "clothe"의 대역어를 "옷을 입다"로 보면 "옷"은 "clothe"에 "camel's hair"의 대역어를 "낙타털옷"으로 보면 "옷"은 "camel's hair"에 포함된 의미이다. 본 논문에서는 "clothe"의 대역어를 "입다"로 보고 "camel's hair"의 대역어를 "낙타털옷"으로 보았다.

[5] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/에서 구할 수 있다. 그러나 낱말 묶기 프로그램에 대한 문서가 없고 실행 프로그램의 형태로 제공된다.



[그림 1] 영한 낱말 묶음 정렬의 시스템 흐름도

원형을 복원한다. 지정된 문장 식별 번호를 갖는 한국어 문장도 한국어 코퍼스에서 검색하여 내용어를 추출하고 낱말 묶기를 수행한다. 각 언어의 분할된 낱말 묶음이 가지는 내용어가 대역 사전을 통해 비교된다. 이같은 과정을 통해 정렬된 낱말 묶음 중에서 한국어 낱말 묶음을 영어 낱말 묶음과 일치시키는 교정 작업이 수행된 후에 가장 많은 내용어가 대응되는 낱말 묶음 쌍을 기준으로 최종 정렬 작업이 완료된다.

본 논문은 그림 1의 전체 시스템 흐름도 중에서 한국어 낱말 묶기 부분만을 다룬다. 즉, 한국어 형태소 품사 표지 부착 문장을 한국어 낱말 묶기 기준에 따라 낱말 묶음으로 나누고 대역 사전을 통해 영어 낱말 묶음과 정렬되는 과정에서 한국어 낱말 묶기 결과가 교정되는 단계가 이 논문에서 소개된다.

3.1 한국어 낱말 묶기 기준

영어의 낱말 묶기 기준은 전치사 및 명사, 문장 부호 등 동질의 낱말 묶음 경계이나 한국어는 조사 및 어미와 같이 문법 기능 형태소가 실질 형태소와 함께 한 어절을 구성하므로 원칙적으로 한국어는 어절을 하나의 낱말 묶음으로 보아야 한다. 그러나 한국어는

다음과 같은 점이 고려되어야 한다.[6]

1. 조동사(VX)는 본동사(VV, VA)와 함께 하나의 낱말 묶음을 형성한다.

   (5) 그리하여 그분의 소문은 곧 갈릴래아 인근 온 지방에 두루 퍼져 나갔다.

   김종복 (2004, 113-127쪽)에서는 한국어 구구조문법의 일부분으로 본동사와 보조 동사가 각각의 논항 구조를 가지고 있으나 마치 하나의 술어처럼 행동하여 복합 술어를 형성하는 것으로 보며 보조 동사는 본동사에 완성, 봉사, 시도, 추측 등의 의미를 추가하는 역할을 하는 것으로 본다. 장석진 (1993, 42-43쪽)에서도 보조 용언이 시상, 서법의 의미에 따라 분류되며 보조 용언 구문을 형성하는 것으로 본다. (5)에서 보조 동사로 사용된 "나가/VX+았/EP+다/EF"는 어떤 독립적인 정보를 가지고 있지 않고 본동사 "퍼지/VV+어/EC"가 계속됨을 나타내는 역할을 한다. 따라서 보조 동사와 본동사는 서로 결합하여 단일한 복합 술어 낱말 묶음을 형성하므로 보조 동사의 앞은 낱말 묶기의 경계로 보지 않는다.

   황영숙 (2002)에서도 이와 동일한 한국어 낱말 묶기 기준이 제시되었다.

2. 관형사형(ETM) 굴절을 보이는 동사(VV, VA)와 이를 뒤따르는 의존 명사(NNB)는 하나의 낱말 묶음을 형성한다.

   (6) 선생님은 하고자 하시면 저를 깨끗하게 하실 수 있습니다

   의존 명사는 그 운용상 독립성이 없는 형태소이며 최현배 (1999, 219쪽)에서는 안옹근 이름씨라고도 한다. 곧, 독립적으로 명사가 될 수 없고 앞선 관형사나 관형형 굴절을 보이는 동사 혹은 수식의 기능을 하는 명사와 함께 쓰인다. 따라서 (6)의 의존 명사 "수/NNB"는 독립적인 낱말 묶음을 형성하지 못하며, 앞선 관형형 전성 어미의 굴절을 보이는 동사 "하/VV+시/EP+ㄹ/ETM"과 함께 하나의 낱말 묶음을 형성한다.

   기존의 연구에서는 의존 명사와 같은 독립성이 없는 형태소를 고려하지 않았으나 형태소의 집합체로 구성되는 낱말 묶음의 분할 기준에서 형태소의 독립성 여부를 고려하는 것은 당연한 이치이다.

