통제불능 상태를 회피하는 한국어 정보처리 방법론 연구

A Study on the Methodologies of Korean Language Processing Avoiding Dead-end State

강 승 식*
(Seung-Shik Kang)

ABSTRACT

It is relatively easy to develop a prototype of a Korean language processing system, but it is very difficult to make it an operational system. In this paper, we survey the current status and methodological issues of the Korean language processing systems such as morphological analyzer, parser and machine translator. In most cases, Korean language processing system easily comes to a dead-end state where its performance can not be improved any more. The reason is that it adopts a general algorithm covering similar problems as a whole because specific low-level problems are not clearly defined and their algorithms are unclear. So, when we add some restrictions to solve an individual linguistic problem, they are also applied to other linguistic phenomena as a side effect. It causes a critical problem that the improvement of the algorithm is very difficult. This paper proposes a 2-step paradigm, a divide-and-conquer method by the functional modularization, a simplification method, and an exception handling technique to develop an operational system that does not fall into a dead-end state.

Keywords: parsing, 2-step paradigm, divide-and-conquer, dead-end state

1. 서 론

자연언어 처리 분야는 1940년대 후반부터 기계번역을 중심으로 매우 활발하게 연구되었으나 1964년 ALPAC 보고서에서 '실현 가능성' 문제가 지적된 것을 기점으로 한동안 침체기를 맞이하기도 하였다(Hutchins, 1986). 그러나 1970년대 말부터 철자 검사기가 실용화되고 정보검색 시스템에서 대용량 정보자료를 색인하는 데 활용되면서 다시 주목을 받기 시작하였다. 이와 같이 자연언어 처리를 비롯한 소프트웨어 분야는 실현 가능성 및 연

* 한성대학교 이공대학 정보전산학부

구-개발의 필요성에 의해 활발하게 연구가 진행되었다가 연구 결과가 응용 시스템이나 사용자들의 요구사항을 충족시키지 못할 때 침체기를 맞이하기도 한다.

기계번역으로 대표되는 자연언어 처리 분야는 그 속성상 시제품 개발이 비교적 용이한 반면에 사용자들의 요구사항을 충족시킬 수 있는 실용적인 시스템으로 발전하기가 매우 어렵다(Nirenburg, 1987). 그 이유는 매우 다양한 언어현상들 중에서 적용범위가 넓은 소수의 언어현상들은 쉽게 규칙화될 수 있지만, 적용범위가 좁은 다수의 개별적인 언어현상들은 발견 및 규칙화가 쉽지 않기 때문이다. 특히, 유사한 언어현상들에 대해 적용범위가 넓은 일반적인 규칙(general rule)과 적용범위가 좁은 개별적인 규칙(specific rule)간에 제약조건이 상충되는 경우가 빈번하여 시제품에서 실용적인 시스템으로 발전할 때 알고리즘의 복잡도가 비선형적으로 증가하는 문제가 있다(Barton, et. al, 1987)[1)]

국내에서는 1980년대 중반부터 기계번역이라는 응용 시스템을 중심으로 연구가 시작되어 영한, 일한 기계번역이 실용화되었으나, 사용자들의 요구를 만족시키기에는 부족한 점이 많다(김영택, 1994; 김태석, 1997). 한국어 정보처리는 1980년대 후반부터 형태소 분석과 구문분석, 말뭉치 구축 등에 대한 기초연구가 시작되었고, 형태소 분석 기술은 맞춤법 검사와 자동색인 등에 활용되고 있다(권혁철, 1994). 현재, 형태소 분석보다 적용범위가 훨씬 넓은 구문분석 기술을 개발하는 연구가 활발하게 진행되고 있으며, 구문분석 기술이 실용화되면 한국어 정보처리 분야의 발전에 미치는 영향이 매우 클 것으로 예상된다. 즉, 한국어 정보처리 분야에서 구문분석 기술이 차지하는 비중이 매우 크다.

한국어 정보처리 기술은 기계번역과 정보검색 등 일부 응용 분야에서 바로 활용되기도 하지만 문법검사(grammar check), 문서 요약(text abstraction), 자연언어 인터페이스 등 다양한 응용 분야에서 핵심적인 역할을 한다. 특히, 기계번역 기술을 이용한 자동통역, 운영체제 등 광범위하게 활용 가능한 자연언어 인터페이스, 키보드가 필요없는 컴퓨터, 필기체 인식 컴퓨터, 인간 수준의 고품질 음성 합성 및 음성 인식, 멀티미디어 정보검색, 지식처리 시스템 등 차세대 첨단 소프트웨어 분야에서 한국어 정보처리 기술이 필수적이다.

이처럼 다양한 응용 분야에서 한국어 정보처리 기술이 활용되려면 실용적인 한국어 정보처리 기술이 개발되어야 한다. 시제품에서 실용적인 기술로 발전할 때 가장 큰 장애요인이 되고 있는 통제불능 상태를 회피하기 위하여 한국어 분석 기술의 현황과 문제점을 고찰하고 이를 극복하기 위한 방법론을 제안한다.

2. 한국어 정보처리의 발전 과정

한국어 정보처리의 발전 과정을 살펴보면 1980년대 초반에 한글 입출력 문제와 한글 프로그래밍 언어, 기계번역 등을 중심으로 연구가 시작되어 시제품을 개발하고 실현 가능성을 확인하는 기초연구가 수행되었다. 1980년대 후반부터는 형태소 분석과 전자 사전 등 기초기술에 대한 연구가 활발하였으며, 이를 기반으로 자동색인과 기계번역 등 응용기술이 개발되었다. 2000년대에는 구문분석을 중심으로 국어정보베이스 구축 등 기초기술이

1) 규칙 충돌 문제를 해결하기 위해 통계적 기법을 사용하기도 하지만, 실용적인 시스템을 개발하기에는 제약이 많다.

한 단계 더 높은 수준으로 발전함으로써 이를 기반으로 다양한 응용 시스템이 개발되어 한국어 정보처리 분야가 매우 성숙될 것으로 예상된다. 한국어 정보처리의 발전 과정을 시기별로 구분하면 다음과 같다.

