한국어 대화체 음성언어 번역시스템에서의 개념기반 번역시스템

최 운 천† · 한 남 용† · 김 재 훈†

요 약

대화체 음성언어번역시스템의 일부인 개념기반 번역시스템은 음성인식의 결과를 이용하여 다른 언어로 번역해 주는 시스템이다. 본 논문은 여행계획 영역에 대해 한국어를 해석하여 영어, 일본어, 한국어로 번역해 주는 시스템에 대해 기술한 것이다. 개념기반 번역은 비정형 문장이 많은 대화체 문장을 처리하기 위해 형태소 분석 등의 구문정보를 이용하지 않고, 의미단위의 번역을 시도한 것으로 화자의 의도를 정확히 번역해 주는 것을 목표로 한다. 개념기반 번역은 280여개의 개념과 개념간의 계층구조에 의해, 인식결과를 개념구조로 변환한 후 다른 언어로 생성해 준다. 효율적인 한국어 처리를 위해 기준단어를 이용한 토큰분리기와 문법자동 수정기를 개발하였다. 그리고 자연스러운 생성문을 위해 각 언어에 대한 후처리기를 개발하였다.

Concept-based Translation System in the Korean Spoken Language Translation System

Un-Cheon Choi† · Nam-Yong Han† · Jae-Hoon Kim†

ABSTRACT

The concept-based translation system, which is a part of the Korean spoken language translation system, translates spoken utterances from Korean speech recognizer into one of English, Jananese and Korean in a travel planning task. Our system regulates semantic rather than the syntactic category in order to process the spontaneous speech which tends to be regarded as the one ungrammatical and subject to recognition errors. Utterances are parsed into concept structures, and the generation module produces the sentence of the specified target language. We have developed a token-separator using base-words and an automatic grammar corrector for Korean processing. We have also developed postprocessors for each target language in order to improve the readablility of the generation results.

1. 서 론

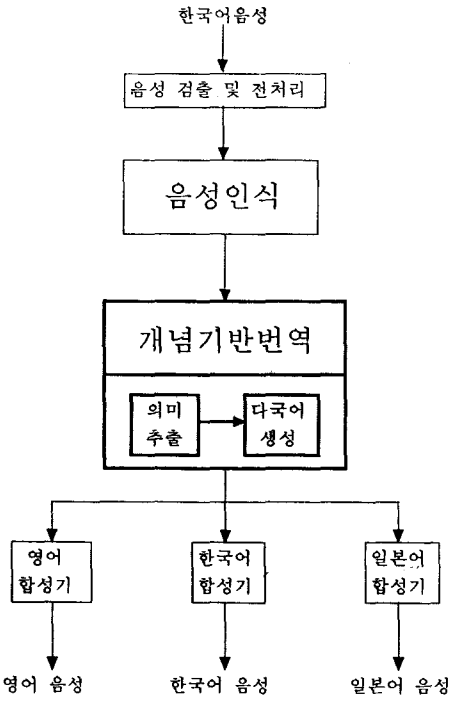
대화체 음성언어번역 연구는 언어장벽을 해소하고 보다 편리하게 외국인과 자연스러운 의사교환을 할 수 있는 시스템 개발을 목표로 세계 여러 나라에서 진행 중이다[3, 4, 5, 6, 8]. 대화체(spoken language)는 낭독체(reading speech)와는 달리 우리가 사용하는 자연스러운 대화를 표현한 글이다. 그래서 대화체 문장은 문법적으로 부정확한 표현을 사용하기도 하고, 말을 하다 도중에 끝내버리기도 하고, 말더듬이나 간투사 등이 포함되기도 한다. 대화체 처리의 어려움으로

† 정 회 원:ETRI, 음성언어연구실
논문접수:1996년 10월 24일, 심사완료:1997년 7월 30일

인해 대부분의 시스템이 한정된 도메인에서 개발되고 있다.