3. 의존 명사(NNB)를 뒤따르는 동사(VV, VA)는 하나의 낱말 묶음을 형성한다.

   (7) 선생님은 하고자 하시면 저를 깨끗하게 하실 수 있습니다

[6] 소괄호 안의 로마자는 지능형 형태소 분석기가 형태소에 부착하는 품사 표지이다.

대부분의 경우 의존 명사는 (7)과 같이 하나의 영어 대역어에 대응되는 구성으로 쓰인다. 이때의 구성은 의존 명사에 격표지가 붙지 않는 것이 대부분이다. 위의 경우는 영어의 조동사 "can"에 대응된다. 따라서 "수/NNB 있/VA+습니다/EF"와 같이 의존 명사 다음에 동사가 나오면 이를 하나의 낱말 묶음으로 본다.[7]

기존의 연구는 한국어라는 개별 언어의 테두리 안에서 낱말 묶음의 분할 기준을 설정하는 입장이었으나 본 연구는 병렬 코퍼스를 대상으로 낱말 묶기의 기준을 설정하며 영어의 단일한 형태소에 대응되는 한국어의 형태소 집합체를 하나의 낱말 묶음으로 봄으로써 한국어 낱말 묶기 기준이 좀더 보편 언어적인 접근을 가능하게 한다.

4. 관형격 조사(JKG)와 결합한 인칭 대명사(NP)와 이를 뒤따르는 명사(N)는 하나의 낱말 묶음을 형성한다.

(8) 예수께서 그의 집에서 음식을 드시게 되었는데 많은 세리들과 죄인들도 예수와 그분 제자들과 함께 상을 받았다.

인칭 대명사가 소유격 조사와 함께 쓰여 뒤따르는 명사를 수식할 때 인칭 대명사가 독립적으로 하나의 낱말 묶음을 형성한다고 보기 어렵다. 따라서 (8)의 "그/NP+의/JKG"는 뒤따르는 "집/NNG+에서/JKB"와 함께 하나의 낱말 묶음을 형성한다.

황영숙 (2002)에서는 관형격 조사 "-의"를 포함하는 명사구를 독립된 낱말 묶음으로 설정하였다. 그러나 대명사의 소유격은 그 쓰임에 있어서 독립된 어절로서 뒤따르는 명사를 수식하는 분포를 보인다. 황영숙 (2002)에서는 명사를 수식하는 관형어를 수식받는 명사와 함께 하나의 명사 낱말 묶음으로 설정하는데, 명사를 수식하는 관형어의 자질이 이와 같은 설정의 근거가 된다. 대명사의 소유격도 그 분포에 있어서 관형어와 유사한 자질의 역할을 하며 이같은 근거로 인칭 대명사의 소유격과 뒤따르는 명사를 하나의 낱말 묶음으로 설정할 수 있다.

구현 중심의 낱말 묶기 기준은 사용된 형태소 분석기에 전적으로 의존한다. 본 논문에서 사용한 형태소 분석기는 문화관광부·국립국어연구원 (2002)의 지능형 형태소 분석기이다. 형태소 분석기에 의존적인 것은 낱말 묶기 기준뿐만 아니다. 한국어 대역 문장의 낱말 묶기의 결과가 영어 낱말 묶기 결과와 내용어 중심으로 비교되면서 영한 정렬이 수행되며 이 과정에서 한국어 낱말 묶기 결과가 교정된다. 이때 수행되는 영한 정

[7] 이러한 유형의 또다른 예로는 "... 듯 하다" (seem ...), "... 것 같다" (is likely ...) 등이 있으나 대역 코퍼스로 선정된 마르코 복음에서는 사용되고 있지 않다.

렬은 대역 사전을 통해 한국어 낱말 묶음의 내용어와 영어 낱말 묶음의 내용어를 기본형의 형태로 비교하는 일이다.[8] 이때의 기본형은 형태소 품사 표지 부착기가 분할한 기본형을 말하며 사용된 형태소 분석기에 따라 형태소 분할 기준이 절대적으로 의존하게 된다.

한영 정렬을 위한 대역어 추출 기준은 내용어의 기본형을 대상으로 하며 다음과 같은 기본형 추출 기준을 추가로 적용한다.

1. 대명사(NP)와 결합하는 복수형 접미사(XSN) "들"은 대명사와 합한 문자열을 대역어로 추출한다.

   (9) a. 그들은 어부들이었다.
   a′. *they* were fishermen.
   b. 사람들은 그분의 가르침에 매우 놀랐다.
   b′. they were astonished at *his* teaching.
   c. 예수께서는 많은 귀신들을 쫓아내셨다.
   c′. he cast out many *demons*.