(1) 태동기(1980년~1984년) : 한글 입출력

대형 컴퓨터의 단말기와 개인용 컴퓨터 등에서 한글 문서를 작성하고 프린터로 출력하는 기능이 구현된 시기이다. 컴퓨터 제조업체를 중심으로 한글 입력 오토마타와 한글 폰트 등이 개발되었고, 형태소 분석과 구문분석 등 기초 연구가 시작되었다. 한글 입출력 시스템은 산업체를 중심으로 연구-개발되었으며, 한글 프로그래밍 언어와 형태소 분석, 구문분석 등은 개략적인 알고리즘을 고안하는 연구가 수행되었다.

(2) 준비기(1985년~1989년) : 기계번역 시제품

맞춤법 검사와 기계번역 등 각 분야별 특성에 따라 응용기술을 중심으로 연구가 이루어졌으며, 실용적인 시스템이 개발될 수 있다는 가능성을 확인했던 시기이다. 기계번역 시스템과 맞춤법 검사기의 실용화 가능성이 제시되었으며, 기계번역 시스템에서 요구되는 형태소 분석과 구문분석에 관한 연구가 수행되었다. 즉, 기계번역 시스템은 한국어 정보처리 연구에 대한 동기를 부여하는 계기가 되었다.

(3) 도약기(1990년~1994년) : 맞춤법 검사기

전자 사전이 구축되고 기계번역과 맞춤법 검사 등이 시제품 수준에서 실용적인 시스템으로 발전할 수 있는 체계를 갖추었다. 다양한 언어현상을 파악하기 위하여 대용량 말뭉치를 구축하기 시작하고 기초기술에 대한 체계적인 연구가 수행되었다. 특히, 한국어 형태소 분석과 영한, 한영 등 기계번역에 대한 연구가 매우 활발하였고, 그 결과로 한글 맞춤법 검사기와 일한 기계번역 시스템이 상용화되는 수준에 이르렀다.

(4) 성장기(1995년~1999년) : 형태소 분석, 기계번역, 말뭉치

인터넷과 정보검색 분야가 활성화됨에 따라 형태소 분석 기술이 각광받은 시기이다. 형태소 분석 기술이 정보검색 시스템에서 활용됨으로써 파급 효과가 매우 컸으며, 영한 기계번역이 실용화되어 자연언어 처리 분야에 대한 사용자들의 관심을 불러일으키는 역할을 하였다. 또한, 한국어 정보처리 기술을 체계적으로 연구하거나 실용적인 시스템을 개발할 수 있는 기반을 마련하기 위하여 대용량 말뭉치와 분야별 전문용어 사전을 구축하기 위한 노력이 매우 활발하다.

(5) 성숙기(2000년 이후) : 구문분석, 기계번역 등

2000년 이후에는 한국어 정보처리 기술이 컴퓨터 소프트웨어 분야를 발전시키는데 핵심적인 역할을 하게 될 것으로 예상된다. 인터넷과 전자거래가 보편화되면서 주제어 추출과 문서 요약 등 응용 분야가 확대되고 있으며, 다국어 검색 등 언어 장벽을 해소하는데 기계번역 기술이 활용될 예정이다. 특히, 음성인식 기술이 발전함에 따라 컴퓨터를 비롯한

대부분의 기계들을 작동시킬 때 자연언어 인터페이스가 보편화될 것으로 예상되고 있다. 이를 실현하기 위해서는 한국어 정보처리 기술이 필수적이며, 구문분석과 기계번역 기술 등 한국어 정보처리 기술은 차세대 소프트웨어를 개발하는데 매우 중요한 역할을 하게 될 것이다.

3. 한국어 정보처리 기술의 현황

한국어 정보처리의 핵심 기술은 한국어 분석 및 생성 기술이다. 영어의 경우 수십년 동안 축적된 기초자료와 기반기술을 바탕으로 기계번역이나 자연어 분석 기술이 많은 발전을 해왔다(Lehnert, 1982). 그러나 한국어 정보처리는 불과 십여 년이라는 비교적 짧은 기간 동안에 기초자료 및 기초연구가 미흡한 상태에서 연구가 진행되고 있다. 한국어 정보처리의 현황을 기초기술과 응용기술로 구분하여 살펴보면 다음과 같다.

3.1 기초기술 현황

한국어 정보처리 분야는 한국어의 형태-통사론적 특성 연구와 기초자료 등 전산언어학에서 바로 활용할 수 있는 자료를 습득하기가 용이하지 않아서 기초기술을 확보하는데 어려움이 많은 편이다. 또한, 상대적으로 연구 인력 등 연구 환경이 취약했기 때문에 영어나 일본어 등 선진국에 비해 기초기술에 대한 광범위한 연구보다는 주로 응용기술을 중심으로 연구가 이루어져 왔다. 특히, 한국어가 교착어이고 비구조적(non-configurational language) 언어라는 특성은 형태소 분석과 구문분석 기술의 발전을 더디게 한 가장 큰 요인이 되고 있다. 현재, 한국어 정보처리의 기초기술을 영어와 비교하면 표 1과 같다.[2)]

표 1. 한국어 정보처리 기초기술 현황

	영 어	한국어
기초 연구	◎	○
형태소 분석	◎	○
구문 분석	□	△
의미 분석	△	△
말뭉치 구축	◎	○

◎ : 응용 및 활용　　○ : 연구 및 활용
□ : 연구 및 부분 활용　　△ : 연구 단계

전산언어학의 관점에서 한국어의 특성을 분석하고 다양한 언어현상들을 파악하는 '기초연구' 분야는 한국어 분석 기술과 응용기술이 발전하는데 중요한 역할을 한다. 그러나 기초기술이나 응용기술을 개발할 때 직접 활용하기에 미흡한 점이 많으며, 한국어 정보처리 연구자들은 경험적이고 단편적인 지식에 의존하기도 한다. 이는 한국어 정보처리에 관

2) 개략적으로 비교한 것으로 구체적인 비교-평가 항목에 따라 달라질 수도 있다.

한 연구를 분석적이고 체계적으로 수행하기 어려운 가장 큰 원인이기도 하다.