대화체 음성언어번역 시스템은 크게 음성인식과 기계번역, 음성합성 세 단계로 되어 있다. 음성인식은 음성을 인식하여 문자로 변환해 주고, 기계번역은 인식의 결과를 받아 다른 언어로 번역해 준다. 그리고 음성합성은 번역된 결과를 다시 음성으로 들려 준다. (그림 1)은 음성언어번역 시스템의 개요를 보인 것이다. 한국어 음성은 음성인식시스템을 거쳐 개념기반 번역시스템에 텍스트 형태로 입력된다. 개념기반 번역시스템은 개념구조를 이용하여 의미를 추출한 후 여러 목적언어로 생성된다. 생성된 결과는 각 언어의 합성기로 합성되어 음성으로 출력된다. 본 논문은 세 단계 중 둘째 단계인 개념기반 번역시스템에 관한 것이다.

대화체 문장의 번역방법으로는 크게 예문을 중심으로 번역이 이루어지는 예문기반 번역방법(example-based machine translation[9], transfer-driven machine translation[10], token-based transfer-driven machine translation[3])과 언어 개별적인 개념(concept)을 단위로 번역하는 개념기반 번역방법[1, 2, 6]이 있다. 본 논문에서는 후자를 이용하여 한국어 입력에 대해서 영어, 일본어 그리고 한국어를 생성하는 시스템에 관하여 기술한다.



(그림 1) 대화체 음성언어 번역시스템의 구성도
(Fig. 1) System configuration of the spoken language translation system

일반적인 기계번역은 원시언어를 해석하는 해석단계와 목적언어로 변환하는 변환단계와 목적언어를 생성하는 생성단계로 구성되며 언어학적 지식 즉, 각 단어의 품사와 문장에서의 역할이 중시된다. 이를 위해 형태소해석, 구문해석, 의미해석, 변환, 구문/의미생성, 형태소생성의 여섯 단계를 거쳐 한 문장을 번역하게 된다. 그러나 개념기반 번역은 언어학적 지식보다는 발화자의 의도를 가장 잘 찾을 수 있는 표현 위주로 번역을 한다. 표현 위주의 번역은 앞에서 언급한 비정형문이 대부분인 대화체를 처리하기에 적당한 방법 중 하나이다. 그리고 각 단계별로 번역을 하지 않고, 의미에 바탕을 둔 개념구조를 이용하여 원시언어를 해석하여 개념구조를 만드는 해석과, 이 개념구조를 이용하여 목적언어로 생성하는 생성, 두 단계로 이루어진다.

개념기반 번역은 시스템에 대해 잘 모르는 초보자라도 쉽게 문법을 작성하고 검증할 수 있다. 그리고 우리가 실생활에서 사용하는 표현을 그대로 문법에 기술할 수 있어 확장이 매우 쉽다. 개념위주로 해석을 하고, 간투사들은 무시할 수 있는 기능이 있어 대화체 처리에 유리하다. 개념기반 번역시스템은 미국 CMU에서 Phoenix Parser[4, 6]라는 이름으로 처음 개발되었고, 만남, 약속(scheduling) 영역을 대상으로 하였다. 본 논문에서 기술하는 시스템은 Phoenix Parser를 한국어 처리에 맞게 수정, 보완한 것으로 새로운 영역인 여행계획영역(traveling arrangement domain)을 대상으로 하고 있다.

개념기반 시스템에서 개념의 추출은 단어 단위로 이루어진다. 영어와 같은 굴절어는 한 단어가 한 의미를 가지는 경우가 대부분이라 단어 단위의 처리가 유리하지만, 한국어와 같은 교착어는 조사나 어미 등의 기능어가 명사나 용언 등의 실질형태소와 함께 한 단어에 나타나기 때문에 하나의 단어가 하나 이상의 의미를 갖는 것이 일반적이다. 그래서 한국어 처리를 위해서는 기능어를 분리할 필요가 있다. 또한, 기존의

시스템에서는 영어의 대소문자를 구별하지 않아 영어 생성문의 이해가 어려웠으며, 최상위개념 간의 구분이 없어, 생성문만 가지고는 모호한 경우가 많았다.