   한국어의 복수형 접미사 "들"의 사용은 수의적이나 인칭 대명사에 사용된 복수형 접미사는 대역어 선정에 있어서 중요한 역할을 한다. 대역 사전은 대역어 쌍이 기본형의 형태로 나열되는데, (9a,a′)의 경우에는 영어 대역어가 "they"이므로 한국어 대역어가 "그들"이어야 하지만 (9b,b′)의 경우에는 영어 대역어가 "he"이므로 한국어 대역어가 "그분"(혹은 "그")이어야 한다. 그러나 (9c,c′)의 경우에는 영어 대역어 기본형이 "demon"이며 이에 대응되는 한국어 대역어는 복수형 접미사 "들"이 없는 "귀신"이다.

2. 명사(N)와 결합하는 동사화 접미사(XSV) 혹은 형용사화 접미사(XSA)는 명사와 결합한 형태를 대역어로 추출한다.

   (10) a. 저분이 더러운 영들에게 지시하니 그들도 복종하는구나
   b. he *commands* even the unclean spirits, and they *obey* him

   (10a)의 "지시하니"는 "지시/NNG+하/XSV+니/EC"로 분석되며 "복종하는구나"는 "복종/NNG+하/XSV+는구나/EF"로 분석된다. 각각의 영어 대역어는 "command"와 "obey"이고 모두 동사의 품사 표지를 갖는다. 따라서 한국어 낱말 묶음에서 동사화 접미사나 형용사화 접미사와 결합하지 않은 형태의 한국어 동사성 명사만을 대역 사전에서 검색하면 품사가 동사인 영어 대역

[8] 내용어는 품사를 기준으로 하며 명사와 동사로 한정하였다.

어를 갖는 영어 낱말 묶음과 정렬되지 않는다. 이 문제점을 해결하기 위해서는 동사화 접미사 혹은 형용사화 접미사와 결합하는 동사성 명사에 특별한 속성을 부여할 수도 있고 경동사와 결합한 형태를 대역 사전에 등재할 수도 있다. 본 논문에서 택한 방법은 후자이다. 이는 지능형 형태소 분석기가 부착한 품사 표지에 동명사의 속성이 나타나 있지 않기 때문이다. 따라서 "지시하니"의 대역어는 "지시하다"이고 "복종하는구나"의 대역어는 "복종하다"이다.[9]

"하다", "받다", "되다", "시키다"의 경동사가 앞선 명사와 결합한 형태를 기본형으로 취하는 형태소로 본다.

3.2 대역 코퍼스 선정 및 한국어 대역 코퍼스의 가공

대역 코퍼스로 성경의 마르코 복음이 선택되었다. 한국어 성경은 가톨릭 200주년 기념 성서를 택하였고[10] 영어 성경은 Revised Standard version을 택하였다.[11] 이들은 인터넷을 통해 구할 수 있고 내용에 있어서 차이점이 적다는 것이 대역 코퍼스로서의 선택을 결정하는 중요한 요소가 되었다.

그러나 영어 성경과 한국어 성경은 둘 중 어느 하나를 상대 언어로 번역한 것이 아니고 히브리어 성경 원문을 각각의 언어로 번역한 것이기 때문에 번역문의 내용은 거의 일치하지만 표현 방법이 다를 수 있다.[12]

(11) a. As it *is written* in Isaiah the prophet, "Behold, I send my messenger before thy face, who shall prepare thy way.

a′. 예언자 이사야의 글에, "보라, 내 심부름꾼을 너보다 먼저 보내니 그가 네 길을 닦아 놓으리라.

(11a)의 "is written"에서는 쓰는 동작을 나타내는 동사 "write"가 사용된 데 반해 (11a′)에서는 "글"이라는 명사가 사용되었다. "쓰는 동작"의 결과물이 "글"이므로 의미에 있어서는 같은 표현이지만 영한 정렬의 관점에서는 어려움이 초래된다.

문장 나눔에 있어서도 두 대역 코퍼스 사이의 불일치가 발견된다.

---

[9] 경동사 처리로 인해 시스템의 복잡도와 대역어 사전에 드는 비용이 증가한다. 다른 방법을 사용한다면 (i) 일반명사의 표지(NNG)를 갖는 형태소에 영어의 동사와 명사 대역어를 등재하거나 (ii) 영어의 동사 대역어를 한국어의 경동사 구성에 맞추는 방법이 있을 수 있다. (i)이 경우, 시스템은 한국어의 명사("지시")와 영어의 대역어 명사를 대역 사전에서 검색하고 또한 영어의 대역어 동사를 대역 사전에서 검색하게 되며 경우에 따라서는 "command"와 같이 품사는 다르지만 형태가 동일한 경우에 의해 애매성이 발생하게 된다. 대역 사전에서도 한국어 명사에 대한 영어 동사 쌍을 등재하게 되므로 비용이 절감되지 않는다. (ii)의 경우, 영어의 동사 "command"를 형태소 수준에서 적당히 분할하여 "지시/NNG"와 "하/XSV"로 대역사전에서 검색하도록 하기가 쉽지 않을 것으로 보인다.