한국어의 형태소 분석 기술은 실용적인 수준에 이르렀으며 타언어와 유사한 수준이라고 할 수 있다. 그러나 대화체 어휘의 분석과 오류어에 견고한 형태소 분석, 조선어와 북한어의 분석, 형태론적 모호성 해결 등 한국어의 특성과 관련된 문제들이 남아 있다. 구문분석은 형태소 분석 결과에 의존하는 특성과 다양한 문장 유형들을 고려해야 하는 어려움으로 인해 타언어와 비교할 때 가장 격차가 크다.

말뭉치와 전자사전의 구축은 한국어 정보처리 기술을 발전시키는 기반이 되고 있으나, 작업 규모가 크고 다양한 요구사항들을 고려해야 하는 어려움이 있다. 전자사전의 경우 구문분석이나 의미분석에 활용될 수 있도록 구체적이고 체계적인 정보들을 구축해 나가야 하며, 말뭉치의 구축도 언어현상을 발견하거나 검증하는 등 기초연구를 수행하는데 충분한 수준으로 발전되어야 한다(노용균 · 박동인, 1994).

3.2 응용기술 현황

한국어 정보처리 응용기술의 현황을 고찰하기 위해 대표적인 응용 소프트웨어들의 개발 시기를 살펴보면 다음과 같다. 1992년에 맞춤법 검사기가 실용화된 것을 시작으로 1994년부터는 형태소 분석 기술이 정보검색 시스템의 자동색인 기능으로 활용되고 있다. 기계번역의 경우는 일한, 한일 기계번역이 인터넷 정보검색을 비롯한 번역 분야에서 활용되고 있고, 영한 기계번역은 실용화 단계에 와 있다. 한국어와 영어의 응용기술이 개발된 시점을 비교하면 표 2와 같다.

표 2. 한국어 정보처리 응용기술 현황

	영 어	한국어
연구 시작	50년대 초	80년대 초
맞춤법 검사	70년대 말	90년대 초
문법 검사	80년대 중	?
자동 색인	80년대 초	90년대 초
문서 요약	90년대 중	?
기계 번역	80년대 중	90년대 중

선진국에 비해 비교적 짧은 연구기간 동안에 맞춤법 검사와 자동색인, 기계번역 등 여러 가지 응용 기술이 개발되었으나, 문법검사와 문서요약 등 구문분석 기술이 요구되는 응용 분야에서는 격차가 크다. 맞춤법 검사와 자동색인은 한국어 정보처리 기술 중에서 가장 실용적인 수준에 근접해 있으며, 이는 형태소 분석 수준에서 실용화가 가능하기 때문이다. 그러나 선진국 수준과 동일한 수준에 이르기 위해서는 각 응용 분야의 특성에 따라 세부적인 기술이 개발되어야 한다. 맞춤법 검사, 자동색인과 더불어 기계번역 기술의 경우에도 질적인 면을 고려할 때 선진국과는 차이가 있다.

문법검사의 경우 영어에서는 이미 보편화되어 있지만 한국어는 소수 정형화된 유형들

만이 활용되고 있으며, 문서 요약이나 문서 분류는 영어와 일본어가 실용적인 수준에 근접하는데 비해 한국어는 매우 초보적인 수준이다. 그 이유는 이 응용 기술들이 구문분석을 토대로 하고 있는데 비해 한국어 구문분석 기술이 실용적인 수준에 이르지 못했기 때문이다. 따라서 구문분석 기술이 요구되는 응용 기술이 발전하려면 구문분석 기술의 개발과 함께 응용 기술에 관한 연구를 병행하여야 한다. 자동색인의 경우에도 영어에서는 구문분석 기술을 활용하여 문장의 구조적 특성을 파악하는 주제어 추출까지도 가능하지만, 한국어는 형태소 분석에만 의존하고 있어서 질적인 면에서 차이가 많다.

기초기술이 타언어와 많은 격차가 있는데 비해 철자 검사와 자동색인, 기계번역 등 일부 응용 분야는 비슷한 수준이라고 할 수도 있다. 그러나 각 응용기술의 다양성과 질적인 면을 고려하면 그 격차는 클 것이라고 판단된다. 특히, 구문분석을 비롯한 기반 기술이 취약한 점 때문에 응용기술의 발전이 늦어지고 있다. 전반적으로 볼 때, 구문분석 등 기초기술의 부족으로 체계적인 연구를 수행하는데 어려움이 있으며, 상대적으로 경험과 직관에 의존하는 경우가 많아서 응용 기술이 발전하는데 어려움이 있다. 이러한 여건속에서 한국어 정보처리 기술은 전반적으로 선진국에 비해 대략 5~20년 정도의 격차를 보이고 있는 것으로 추정된다.

4. 한국어 정보처리의 문제점

한국어 정보처리 기술은 지난 10여년 동안의 연구 수행 과정을 통해 여러 가지 문제점들이 발견되고 있다. 그 중에서도 특히 다양한 언어현상들을 발견하기가 어려운 문제, 지속적이고 심도있는 연구를 수행하기 위한 연구 체계의 미흡, 기초기술과 응용기술의 조화 등이 한국어 정보처리 시스템이 한 단계 더 높은 수준으로 발전하기 위해서 극복해야 할 과제로 지적된다.

선진국에서는 사전을 구축하거나 언어현상을 발견하고 규명하는데 오랜 기간 동안 연구가 수행되었으며, 구문분석 기술을 개발하기 위해 십여년 이상 집중적으로 연구를 수행하고 있다. 이에 비해 한국어 정보처리 분야는 단기적인 성과 위주로 연구가 수행됨으로써 지속적으로 세부적인 문제점들을 해결하는 연구를 수행하기 어려운 실정이다.

4.1 언어정보의 구축 및 활용

형태소 분석이나 구문분석 등 한국어 정보처리에서 필요한 정보들은 각 기능별 관점에서 과학적-분석적인 방식으로 정립되어야 한다. 어휘 사전의 예를 들면, 용도면에서 기존의 사전과 많은 차이가 있다. 기존의 사전이 주로 단어의 의미를 알기 위한 목적으로 사용되는데 비해, 자연언어 처리에서는 언어를 분석 혹은 생성하기 위한 목적으로 사용되기 때문이다.