본 논문은 기존의 Phoenix Parser의 기본 특징을 살리면서, 한국어 처리에 맞도록 수정, 보완한 시스템과 여행계획영역에 대해 새로 정의한 한국어 해석문법과 다국어 생성문법에 관해 기술한 것이다. 여행계획영역을 처리하기 위해 가장 중요한 것이 개념의 재정립이다. 이를 위해 총 282개의 개념을 정의하고, 개념의 계층구조를 정립하였다. 정의된 개념을 바탕으로 한국어 해석문법과 영어, 일본어, 한국어 생성문법을 만들었다. 또한, 토큰분리기를 개발하여 한국어의 특징인 조사와 어미의 활용을 처리할 수 있도록 하고, 숫자나 영어 대소문자의 구별, 최상위개념 간의 분리 등을 통해 생성문의 모호함을 해소하고 이해를 높였다. 본 시스템은 다국어 번역시스템의 초기 모델로 개발 중인데, 한국어를 해석하여 영어, 일본어, 한국어로 생성하여 준다.

2. 개념기반 번역시스템의 개요

개념기반 번역시스템에서 개념파서는 입력문을 분석하여 개념구조로 변환하고, 생성시스템은 개념구조를 목적언어로 생성하여 준다. 개념기반 번역시스템이 개념을 중심으로 발화를 해석하기 때문에 시스템이 사용하는 파서를 개념파서라고 부른다. 개념파서는 의미를 바탕으로 발화(utterance)를 해석하고 번역과 생성에 이용된다[1, 2, 4]. 개념파서는 입력된 발화에서 구체적으로 말하고자 하는 의미만 추출하여 번역하고, 상대국어로 생성하는 방법이다[1, 2]. 의미추출과 직접적인 관련이 없는 간투사 등의 단어들은 무시된다.

2.1 개 념

개념이란 영역에 따라 정의된 의미표현의 단위로 개개의 단어와는 다르고 특정언어에 의존적이지 않으며 발화자의 명확한 의도를 표현하기 위해 사용된다[7]. 본 논문에서 개념은 일반적으로 자연언어 처리 영역에서 말하는 개념과 다르게 개념파서에서 사용하는 기본 단위로서 발화자가 사용한 서로 다른 의미를 표현하는 토큰의 집합으로 설계하였다. 개념은 의미의 틀을 만들어 두고, 그것에 해당되는 단어들을 넣어 두기 때문에 의미슬롯이라고도 부른다. 개념들 중에서 제안이나 동의같은 화행(speech act)을 본 논문에서는 최상위개념이라 하며 생성의 기본 단위로 사용된다. 하위개념은 날짜나 요일같은 발화의 상세한 의미를 나타낸다[7].

2.2 해 석

해석문법은 영역 내에서의 개념을 표현하는 패턴들을 상세히 나타낸다. 패턴은 단어나 다른 의미슬롯으로 구성되어 있다. 패턴의 요소들(단어나 의미슬롯)은 생략될 수도 있고 반복될 수도 있다. 각각의 개념은 계층구조 내에서의 위치에 관계없이 개개의 파일로 표현되며 이것을 문법파일이라 한다. 이 문법들은 효율적인 검색을 위해 RTN(recursive transition network)으로 컴파일된다[4, 7].

개념파서는 RTN에 의해 정의된 패턴들 중 입력과 비교하여 가능할 때까지 매치를 시도한다. 개념파서는 최상위개념을 중심으로 해석을 한다. 최상위개념과 최상위개념 사이의 매치되지 않는 단어(미등록어(out-of-lexicon words) 포함)는 무시한다. 그러나 최상위개념 내에서 해석문법에 있지만 패턴과 매치되지 않은 단어가 있는 경우 매치는 실패하게 된다. 그러나 단순히 그 최상위개념에 대한 실패이지 발화전체가 실패하는 것은 아니다. 의미추출과 직접 관련이 없는 간투사가 최상위개념과 최상위개념 사이에 나타나면 무시한다. 만약 최상위개념 내에 간투사가 나타날 경우, 해석문법에 생략가능을 의미하는 '*'가 간투사와 함께 나타날 때, 그 간투사는 무시된다.