[10] http://bible.paolo.net/read.php3?b=200&t=48

[11] http://etext.lib.virginia.edu/toc/modeng/public/RsvMark.html

[12] 표현 방법이 상이한 문제는 성경에서 뿐만 아니라 원천 언어를 직접적으로 번역한 목표 언어와의 사이에서도 나타날 수 있다. 예를 들어 영어 문장의 수동형을 한국어의 능동형 문장으로 번역한다든가 영어 문장의 종속절을 연결어미를 사용하여 한국어로 번역할 때 번역문의 의미가 훨씬 또렷할 경우가 많다. 이러한 표현 방법의 차이를 극복하고 두 언어 사이의 의미 정렬을 수행하기 위해서는 영한 정렬 알고리듬이 어려운 과제를 해결해야 하는데, 이 논문에서는 이에 대한 논의를 생략하기로 한다.

(12) a. And immediately there was in their synagogue a man with an unclean spirit

a′. 마침 그 때 그들의 회당에 더러운 영에 사로잡힌 사람이 있었는데 그가 외쳐

한국어 문장 (12a′)의 "그가 외쳐"는 영어 코퍼스에서는 다음 문장에서 "and he cried out," 으로 나타난다.

또한 영어의 등위접속문이나 복문이 한국어에서는 별도의 문장으로 쪼개지는 경향이 있다.

(13) a. Now Simon's mother-in-law lay sick with a fever, and immediately they told him of her

a′. 그런데 시몬의 장모가 열이 나서 누워 있었다. 그래서 사람들은 즉시 부인의 사정을 예수께 말씀드렸다.

영어 번역문 (13a)은 하나의 등위접속문으로 되어 있으나 한국어 번역문 (13a′)은 두 개의 단문으로 처리되었다.

병렬 코퍼스의 정렬은 나누어진 문장을 대상으로 수행되므로 문장 나눔의 불일치를 해소할 방법은 없다. 그러나 표현 방식의 불일치는 정렬 알고리듬이 해결해야 할 과제로 보아야 한다.

한국어 코퍼스는 문화관광부·국립국어연구원 (2002)의 지능형 형태소 분석기를 사용하여 형태소 품사를 부착한 후 표 1과 같은 구조의 XML 문서로 저장되었다. XML 문서의 뿌리 노드는 `text`이며 각 문장이 뿌리 노드의 자식 노드 `sent`로 나열된다. `sent` 노드는 속성 `id`를 가지며 그 값은 장 번호와 절 번호가 마침표를 경계로 나뉜 문자열이다. 절 번호의 앞에는 문자 `v`가 앞선다. 문장 노드 `sent`의 자식 노드는 말단 노드인 `word`이다. `word` 노드의 속성은 어절의 일련 번호를 값으로 갖는 `id`와 미가공 어절을 값으로 갖는 `rawword`이다. `word` 노드의 내용이 형태소 품사 부착 어절이다. 이 형태소 품사 부착 어절에서 동사와 명사의 형태소 품사 표지를 갖는 형태소를 내용어로 추출하게 된다.

3.3 한국어 낱말 묶기

한국어 낱말 묶음 정렬 프로그램은 C 언어로 구현되었다. 주함수 내에서 영어와 한국어 낱말 묶음 분석 모듈의 주함수를 호출하고 대역 사전 열람 함수를 실행시킨 후에 낱말 묶음 정렬 함수, 낱말 묶음 정렬 결과 함수, 형태소/낱말묶음 분석 문장 가시화 함수를 차례로 호출한다. 이러한 함수들의 위계는 그림 2와 같다.

그림 2에서 한국어 낱말 묶기를 담당하는 함수들은 **다른 활자체**로 구별하여 표시되었다. 한국어 낱말 묶기의 일차 기능이 "한국어 낱말 묶음 분석 모듈"에 있으며 프로그

```
<?xml version="1.0" encoding="utf-8" ?>
<text>
...
<sent id="4.v36">
<word id="0" rawword="그 때서">그 때서/MAJ</word>
<word id="1" rawword="그 들은">그/NP+들/XSN+은/JX</word>
<word id="2" rawword="군 중을">군 중/NNG+을/JKO</word>
<word id="3" rawword="남 겨">남 기/VV+어/EC</word>
<word id="4" rawword="두 고">두/VX+고/EC</word>
<word id="5" rawword="배에">배/NNG+에/JKB</word>
<word id="6" rawword="타신">타/VV+시/EP+ㄴ/ETM</word>
<word id="7" rawword="예수 를">예수/NNP+를/JKO</word>
<word id="8" rawword="그 대도">그 대도/MAG</word>
<word id="9" rawword="모 시 고">모 시/VV+고/EC</word>
<word id="10" rawword="갔는 데">가/VX+았/EP+는 데/EC</word>
<word id="11" rawword="다른">다른/MM</word>
<word id="12" rawword="배들 도">배/NNG+들/XSN+도/JX</word>
<word id="13" rawword="함 께">함 께/MAG</word>
<word id="14" rawword="갔 다.">가/VV+았/EP+다/EF+./SF</word>
</sent>
...
</text>
```