기계번역에서는 많은 용어가 수록되는 것이 타당하지만, 형태소 분석시에는 저빈도어가 사전에 수록됨으로 인하여 중의성 발생률이 높아져서 오히려 분석 성능을 저하시키는 경우가 발생하기도 한다. 이와 같이 세부 기능별로 어휘의 범위나 수록되는 정보에 따라 요구사항이 다르기 때문에 언어정보는 분석이나 생성, 중의성 해결, 기계번역 등 시스템의

성능 향상에 적합한 형태로 구축되어야 한다.

한국어 분석이나 생성을 위해서는 한국어의 단어 유형이나 단어 구성 전이도, 접두사와 접미사 유형, 문장 구조 등 기초적인 언어 지식에 대한 연구가 필수적이다. 이에 관한 많은 연구가 있었으나 한국어 분석/생성 기술을 개발하는데 필요한 구체적이고 명확한 언어정보를 습득하는데 활용하기에 미흡한 점이 많다. 한국어의 분석 및 생성 기술이 발전하려면 한국어의 형태-통사론적 특성에 대한 구체적인 정보자료의 수집이 가능해야 하며, 이를 위해서는 대량의 말뭉치가 구축되어 한국어의 다양한 언어현상들을 쉽게 발견할 수 있어야 한다.

4.2 분석적-체계적인 연구 체계

한국어 정보처리 기술을 개발하고 실용화하기까지는 많은 어려움이 따른다. 형태소 분석이나 구문분석, 의미분석 등 각 기술의 난이도에 따라 차이가 있지만, 시스템의 성능을 향상시키는데는 한계가 있다. 시스템이 어느 정도 안정된 수준에 이르면 개별적인 언어현상들을 처리하는데 많은 노력을 기울이더라도 전체 시스템의 성능에 미치는 영향이 거의 없는 포화(saturation) 상태에 이르게 된다. 전체 시스템의 성능을 100%라 하고 포화 상태에 이르기까지 단계별로 동일한 시간과 비용을 투자했을 때 성능 개선 효과는 아래와 같이 추정된다[3].

1 단계 : 초기 성능 70~90 %
2 단계 : 5~20 % 성능 개선
3 단계 : 3~10 % 성능 개선
4 단계 : 1~5 % 성능 개선

2단계 이후의 성능 개선 효과가 1~20 %에 머무는 이유는 언어현상의 다양성과 문서내 출현 빈도 때문이다. 한국어 정보처리 시스템을 개발할 때는 자주 출현하는 유형들을 우선적으로 처리하게 되는데 흔히 발견되는 유형들이 70~90 %를 차지하고 있다. 이러한 고빈도 유형들을 처리하기는 어렵지 않으나, 그외 저빈도 유형들은 발견하기도 쉽지 않을 뿐더러 고빈도 유형과 충돌이 발생하는 경우가 많다. 따라서 저빈도 유형으로 시스템을 확장할 때 고빈도 유형과의 충돌현상을 피하면서 성능을 개선하기가 쉽지 않으며, 특이한 언어현상들은 예외처리가 요구되기도 한다.

일반적으로 2~4 단계의 과정을 거쳐 시스템의 성능이 포화상태에 도달하기까지 형태소 분석의 경우 약 99 % 이상의 성능을 기대할 수 있지만, 구문분석이나 기계번역은 70~80 %일 것으로 추정된다. 성능 개선 효과는 개발 방법론에 따라 차이가 있을 수 있으며, 포화 상태의 시스템 성능 또한 방법론에 따라 가변적이다. 이러한 문제점은 언어현상들이 실제 문서에서 출현하는 빈도와도 밀접한 관련이 있다.

한국어 정보처리에서는 모든 언어현상을 포괄할 수 있도록 시스템을 설계하기가 매우

3) 기계번역이나 구문분석 등 알고리즘이 복잡한 기능을 중심으로 추정한 것이며, 통계적 태거와 같이 포화상태에 쉽게 도달되는 경우에는 적용되지 않을 수도 있다.

어렵다. 따라서 자주 출현하는 언어현상들을 중심으로 시스템을 개발한 후에 새로운 언어현상들을 추가하는 점진적인 방법론을 취하고 있다. 새로운 언어현상들이 추가되면 처리방법이나 자료구조 등이 달라지는 경우가 발생하기도 한다. 새로 발견된 개별적인 언어현상을 추가하면 새로운 언어현상은 처리되지만 기존의 보편적인 언어현상들이 처리되지 못하는 부작용(side effect)이 발생하여 더 이상 시스템을 발전시키기 어려운 '통제불능' 상태(dead-end state)에 빠지기가 쉽다.

새로운 언어현상을 추가하기 어려운 통제불능 상태에 빠지는 이유는 문제의 해결 범위와 해결 방법을 분석적이고 체계적으로 접근하기 어려운 자연언어 처리의 속성 때문이다. '통제불능' 상태에 빠진 시스템은 더 이상 발전 가능성이 없으므로 기존의 시스템을 폐기하고 더 다양한 언어현상들을 포괄하도록 처음부터 다시 설계하여야 한다. 그러나 일반적으로 시스템을 재설계하기는 쉽지 않으며 위험 부담이 크다. 따라서 시스템을 처음 설계할 때와 성능 개선 과정에서 기능별 모듈화 및 점진적인 발전 가능성을 중점적으로 고려해야 한다.

4.3 기초기술과 응용기술

자연언어 처리 분야는 분석-생성 기술, 사전 구축 등 기초기술과 이를 활용하여 응용 소프트웨어를 구현하는 응용기술로 구분된다. 자연언어 처리 연구는 기계번역과 자연언어 이해 시스템이라는 응용기술로부터 시작되었다. 자연언어 이해 시스템은 실현 가능성이 매우 낮다고 판단되고 있으며, 기계번역 또한 언어장벽을 해소할 것이라는 초기의 기대치에는 훨씬 못미치고 있다. 그러나 번역 업무를 비롯한 특정 분야(domain-specific)에서 번역 전문가의 수작업을 부분적으로 대신하는 기능으로 활용되고 있으며, 성능이 개선됨에 따라 활용 범위가 확대될 것으로 예상된다[4].