파서는 모호성이 발생한 결과에 대해서는 다음과 같이 하나의 결과를 결정한다. 첫째, 파서는 가능한 많은 단어가 매치된 결과를 찾는다. 둘째, 의미슬롯의 수가 적은 것을 찾는다. 셋째, 파스트리의 최상위레벨에서 가장 적은 수의 개념을 가진 것을 고른다. 넷째, 개념들이 보다 적은 계층에 걸쳐 있는 경우를 고른다.

2.3 생 성

생성문법은 각 개념에 대한 목적언어 표현의 집합으로 되어 있다. 입력문자들이 개념레벨로 축소되었기 때문에 목적언어의 생성은 쉽게 할 수 있다. 생성은 단순히 파싱결과를 왼쪽에서 오른쪽으로 차례로

처리함으로써 가능하다. 처리 대상이 되는 개념에 대해 하위개념이 있는 경우 먼저 하위개념을 생성한 후, 상위개념으로 그 결과를 전달함으로써 이루어진다. 생성의 단위는 최상위개념이 되며, 모든 최상위개념에 대한 처리가 끝나면 생성도 완료된다. 날짜나 요일과 같이 일대일 번역이 필요한 경우는 검색테이블을 포함한다. 생성의 결과는 의미만을 전달하기 때문에 매우 간결하고 명확하다[4].

3. 개념의 분류 및 계층구조

3.1 개념의 분류 및 문법작성

개념분류를 위해 여행계획 영역에서 수집된 300대화를 사용하였다. 300대화의 총 발화수는 1500여개이다. 개념의 분류는 주어진 대화를 읽어보면서 먼저 쉽게 개념을 정할 수 있는 큰 개념부터 시작한다. 큰 개념은 흔히 화행(speech act)을 의미하지만 문법기술자가 자유롭게 정의할 수 있다. 큰 개념이 나름대로 정해지면 그 큰 개념에 따라 발화를 나눈다. 큰 개념을 좀더 작은 개념으로 분리하여 더 이상 나눌 수 없을 때까지 분리한다. 이때 고려해야 할 점은 이미 만들어진 개념 중에 비슷한 의미를 가진 것이 있는지 여부를 살펴 보는 것이다. 비슷한 의미를 가진 것이 있는 경우는 이미 있는 개념을 이용할지, 아니면 그것과 명확히 구분되는 새로운 개념을 만들지를 결정해야 한다.

개념은 크게 세 가지 종류로 나눌 수 있다. 첫째는 일대일 대응을 필요로 하는 것들로 비슷한 의미를 모아둔 단어들을 하나의 개념으로 만드는 것이다. 예를 들면 "요일이름"이다. "요일이름"은 요일이름이란 한 개념 내에 모든 요일의 이름을 포함시킬 수 있다. 번역할 경우 각 요일이름에 대응하는 상대 언어의 표현이 있다. 즉, 일대일 대응이 되는 경우이다. 이 경우는 상대언어에 의존적이며 생성시 매치되는 번역표현을 출력한다. 아래의 예는 개념 "resort"에 대한 해석문법과 영어 생성문법이다. "resort"라는 개념은 도시이름을 포함한 모든 관광지에 대한 지명을 포함하는 것이다. 이런 종류의 개념은 아래의 생성문법에서 나타나 있듯이 해석문법에 나타난 모든 표현에 대해 목적언어의 표현이 일대일 매칭이 되도록 작성되어야 한다. 개념은 '['와 ']'사이에 한 단어로 나타낸다.

해석문법

[resort]

(강릉)
(계룡산)
(과학 문화센터)
(괌)
(그랜드 캐년 공원)
...