[표 1] 한국어 코퍼스 XML 문서

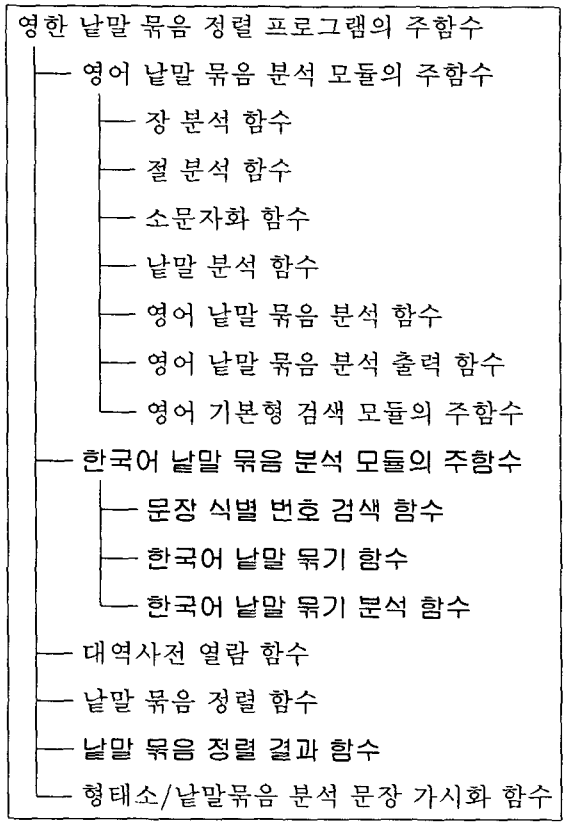
램의 주함수 내의 "낱말 묶음 정렬 결과 함수"를 통해 영한 낱말 묶음이 정렬될 때, 분할된 한국어 낱말 묶음의 마지막 교정이 수행된다.

3.4 한국어 낱말 묶기 결과

영한 낱말 묶음 정렬 결과는 웹 언어로 가시화되며 웹 브라우저를 통해 그림 3과 같은 모습으로 나타난다. 상단 프레임에서는 문장을 선택할 수 있으며 선택된 문장의 영한 낱말 묶음 정렬 결과가 하단 프레임에 나타난다. 낱말 묶기 결과는 하단 프레임에서 밤색 테두리를 두른 상자 안에 표시된다. 낱말 묶음의 분할 경계는 "■"나 "□" 기호로 표시되는데 두 언어 간에 대응 낱말 묶음이 있으며 "■" 표시가 해당 낱말 묶음을 앞서고 대응 낱말 묶음이 없으면 "□" 표시가 해당 낱말 묶음을 앞선다. 두 언어 사이에 정렬된 낱말 묶음은 마우스가 한 언어의 낱말 묶음 위를 지날 때 상대 언어의 낱말 묶음과 함께 바탕색이 노란색으로 바뀌면서 두드러지게 표시된다.

64쪽에 예시된 그림 3의 문장에서 한국어 낱말 묶음 정렬 결과는 (14)와 같다.