마찬가지로 한국어 정보처리 기술은 그 속성상 완벽할 수 없기 때문에 응용 시스템의 요구사항을 100 % 만족시켜 줄 수는 없다. 따라서 그 당시의 기술 수준에서 각 응용 분야의 요구사항을 어느 정도까지 만족시킬 수 있을 것인지가 한국어 정보처리 분야의 지속적인 발전에 미치는 영향이 매우 크다. 이 때 발생할 수 있는 문제점 중 하나는 분석-생성 등 기초기술이 취약한 상태에서 응용기술에 치중할 때 발생하고 있다. 형태소 분석이나 구문분석, 전자사전 구축 등 기초기술이 미비한 상태에서 응용 소프트웨어의 개발을 추진했을 때 곧 통제불능 상태에 도달하기 때문이다.

특히, 기초기술은 그 응용 분야가 다양하므로 특정 응용 기술을 중심으로 연구되었을 때 다른 응용 분야에 적용하기 어려운 제약이 있다. 따라서 자연언어의 분석 및 생성, 사전 구축 등 기초기술은 다양한 응용 분야에 적용될 수 있도록 핵심적인 기능을 중심으로 개발되어야 하며, 응용 분야에 따라 요구사항이 조금씩 다를 때 적응할 수 있도록 확장성을 충분히 고려해야 한다. 한국어 정보처리 기술은 정보검색, 문자인식, 음성인식 등 다양한 응용 분야에서 언어처리 문제를 해결하는 요소기술로서 활용되기 때문에 각 응용 분야의 언어처리 요구사항을 해결하는 역할을 충실히 수행할 수 있어야 한다.

---

4) 특히, 기계번역은 실생활에 미치는 영향이 매우 크기 때문에 실현 가능성이 불투명하더라도 이를 실현하기 위해 꾸준히 노력할 만한 가치가 있다.

5. 한국어 정보처리 방법론

한국어 정보처리 기술은 실험실 수준의 시제품을 개발하는 수준에서 시작하여 여러 응용 분야에서 활용될 수 있는 실용적인 시스템이 요구되고 있다. 시제품 단계는 기본적인 실험에 통하여 실현 가능성을 검증하는 작업이고, 실용화 단계는 문제해결 방법론을 정립하여 다양한 언어현상을 포괄할 수 있도록 구현하는 작업이다.[5] 그런데 한국어 정보처리 시스템은 그 속성상 시제품을 개발하는 일은 어렵지 않은 반면에 실용적인 시스템으로 발전시키기가 쉽지 않다.

시스템이 어느 정도 안정된 상태에서 1 %의 성능을 향상시키는 것은 초기 시스템의 성능을 10 % 향상시키는 것보다 훨씬 어렵다. 어떤 경우에는 부분적인 시스템의 성능을 향상시키는 노력에 의해 전체적으로 성능이 저하되는 부작용(side effect)이 발생하기도 한다. 이러한 부작용은 적용 범위가 좁은 언어현상을 처리하는 모듈이 적용 범위가 넓은 언어현상과 충돌할 때 발생하는데, 어떤 언어현상들이 서로 충돌할 지에 대해서 미리부터 파악하기는 쉽지가 않아서 다시 원상복귀(backtracking)하는 경우도 있다.

따라서 성능 향상을 위해 기능을 개선할 때는 유사한 언어현상들에 미치는 영향을 신중하게 고려하는 것이 매우 중요하다. 때로는 미시적인 관점이 아니라 거시적인 관점에서 방법론을 포함한 전체 시스템의 구조를 재검토하는 것이 더 효율적일 수도 있다. 시스템을 설계할 때 가급적 언어현상끼리 충돌문제를 예방함으로써 통제불능 상태에 빠지지 않으며, 시스템 재구성으로 인한 추가 부담을 최소화할 수 있는 방법론으로 2-step 패러다임과 이를 실현하기 위한 방법론을 제안한다.

5.1 2-step 패러다임

한국어 정보처리에서 시제품을 개발한 후에 일정한 수준의 신뢰도를 보장할 수 있는 실용적인 시스템으로 발전시킬 때는 어느 정도의 노력으로 어느 정도의 성능 개선 효과가 나타날지를 예측하기가 쉽지 않다. 이러한 특성을 극복하기 위한 2-step 패러다임은 실용적인 시스템을 개발할 때 주어진 문제를 하나의 틀 속에서 접근하는 대신에 두 개 혹은 그 이상의 독립된 단계들로 구분하여 접근하는 방법이다. 이 패러다임은 시제품을 개발한 후에 그 경험을 바탕으로 실용적인 시스템을 개발하는 것과 같이 보편적으로 알려져 있으나 한국어 정보처리 연구에서는 잘 활용되지 않고 있다. 여기서는 이를 한국어 정보처리 시스템에 적용하여

- 핵심 기능과 확장 기능(또는 부가 기능)
- 적용 범위가 넓은 것과 좁은 것
- 해결 방법이 명확한 것과 그렇지 않은 것
- 정상적인 처리가 가능한 것과 아닌 것

---

5) 형태소 분석기는 약 2~4 man/month, 기계번역은 약 2~4 man/year의 작업으로 시제품 개발이 가능하다고 추정된다.

등으로 1 단계에서 처리해야 할 문제와 1 단계 처리가 끝난 후에 확장-보완하여 2 단계에서 처리할 문제로 명확히 구분한다.

1 단계는 기본단계(basic step)로 어떤 문제에 대한 기본적인 기능(단순하고 처리 방법이 비교적 명확한 기능)을 처리하고, 2 단계는 확장단계(extended step)로서 부가적인 기능 혹은 처리 방법이 명확하지 않아서 핵심기능이 구현된 후에 알고리즘을 고안할 수 있는 확장기능들을 의미한다[6]. 그 예로는 형태소 분석시에 복합어 분해 문제가 있다. 체언 접미사나 보조용언이 결합된 복합어는 형태소 분석 결과에 미치는 영향이 크기 때문에 형태소 분석 과정에서 처리되어야 한다. 그러나 복합명사와 결합형 조사/어미의 분해는 형태소 분석 결과에 미치는 영향이 거의 없고, 오히려 형태소 분석 알고리즘의 복잡도를 증가시키는 역할을 한다. 따라서 이 기능은 조사/어미가 분리된 후에 처리하는 확장 기능으로 분류하는 것이 바람직하다.