생성문법

[resort]

(Kangreung = 강릉)
(Kyeoryng Mountain = 계룡산)
(Science Cultural Center = 과학 문화센터)
(Kuam = 괌)
(the park of Grand Canyon = 그랜드 캐년 공원)
...

둘째는 단독 표현을 사용하는 것으로 있는 그대로의 표현을 사용하는 경우이다. 예를 들면 "call_me(나에게 전화 요망)"라는 개념을 만든 경우 이 개념에는 "필요할 때 나에게 전화를 주십시오"라는 의미의 어떠한 표현도 모두 나타날 수 있다. 유사한 표현이 발견될 때마다 단순히 해석문법에 그 표현 자체를 기술함으로써 해석문법을 작성할 수 있다. 생성문법에는 "나에게 전화 요망"이라는 개념을 나타내는 상대 언어의 가장 자연스러운 표현만 기술하면 된다. 생성문법에 기술된 표현이 생성의 결과가 된다. 이 방법과 첫번째 방법은 모두 하위개념이 없다는 공통점을 가지는데 가장 작은 단위의 개념(최하위 개념)이다. 아래의 해석문법에 있는 '*'는 그 단어를 생략할 수 있음을 의미한다.

해석문법

[call_me]

*(*꼭 *다시 연락 주 시 길 바 랍니다)*
(달 리 문의 하고 싶 은 사 항이 있으면 지금 말씀 해 주 세 요)
(더 더 문의 하실 게 있으면 다시 연락 주 십 시 요)
(더 문의 하실 일 이 있으면 언제 라도 전화 해 주 십 쇼 감사 합니다)

(뭐 그밖에 궁금하신 점이 있으면 언제 라도 전화 해 주 십 시 오)

...

생성문법

[call_me]

Please contact me in your convenient time if you have further questions.

셋째는 복합된 구조를 사용하는 것으로 하위개념을 가질 수 있는 구조화된 개념들이다. 이 구조는 가장 복잡하면서도 다양한 표현을 수용할 수 있는 개념 표현방법이다. 문법작성의 기본은 대화체에 나타나는 표현을 그대로 해석문법에 기술하는 것이다. 이때 하위개념을 사용하여 문법을 구조화할 수 있다. 하위개념의 사용은 '['와 ']'사이에 하위개념의 이름을 적음으로써 가능하다. 하위개념들을 잘 정의함으로써 각 개념이 명확히 구분될 수 있다. 이 구조는 여러 레벨의 계층구조를 가진 하위개념도 포함이 가능하다. 해석문법 파일의 각 문장에 나타나는 개념들의 순서는 생성문법에서 다르게 나타나기도 한다. 이것은 언어에 따라 개념들의 순서가 달라질 수 있는 특징을 반영한 것이다. 아래의 해석문법에 나타난 "+[city]"에서 '+'는 한 번 이상 나올 수 있음을 의미는 것으로 도시 이름이 반복적으로 나올 수 있음을 의미한다. 그리고 "include JOSA.nt"라는 표현은 파일 "JOSA.nt"를 포함시키라는 의미이다. "nt"라는 확장자가 붙은 파일은 비슷한 의미를 가진 비단말(non-terminal)들을 모아 둔 것으로 파일 "JOSA.nt"에는 조사와 관련된 비단말들이 들어 있다. 생성문법을 이용한 생성은 생성문법에서 매치되는 하위개념이 있는 경우는 그 하위개념의 번역결과를 사용한다. 만약에 하위개념이 없는 경우는 "〈empty-tok〉" 이후에 있는 표현을 사용하고, 하위개념은 있지만 생성문법에 있는 하위개념과 매치되지 않을 경우는 마지막 문장의 '[ ]' 대신에 하위개념의 번역결과를 사용한다.