(14) □ 그리하여 ■ 그̊문̊의 소̊문̊은 □ 곧 ■ 갈릴래아 □ 인근 □ 온 ■ 지방에 □



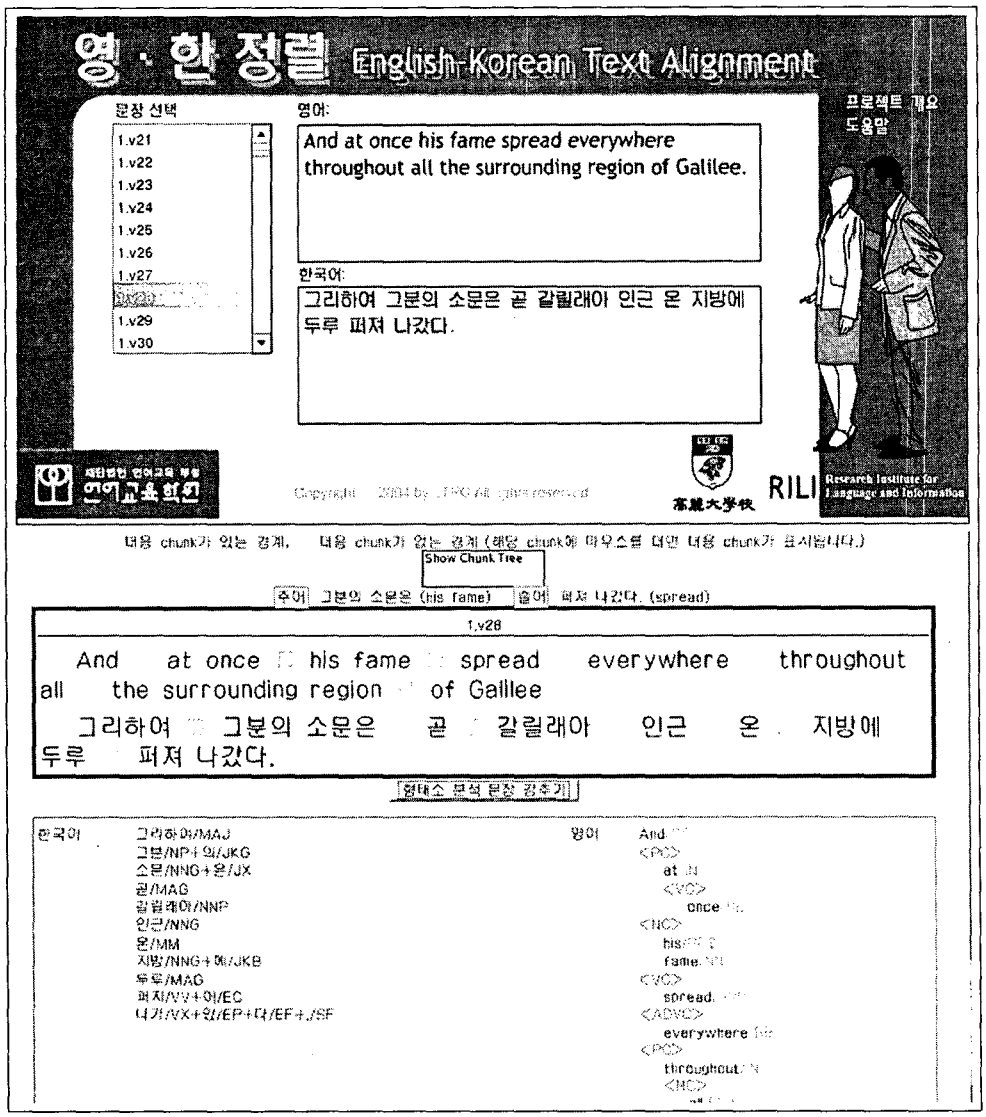
[그림 2] 영한 낱말 묶음 정렬 프로그램의 함수 위계

두루 ■ 퍼져 나갔다.

두 번째 낱말 묶음 "그분의 소문은"은 한국어 낱말 묶기 기준 중에서 소유 표지 "의"가 붙은 대명사 어절에 명사 어절이 뒤따르는 경우 두 어절이 하나의 낱말 묶음이 되는 기준에 의한 것이고 마지막 아홉 번째 낱말 묶음 "퍼져 나갔다"는 보조 동사와 본동사가 하나의 낱말 묶음을 이룬다는 기준에 따라 낱말 묶음이 분할된 결과이다. 좀 더 자세히 설명하면 "그분의 소문은"의 경우 그림 3의 형태소 분석 문장에서 보듯 "그분/NP+의/JKB 소문/NNG+은/JX"으로 형태소가 분석되며, 대명사 "그분/NNP"에 소유 표지 "의/JKB"가 붙고 다음 어절이 일반명사 "소문/NNG"이므로 이 두 어절 사이가 낱말 묶기의 경계가 되지 않도록 한다.

(15) a. ■ the voice ■ of *one crying* ■ in the wilderness □ : ■ Prepare ■ the way ■ of the Lord □ , ■ make ■ his paths □ straight □ –