다른 예로는, '은/는'이나 '아/어'와 같이 변형이 일어난 어미의 원형복원 기능, 분석결과에서 단어 유형이나 조사/어미의 빈도수에 따라 우선순위를 정하는 기능, 붙여쓴 의존명사의 인식 기능, 준말처리 등을 확장단계로 분리할 수 있다. 2-step 패러다임의 실현 방안으로는 기능별 모듈화에 의한 분할-정복(divide-and-conquer) 기법, 단순화 기법[7], 예외처리(exception handling) 기법 등이 있다.

5.2 기능별 모듈화 기법

기능별 모듈화는 하나의 알고리즘으로 처리할 수 있을 것이라고 예측되는 다수의 유사한 언어현상들에 대한 명확한 알고리즘이 알려져 있지 않다면, 각 언어현상의 개별적인 특성에 따라 기능별로 독립시키는 방법이다. 한국어의 언어현상들은 유사한 경우가 많기 때문에 한꺼번에 문제를 해결하려고 시도하게 된다. 그런데 유사한 언어현상들이 각각 고유한 특성을 가지고 있기 때문에 구체적으로 시스템의 성능을 개선할 때 문제가 된다. 따라서 모듈화가 가능한 언어현상들은 가급적 별개의 모듈로 독립시키는 것이 바람직하다.

한국어 정보처리 시스템은 분석과 생성, 그리고 사전 탐색이나 한글코드 문제 등 부수적인 기능들로 구분된다. 분석 문제는 다시 형태소 분석, 구문분석, 중의성 해결 등으로 구분되고 각 모듈은 다시 세부 기능으로 구성된다. 그런데 형태소 분석이나 구문분석 등 핵심적인 문제는 하나의 독립된 문제로 간주되는 것이 일반적이다. 형태소 분석의 예를 들면, 입력 단어로부터 형태소들을 분리하고 형태소간의 결합 제약을 검사하는 포괄적인 접근 방법이 가능하다. 그러나 포괄적인 형태소 분석 방법을 취했을 때 형태소 유형에 따른 개별적인 특성들을 반영하기가 어렵다[8]. 개별적인 언어현상들을 포괄적으로 처리함으로써 처리 범위가 커져서 통제불능 상태에 빠지기가 쉬우며 성능 향상이 불가능한 포화상태에 빠질 가능성이 높아진다.

---

6) 강승식(1996)은 단어 유형과 품사 유형을 단순형과 확장형으로 구분하고, 형태소 분석 알고리즘을 어형인식과 어형확장 단계로 구분하는 두 단계 확장 모델을 제안하였다.

7) 이와 유사한 기법으로 Kai Hwang 교수는 KISS(Keep It Simple and Scalable)를 제안하였다.

8) 특정 형태소 유형을 처리하기 위해 어떤 기능을 수정할 때 이 기능이 다른 유형의 형태소를 분리하는데 미치는 영향을 고려하기가 쉽지 않다.

형태소 분리 기능의 경우에 조사와 어미 분리, 단위 조사/어미 인식, 선어말 어미 처리, 보조용언 분리, 불규칙 용언의 원형 복원, 복합명사 분해, 숫자와 영문자 처리, 준말 처리 등 세부 기능에 대한 처리 방법이 서로 독립적이므로 세부 기능별로 분할하여 모듈화하는 것이 지속적인 성능 향상을 위해 많은 도움이 된다(강승식, 1993). 이처럼 처리 범위가 넓은 모듈은 통제불능 상태에 빠질 가능성이 많으므로 독립적인 모듈로 세분화할 수 있는 것은 모두 세부 기능으로 독립시키는 분할-정복 기법을 활용하는 것이 효율적이다. 이 때 한 모듈이 다른 모듈에 영향을 미치는 것들은 그 내용은 명확히 명시해 둠으로써 모듈별로 기능을 개선할 때 관련 모듈에서 반영될 수 있도록 한다.

기능별 모듈화의 장점은 시제품이나 중간 제품을 폐기하고 재설계해야 할 필요성이 발생할 때 특히 유용하므로 한국어 정보처리의 특성에 적합한 방법이다. 독립적인 모듈들의 기능을 명확히 정의할수록 모듈별 재사용이 쉬워서 재설계로 인한 부담이 적어지기 때문이다. 또한 모듈별로 보다 효과적인 처리 방법론이 고안되었을 때 이를 전체 시스템에 반영하기가 용이하다.

5.3 단순화 기법

일반적으로 한국어 정보처리 시스템을 개발할 때는 대표적인 언어현상들을 처리하는데서부터 출발한다. 우선, 대표적인 문장들을 수집하고 이를 중심으로 전체적인 처리 범위를 설정하여 시스템을 설계하게 된다. 이 때 일반화 오류(generalization error)가 발생하기 쉬우며, 개발 과정에서 처리 범위 혹은 방법론을 수정해야 하는 일이 자주 발생한다. 방법론의 소폭 수정은 불가피하지만 한 방향으로 계속해서 수정되면 전반적인 방향이 의도했던 것과 점점 차이가 커질 수도 있다. 방법론을 수정해야 할 시점에 이르면 개발 과정에서 발견된 문제점들을 고려해서 처음부터 다시 시작해야 하지만, 이 정도 수준의 결과를 얻기까지의 세부적인 작업들을 추적하는 문제와 함께 유사한 노력을 반복해야 하는 문제로 인해 쉽지가 않다[9]. 더군다나 알고리즘 혹은 시스템의 구조가 복잡할수록 재시작 문제는 더욱 어려워진다. 그 이유는 세분화되거나 구조가 복잡할수록 적용 범위가 좁아지고 미묘한 차이에 의해 의도했던 바와 다른 결과가 나타나기 때문이다.