해석문법

[id_like]

(다양한 곳 을 많이 여행 하고 싶 은데)

*(*제가 [kind_of_room] *JOSA 쓰고 싶 은데)*

([period] *[duration] 여행 하려고 하는데)*

([period] *[duration] [no_of_people] +[city] *JOSA 여행 하려고 하는데)*

([postpone] 하고 싶 은데)

*([postpone] *JOSA 좀 했으면 합니다)*

([period] *[duration] [language_training] 하고 싶 은데)*

...

#include JOSA. nt

생성문법

[id_like]

i want to take [language_training] during [period] [duration]

i want to take [language_training] during [period]

i want to take [language_training] during [duration]

if I want [kind_of_room] by myself

I'd like to [postpone]

...

〈empty-tok〉 I'd like to know

I'm planning to make a trip [ ]

개념들 중 대표개념의 성격을 가진 최상위개념은 특별한 의미를 갖는다. 최상위개념은 화행(speech act)으로서 영역에서 핵심이 되는 개념들로 구성한다. 이 개념은 해석에서 생성으로 정보를 넘겨주는 단위가 된다. 즉, 생성은 최상위개념별로 생성을 하게 된다. 한 발화는 여러 개의 최상위개념을 가질 수 있다. 그러나 번역실패가 나오지 않으려면 적어도 하나의 최상위개념과 일치하는 표현이 있어야 한다. 이것은 문법을 만들 때 중요한 의미가 있다. 만약 최하위개념들이 곧바로 최상위개념의 하위개념으로 있을 경우 단어 대 단어의 번역도 가능하기 때문이다.

현재 시스템에서 사용하고 있는 최상위개념의 수는 17개이며, 전체 개념의 수는 282개이다. 이들 중, 일대일 대응에 관련된 개념은 44개, 단독개념은 126개, 복합개념은 최상위개념 포함하여 112개이다. 단독개념이 많을수록 개념이 세분화된 것이고, 일대일 대응 개념이 적으면 적을수록 중간언어방식에 가깝게 된다. 참고로 미국 CMU에서는 최상위개념 20개를 포함하여 139개의 개념을 사용하였다. 본 시스템에서 사용

하는 개념 중 약 10%는 CMU의 개념을 이용하였다. 나머지 90%에 해당하는 개념을 새로 정의한 이유는 영역이 달라 사용하는 표현들이 상이하기 때문이다.

3.2 개념의 계층구조

개념은 최상위개념과 하위개념으로 구성된다. 하위개념은 또 다른 하위개념을 가질 수 있다. 최상위개념은 인사관련 개념인 "nicety", 정보제공 개념인 "give_info", 무엇을 가지고 있음을 나타내는 "we_have" 등 모두 17개를 사용하고 있다. 개념의 계층구조는 아래에 보인 최상위개념 "give_info"를 예로 들어 설명한다. 정보 제공을 의미하는 개념 "give_info"는 일곱 개의 하위개념을 가진다. 하위개념은 이름 관련 개념인 "names", 카드 관련 개념 "my_card", 관광 안내인을 의미하는 "tour_guide", 항공편을 의미하는 "flight", 비용 관련 개념 "cost", 숫자관련 개념 "number" 그리고 할인권을 의미하는 "discount_ticket"과 가능함을 의미하는 "possible"이다. 이중 "discount_ticket"과 "possible"은 함께 순서적으로 나와야 함을 의미한다. 그리고 하위개념 "names"는 도시이름 관련 개념 "city"와 인명을 나타내는 "name"을 하위개념으로 가진다. 아래에서 예로 보인 발화는 모두 "give_info"라는 개념으로 표현될 수 있다.

give_info
- *names*
  - **city name*
- *my_card*
- *tour_guide*
- *flight*
- discount_ticket possible
- cost
- number

〈발화의 예〉
저는 서울에 살고 있는 박재현입니다.
제 카드는 다이너스입니다.
영어 관광 가이드가 있는데요.
비행기는 두 편이 있습니다.
기차와 버스의 할인권을 이용할 수 있습니다.
가격은 일박에 오만원입니다.
제 전화번호는 팔이팔에 육이삼육입니다.