a′. ■ 광야에서 ■ 부르짖는 이의 ■ 소리니라. □ '너희는 ■ 주님의 ■ 길을



[그림 3] 영한 낱말 묶음 정렬 결과의 가시화

■ 마련하고 ■ 그분의 □ 굽은 ■ 길을 □ 바르게 ■ 만들라!'" □ 고 □ 기록되어 있는 대로

(15a′)의 두 번째 낱말 묶음 "부르짖는 이의"는 한국어 낱말 묶기 기준에 의해 두 개의 독립된 낱말 묶음을 형성한다. 그러나 영어 낱말 묶음과의 정렬 과정에서 내용어 "부르짖다(cry)"와 "이(one)"가 영어의 두 번째 낱말 묶음에 대응됨으로써 연속되는 한국어 낱말 묶음이 하나로 통합되었다. (15a′)의 마지막 낱말 묶음 "기록되어 있는 대로"의 경우에는 한국어 낱말 묶기 기준에 의해 보조 동사와 본동사가 하나로 합쳐졌고 연이어서 관형형 전성 어미의 굴절을 보이는 동사와 뒤따르는 의존 명사가 하나로 합쳐졌다. (15a)의 마지막 부분에 있는 영어 낱말 묶음 "his paths"의 한국어 직역은 "그분의 길을"인데 (15a′)에서는 한국어로 "그분의 굽은 길을"로 번역됨으로써 한국어 낱말 묶기가 하나로 통합되지 못했다. 길을 바르게 만들기 위해서는 그 길이 굽은 상태이어야 하므로 한국어 번역문에서의 "굽은"의 첨가는 원문의 뜻을 명료하게 한다. 그러나 영한 정렬의 입장에서는 이러한 의미상의 첨가어가 처리되기 힘든 하나의 문제점으로 야기된다.

4. 결론

이 논문에서는 영한 정렬의 문제를 해결하기 위해 대역 문장을 낱말 묶음으로 나누고 내용어 중심으로 대역 사전을 통해 대응 낱말 묶음을 찾는 작업이 소개되었다.

프로그램으로 구현된 기존 연구가 없어서 본 논문의 결과를 비교할 대상이 없다. 김재훈 (2000)에서는 부분 구문 표지를 분류하는 연구에 그쳤고 황영숙 (2002)에서는 한국어 구문구조 부착 말뭉치를 기본구 표지 부착 말뭉치로 변형하여 사용하였다. 어떤 연구가 확률을 기반으로 프로그램을 구현했다면 프로그램에 사용된 확률 알고리듬의 가치를 판단할 수 있는 척도로 정확률이 중요한 역할을 한다. 그러나 규칙을 기반으로 하는 프로그램의 정확률은 100% 이하가 되어서는 안된다.

반면에 규칙 자체의 정확률을 살펴봐야 한다. 예를 들어 (15a′)에서 첫 번째 낱말 묶음으로 제시된 "광야에서"가 과연 하나의 낱말 묶음이라고 할 수 있는지를 재고해야 한다. 본 논문의 한국어 낱말 묶기 모듈은 영한 정렬을 위해 내용어를 중심으로 진행된다는 점에서 그 효율성이 입증된다. 한 언어의 일반화된 낱말 묶기라는 관점에서 보면 한국어 낱말 묶기 기준에 있어서 다음과 같은 점들을 고려하여 보완될 필요가 있다.

(16) a. ■ John □ the baptizer ■ appeared ■ in the wilderness □ , ■ preaching ■ a baptism ■ of repentance ■ *for the forgiveness* ■ of sins

a′. ■ 세례를 □ 베푸는 ■ 요한이 ■ 광야에 ■ 나타나 ■ 죄를 ■ 용서받기 □ 위한 ■ 회개의 ■ 세례를 □ 받으라고 ■ 선포하였다.

b. □ And □ immediately □ he ■ left ■ the synagogue □ , □ and □ entered ■ the house ■ of Simon and Andrew □ , ■ *with James and John*

b′. □ 그리고 □ 그들은 □ 곧 ■ 회당에서 ■ 떠나 ■ 야고보와 요한과 □ 함께 ■ 시몬과 안드레아의 ■ 집으로 □ 갔다.

(16a)의 "for the forgiveness"와 (16a′)의 "용서받기 위한"은 서로 대응되는 낱말 묶음이고 (16b)의 "with James and John"과 (16b′)의 "야고보와 요한과 함께"는 서로 대응되는 낱말 묶음이다. 두 영어 낱말 묶음은 전치사가 이끄는 낱말 묶음으로 각각 하나의 낱말 묶음으로 분할되지만 한국어의 경우에는 한국어 낱말 묶음이 각각 두 개의 낱말 묶음으로 분할된다. "○○를 위한" 혹은 "○과 함께"와 같은 숙어 표현을 처리할 낱말 묶기 기준이 필요하다.

(17) a. □ And □ there ■ went □ out ■ to him □ all ■ the country ■ of Judea , □ and □ all ■ the people ■ of Jerusalem □ ; □ and □ they □ *were baptized* □ by him ■ in the river ■ Jordan □ , ■ confessing ■ their sins

a′. □ 그래서 □ 온 ■ 유대 ■ 지방 □ 주민과 ■ 예루살렘 ■ 사람들이 □ 모두 ■ 그에게로 ■ 나가서 ■ 자기들의 죄를 ■ 고백하며 ■ 요르단 ■ 강물에서 □ 세례를 □ 받았다.