언어현상은 문법규칙으로 기술되는 보편적인 현상보다 문법규칙으로 기술되기 어려운 개별적인 현상들이 훨씬 많다. 따라서 개별적인 언어현상들을 처리할 때마다 시스템의 구조는 점점 복잡해지게 된다. 더군다나 언어 유형이 세분화된 상태에서 출발했을 때는 복잡도가 더욱 심화될 수밖에 없다. 이처럼 복잡도가 증가하는 문제를 해결하려면 확장 가능성이 충분히 고려된 단순한 모델로부터 시작하여 가급적 복잡도가 증가하지 않도록 하면서 이를 확장해 나가는 방식을 취하는 것이 바람직하다.

알고리즘의 복잡도를 줄이는 방법 중의 하나는 전체 시스템의 구조와 독립적인 부가적인 문제들을 선별하여 다음 단계의 작업으로 남겨두는 것이다. 예를 들어, 형태소 분석시에 복합명사 분해라든지 조사/어미의 결합형으로부터 단위조사와 단위어미를 분리하는 기능은 체언부가 인식된 후에 처리해도 되므로 꼭 형태소 분석 과정에서 처리할 필요는 없다.

---

9) 그렇더라도 재시작하는 것이 장기적으로 볼 때 바람직하며, 최소한 전체 시스템의 구조를 재정립해야 한다.

5.4 통계적 기법과 예외처리

분석적 기법은 언어현상을 처리하는 규칙의 발견 및 확장에 의해 문제를 해결하려 하는데, 모든 규칙을 발견하기가 어렵고 규칙간에 충돌 현상이 발생하여 모호성이 증가하기도 한다. 이러한 문제점을 극복하기 위해 통계적 기법을 사용하기도 하지만 통계적 기법은 개별적인 언어현상을 처리하기가 어려운 단점이 있다. 따라서 분석적 기법과 통계적 기법, 그리고 예외처리 기법을 적절히 조합함으로써 성능 개선이 용이한 시스템을 구축할 수 있다.

시스템을 설계하거나 성능 개선 과정에서 '이런 경우는 발생하지 않겠지'라고 단언해서는 안된다. 예를 들어, 50단어가 넘는 문장은 없을 것이라든지, 한 단어의 길이가 20음절을 넘지는 않을 것이라는 가정, 혹은 조사와 결합할 수 있는 품사는 체언밖에 없을 것이라는 제약 등은 구조적인 오류를 내포할 가능성이 많다. 이러한 오류가 발생하지 않게 하려면 가능성이 거의 없다고 판단되는 사소한 기능이라 할지라도 오류처리(error handling) 기능으로 남겨두는 것이 현명하다.

자연언어의 개별적 현상들은 일부 특수한 경우에만 적용되는 경우가 많기 때문에 정상적인 방법이 아니라 쉽게 편법으로 처리하려는 유혹이 있다. 그러나 계속되는 성능 개선 과정에서 오류가 발견되어 기존의 작업이 헛수고가 되기도 한다. 따라서 개별적인 언어현상을 처리할 때 그 유형만을 위한 방법론은 부적합하며 유사한 유형들이 발생할 가능성을 고려하여 확장 가능성을 남겨 두는 것이 바람직하다.

6. 사례 연구

6.1 한국어 형태소 분석

형태소 분석 방법론 중에서 Tabular 파싱법과 음절단위 분석법을 2-step 패러다임의 관점에서 비교해 보면 다음과 같다. Tabular 파싱법은 형태소를 분리하고 각 형태소에 품사정보를 부착한다는 형태소 분석의 의미에 충실한 방법론이다(김성용, 1987). 이 방법에서는 모든 형태소들의 분리 문제를 동일한 알고리즘으로 처리하고 있으며, 형태소의 유형에 따라 개별적인 현상들은 결합제약 정보로서 해결하려 한다. 그런데 복합명사와 복합조사, 복합어미 등 개별적인 언어현상으로 인하여 알고리즘의 복잡도가 증가하고 시스템의 성능이 개선되기 어려워지게 되었다.

이 방법의 근본적인 문제점은 조사/어미의 분리, 선어말어미 인식, 불규칙 용언의 원형복원, 접두사와 접미사의 분리, 복합명사 분리 등 개별적인 언어현상들을 종합적으로 처리하려고 시도한데서 발생하고 있다. 따라서 형태소 분리라는 공통적인 현상과 각 형태소마다 개별적인 현상이 복합되어 알고리즘의 복잡도가 증가하고, 새로운 언어현상이 발견될 때마다 전체적인 알고리즘을 전반적으로 검토해야 하는 문제가 있다.

음절단위 분석법에서는 단어 유형과 품사 유형을 가급적 단순화함으로써 알고리즘의 복잡도를 줄이는 단순화 기법, 복합명사와 복합조사/어미는 후처리 기능으로 모듈화하는 기법, 축약 현상에 대해서는 예외처리 기법 등 2-step 패러다임을 적절히 활용하고 있다. 또한, 선어말어미의 인식이나 불규칙 용언의 원형복원 기능 등이 독립적으로 모듈화되어 보조용언을 붙여 쓴 어절에서 본용언이 불규칙인 경우와 본용언에 선어말어미가 결합된

현상에도 동일하게 적용되는 장점이 있다.

이 방법론에서는 단순화 기법과 모듈화 기법을 적절하게 활용했기 때문에 성능이 좋은 복합명사 분해 알고리즘이 고안되었을 때 기존 알고리즘을 쉽게 교체할 수 있다. 또한, 개별적인 언어현상들에 대해 각 언어현상에 적합한 알고리즘이 사용됨으로써 정확한 형태소 분석이 가능하고 형태소 분석 기능을 바로 맞춤법 검사 용도로 활용할 수 있는 장점도 있다. 형태소 분석기의 성능에 가장 큰 영향을 미치고 있는 미등록어를 추정하는 부분도 모듈화 되기 때문에 미등록어 추정 기능의 개선이 용이하다.

형태소 분석기에서 기능별 모듈화는 일부 모듈이 개선되었을 때 전체 시스템의 성능 개선으로 직접 반영되는 장점이 있을 뿐만 아니라 특정 응용 시스템의 요구사항에 적합하게 수정하기가 편리하다. 2-step 패러다임에 충실하게 구현된 형태소 분석기인 HAM (Hangul Analysis Module)의 경우에 단순화 기법과 모듈화 기법, 예외처리 기법을 잘 활용함으로써 다양한 형태의 분석결과를 생성할 뿐만 아니라 처리속도와 정확도가 우수하며, 자동색인 및 맞춤법 검사기로 실용화되기도 하는 등 형태소 분석과 관련된 다양한 기능을 제공하고 있다.