4. 한국어 처리를 위한 개념기반 번역시스템

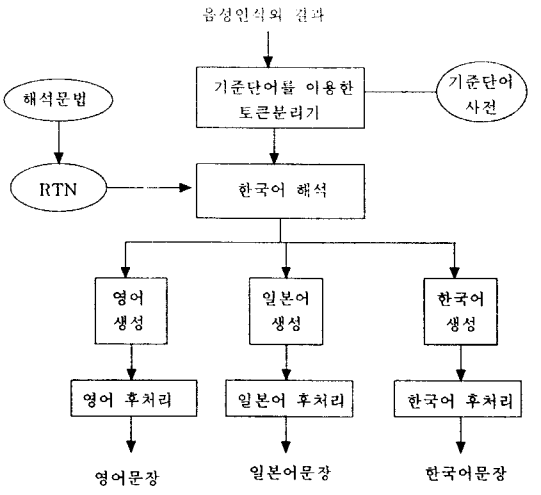
4.1 시스템 구성

개념기반 번역시스템은 (그림 2)와 같이 음성인식 결과를 해석하여 영어, 일본어, 한국어로 각각 생성해 준다. 해석문법은 문법개발자의 편의를 위해 아스키(ASCII)코드로 되어 있지만, 실행 처리 속도를 향상시키기 위해 RTN으로 컴파일되어 사용된다. 한국어 처리를 위해 4.2절에서 정의한 기준단어를 이용한 토큰분리기와 기준단어 사전이 개발되었고, 기준단어 사전의 변동시 모든 관련 문법을 자동으로 수정해 주는 문법자동 수정기가 개발되었다. 그리고 생성문의 이해를 높이기 위한 각 언어에 대한 후처리기가 개발되었다. 현재 시스템은 한국어를 해석하여 영어, 한국어와 일본어로 생성해 준다. 입력으로는 대화를 전사한 문장과 음성인식의 결과를 사용한다.

4.2 기준단어를 이용한 토큰분리기

개념구조를 이용한 언어번역의 첫 단계인 해석(파싱)은 입력된 문장을 보고, 그 문장에서 나타내고자 하는 개념을 찾아서 개념의 트리구조로 만들어 낸다. 이 과정에서 처리 단위는 단어가 된다. 이 원리는 영어와는 잘 맞지만 한국어와는 잘 맞지 않는다. 그것은 영어의 경우는 한 단어가 대부분 한 개념과 대응될 수 있지만 한국어의 경우는 한 단어(여기서의 단어는 어절, 즉, 공백으로 분리되는 것)가 두 가지 이상의 개념과도 대응될 수 있기 때문이다. 그래서 한국어에 맞는 해석을 위해서는 한 개념이 한 단어에 대응하도록 어절을 분리할 필요가 있다. 본 논문에서는 한 개념이 한 단어에 대응되도록 하기 위해 개념의 정의에 중요한 영향을 미치는 단어들을 기준단어라는 이름으로 정의하였다. 그리고 기준단어들을 모아 기준단어사전을 만들었다.

토큰분리기는 기준단어 위주로 입력 문장을 분리하여 준다. 기준단어는 주로 명사 위주지만 용언의 불변하는 부분을 포함한다. 토큰분리기는 개념을 명확하게 하여 준다. 예를 들면 "사일부터 오일까지"라는 문장에서 "사일부터"와 "오일까지"는 실제로 두 단어지만 개념적으로 봐서는 네 개이다. 그러나 기존의



(그림 2) 개념기반 번역 시스템의 개요
(Fig. 2) Overview of the concept-based translation system for korean

방법은 영어나 한국어나 모두 단어위주(공백으로 분리)로 분석을 하기 때문에 네 개가 아닌 두 개의 개념으로 밖에 볼 수가 없다. 그러나 토큰분리기를 이용하면 "사일, 오일, 부터, 까지"가 기준단어이므로 처음의 문장은 "사일 부터 오일 까지"로 분리되어 네 개의 개념이 명확해진다.