(17a)의 단일한 영어 낱말 묶음 "were baptized"는 (17a′)에서 두 개의 한국어 낱말 묶음 "세례를"과 "받았다"에 대응된다. 대응되는 한국어 낱말 묶음의 "받다"가 경동사로 사용되기도 하나 이 경우에는 "세례를"을 보충어로 취하는 본동사로 품사 표지가 부착된다. 따라서 현재의 한국어 낱말 묶기 기준에 의해 두 낱말 묶음으로 분할되는 것을 막을 길이 없다. 이러한 문제가 대역어를 "세례받다"로 선정함으로써 해결될 수 있다고 그 문제의 원인을 돌릴 수도 없다. 이러한 유형의 문제는 "장가가다, 장가를 가다, 시집오다, 시집을 오다"(marry), "구멍내다, 구멍을 내다"(break up), "잠자다, 잠을 자다"(sleep) 등의 여러 경우가 있다.

(18) a. ■ That evening □ , □ *at sundown* □ , □ they ■ brought □ to him □ all □ who □ were □ sick □ or ■ possessed □ with demons

a′. ■ 저녁이 □ 되어 □ 해가 □ 지자, □ 사람들이 ■ 앓는 이들과 귀신들린 이들을 □ 모두 □ 예수께 ■ 데려왔다.

(18)은 (17)과 유사하나 보충어를 취할 때에 보충어가 주어 역할을 하는 경우이다. "병들다, 병이 들다"(get sick), "소문나다, 소문이 나다"(rumor) 등이 이에 속한다.

이와 같이 개별 언어 자체 내에서의 낱말 묶기 기준보다는 보편 언어적인 측면에서 본 낱말 묶기 기준을 마련하면 영한 정렬과 같은 언어 처리에 합리적인 기준이 마련될 것으로 보인다. 이러한 기준은 한국어 낱말 묶기 알고리듬을 독자적으로 연구하여 해결될 문제로 보이지는 않고, 중간 언어와 같은 보편언어적인 이론의 연구와 함께 진행되어야 할 것 같다.

<참고문헌>

Abney, Steven. 1991. Parsing by Chunks. In Robert Berwick, Steven Abney, and Carol Tenny (eds.), *Principle-Based Parsing.* Kluwer Academic Publishers, Dordrecht.

Brill, Eric. 1993. *A Corpus-Based Approach to Language Learning.* Ph.D. thesis, University of Pennsylvania.

Buchholz, Sabine, Jorn Veenstra, and Walter Daelemans. 1999. Cascaded Grammatical Relation Assignment. In *EMNLP/VLC-99*, USA. University of Maryland.

Miller, George A. 1956. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychological Review* 63, 81–97.

Ramshaw, Lance A. and Mitchel P. Marcus. 1995. Text Chunking Using Transformation Based Learning. In *The Third ACL Workshop on Very Large Corpora*, Cambridge MA, USA.

김미영, 강신재, 이종혁. 2000. 단위 분석과 의존문법에 기반한 한국어 구문분석. *제27회 한국정보과학회 봄 학술대회* 발표 논문집에서, 327-326쪽. 한국정보과학회.

김재훈. 2000. 한국어 부분 구문분석의 단위와 그 표지. 기술문서, 한국해양대학교 컴퓨터공학과. KMU-NLP-TR-2000-006.

김종복. 2004. *한국어 구구조문법.* 한국문화사.

김창제, 정천영, 김영훈, 서영훈. 1995. 부분적인 어절 결합을 이용한 효율적인 한국어 구문 분석. *제18회 한국정보과학회 가을 학술대회* 발표논문집에서, 597-600쪽. 한국정보과학회.

문화관광부·국립국어연구원. 2002. 21세기 세종계획 연구보고서. 기술문서, 문화관광부.

박상규, 정창민, 조준모, 이상조. 1995. 최장 묶음을 이용한 한국어 구문 분석기. *제17회 한국정보과학회 봄 학술대회* 발표논문집에서, 961-964쪽. 한국정보과학회.

박성배, 장병탁, 김영택. 2000. k-NN으로 확장된 한국어 단위화. *정보과학회 가을 학술* 발표 논문집에서, 182-184쪽. 정보과학회.

신효필. 1999. 최소자원 최대효과의 구문분석. *제11회 한글 및 한국어 정보 처리 학술대회* 논문집에서, 242-247쪽.

안동언. 1987. 기계번역을 위한 한국어 해석에서 형태소로부터 구문요소의 형성에 관한 연구. 석사학위 논문, 한국과학기술원 전산학과.

이공주·김재훈. 2003. 규칙에 기반한 한국어 부분 구문분석기의 구현. *정보처리학회논문지 B* 10-B.4, 389-396.

장석진. 1993. *정보기반 한국어 문법.* 도서출판 언어와 정보.

최현배. 1999. *우리말본*, 열여덟번째판. 정음문화사.

황영숙. 2002. *자연어 처리를 위한 2단계 적합자질 선택 기법과 가중적 확률 결합 모델*. 박사학위 논문, 고려대학교.

접수 일자: 2005년 11월 15일
게재 결정: 2005년 12월 18일