6.2 기계번역 시스템

기계번역 시스템은 원시언어의 분석, 원시언어 분석결과를 목적언어로 변환, 그리고 목적언어를 생성하는 과정으로 구성된다(심광섭 · 김영택, 1994). 초기의 기계번역에 관한 연구는 원시언어의 분석이나 구조변환, 목적언어의 생성 등에 대한 기반 기술이 없는 상태에서 각 부분을 한두 가지 기능으로 구분하여 시스템을 설계하였으며, 모든 언어현상들을 처리하는 규칙에 의해 시제품이 개발되었다.

기계번역을 비롯한 자연언어 처리 시스템은 일반적으로 시제품을 LISP이나 PROLOG 언어로 구현하여 가능성을 확인한 후에 실용적인 시스템은 C언어로 개발하는 2-step 패러다임을 적용하고 있다. 그러나 시제품을 성공적으로 개발했더라도 실용적인 시스템이 개발되기까지는 많은 기간이 소요되고 있으며, 시제품 수준에서 더 이상 발전하지 못하는 경우가 많다. 그 이유는 시제품을 개발할 때 취했던 방법론인 유사한 언어현상들을 한꺼번에 처리하는 기법을 그대로 적용하고 있기 때문이다.

다양한 문장 유형에서 발생하는 개별적인 현상들을 문제 유형에 따라 모듈화하거나 예외처리할 수 있도록 시스템을 재구성하지 않고 전체적인 알고리즘을 개선하는 방법을 취했을 때 성능 개선 효과가 미미하게 된다. 특히, 짧은 문장 번역에서 긴 문장으로 확장할 때 긴 문장들의 유형에 따라 언어현상들을 하나씩 추가하면, 통제불능 상태에 빠지기가 쉽다. 이러한 이유로 인하여 기계번역 시제품은 쉽게 개발되지만 실용적인 제품이 개발되지 못한 경우가 많다.

이러한 문제점을 극복하려면 시스템을 설계할 때 단순화 기법을 취하면서 구체적인 새로운 문제가 나타나면 그 기능을 모듈화하거나 혹은 예외처리 기법을 도입하여 각 모듈들이 유기적으로 결합하도록 해야 한다. 예제기반(example-based) 기계번역 기법은 적용범위가 넓은 보편적인 언어현상을 규칙화하고 적용범위가 좁은 개별적인 언어현상들은 예제기반 기법으로 구분하여 2-step 패러다임을 적용한 좋은 예라 할 수 있다(Nagao, 1984).

7. 결 론

한국어 정보처리 분야의 가장 심각한 문제점은 시제품으로부터 실용적인 시스템으로 발전할 때 국부적인 성능 개선 효과가 전체 시스템의 성능으로 거의 반영되지 않고 통제불능 상태에 빠지는 것이다. 한국어 정보처리 시스템의 현황 및 속성을 검토함으로써 통제불능 상태에 빠지는 원인을 고찰하였으며, 이를 회피하는 방법론으로 2-step 패러다임을 제안하였다. 또한, 2-step 패러다임을 실현하기 위한 방법론으로는 분할-정복 방식에 의한 기능별 모듈화 기법, 단순화 기법, 예외처리 기법 등을 적용하는 방안을 제안하였다.

일반적으로 한국어 정보처리 문제는 완벽한 알고리즘이 존재하지 않기 때문에 처음 선택한 알고리즘을 지속적으로 개선하여 성능이 향상될 수 있는지가 매우 중요한 요소이다. 한국어 정보처리 시스템을 설계하거나 그 기능을 확장할 때 2-step 패러다임과 이를 실현하는 방법론에 충실함으로써 시제품에서 실용적인 시스템으로 발전할 때 발생하는 통제불능 상태를 회피할 수 있다.

참 고 문 헌

강승식. 1993. *음절정보와 복수어 단위정보를 이용한 한국어 형태소 분석*. 서울대학교 박사학위 논문.

강승식. 1996. "음절 특성을 이용한 범용 한국어 형태소 분석기 및 맞춤법 검사기." *정보과학회논문지(B)*, 530-539.

권혁철. 1994. "한글 및 한국어 정보처리의 현황." *정보과학회지*, 12권, 8호, 3-16.

김성용, 최기선, 김길창. 1987. "Tabular Parsing 방법과 접속 정보를 이용한 한국어 형태소 분석기." *정보과학회 인공지능연구회 춘계 인공지능 학술발표회 논문집*, 133-147.

김영택. 1994. *자연언어처리*. 교학사.

김태석. 1997. "일한 기계번역 시스템의 연구 및 개발." *정보과학회지*, 15권, 10호, 9-15.

노용균 · 박동인. 1994. "Corpus Linguistics의 현황과 한국어 Corpus 구축 및 활용의 제문제." *정보과학회지*, 12권, 8호, 67-71.

심광섭 · 김영택. 1994. "기계번역 시스템." *정보과학회지*, 12권, 8호, pp.17-23.

Barton, G, Berwick, R & E. Ristad. 1987. *Computational Complexity and Natural Language*. MIT Press.

Hutchins, W. J. 1986. *Machine Translation: Past, Present, Future*, Ellis Horwood Limited, 164-167.

Lehnert, W & Ringle, M. 1982. *Strategies for Natural Language Processing*. Lawrence Erlbaum Associates.

Nagao, M. 1984. "A Framework of a Mechanical Translation between Japanese and English by Analogy Principle." *Artificial and Human Intelligence*. North-Holland Publishing Company.

Nirenburg, S. 1987. *Machine Translation: Theoretical and Methodological Issues*. Cambridge University Press.

접수일자 : '99. 2. 25.
게재결정 : '99. 3. 21.

▲ 서울특별시 성북구 삼선동2가 389번지
한성대학교 이공대학 정보전산학부(우: 136-792)
Tel : (02) 760-4136, Fax: (02) 760-4217
e-mail: sskang@hansung.ac.kr