토큰분리기는 해석문법과 생성문법의 크기를 현저하게 줄여준다. 예를 들면 기존의 방법을 이용할 경우 요일이름을 의미하는 week_name이라는 개념을 정의하기 위해서는 요일이름이 포함된 모든 단어(어절)가 문법에 포함되어야 한다. 요일이름 뒤에 붙을 수 있는 조사나 어미의 수는 대략 40개가 넘는다. 만약 변화형의 최소치인 40을 기준으로 하더라도 요일이름 일곱 가지에 변화형 40가지를 곱하면 총 280개의 단어가 week_name이라는 문법을 만드는데 필요하다. 그러나 토큰분리기를 이용하면 총 47개의 단어와 한 개의 비단말(어미를 나타내는 것)만으로 표현이 가능하다. 그리고 한 개의 비단말은 다른 개념에서도 사용이 가능하므로 문법의 크기를 현저하게 줄일 수 있다. 이것은 생성문법에도 영향을 미쳐, 대응하는 생성문법의 크기를 해석문법과 같은 비율로 줄일 수 있다.

기준단어를 이용한 토큰분리기는 개념을 명확하게 하여 주고, 해석문법과 생성문법의 크기를 줄일 수 있기 때문에, 문법기술자의 노력을 감소시키고, 보다 적은 크기의 RTN을 만들기 때문에 시스템의 성능을 높일 수 있다.

토큰분리기는 기능이 제한된 형태소해석기라고 볼 수 있다. 명사 위주로 되어 있고, 최장일치 원칙을 따른다. 이 방법의 장점은 기준단어 테이블만 유지하면 되기 때문에 유지가 쉽고, 처리속도가 빨라 번역시스템 전체 성능에 거의 영향을 미치지 않는다. 그리고 모호한 출력을 만들지 않는다. 대화체에서 흔히 나타나는 조사의 오용도 처리가 가능하다. 그러나 기준단어사전에 등록되지 않은 새로운 단어에 대해 잘못된 분리를 할 가능성도 높아 번역의 질을 떨어뜨릴 수 있다. 제한된 도메인이라는 가정이 토큰분리기가 형태소해석기보다 대화체 음성언어번역 시스템의 성능 향상에 기여할 수 있다고 본다.

〈표 1〉 기준단어의 종류
〈Table 1〉 A kinds of base-words

종 류		단어수	비율	예
고유명사		272	33.5	
	인명	86		김경수 김정숙 박상민 손종원 루이스 스즈끼
	지명	146		가고시마 강릉 미국 연세대 제주도 봄베이
	호텔, 여행사명	40		고려 대한항공 존스 한길
날짜 및 숫자		168	20.7	구십 오전 일월 일요일 월말 여섯시 첫째
여행관련 단어		207	25.5	가격 가이드 비행기 여행 전화 특실 사용료
용언		37	4.6	곤란 되겠 모르 불가능 어떻 좋 주셨 괜찮
일반 명사		89	11.0	가능 거리 비용 약속 종류 추천 화폐 질문
지시사 및 관형형		18	2.2	거기 그쪽 어느 언제 얼마 다른 지난
기타		21	2.6	가지 부근 앞 일찍 주변 박사 안녕 잘

토큰분리기에서 사용된 기준단어를 종류별로 살펴보면 〈표 1〉과 같다. 여행계획 영역의 가장 큰 특징은 고유명사가 많다는 것이다. 고유명사는 번역에 중요

한 의미를 가진 단어들이기 때문에 가능한 번역 실패가 있어서는 안 된다. 그래서 고유명사가 기준단어의 1/3을 차지하고 있다. 그러나 이런 다수의 고유명사는 그 다양성으로 인해 미등록어일 가능성도 높아 번역의 질을 떨어뜨리는 요인이 되고 있다. 기준단어 중 용언은 영역에서 자주 나타나는 표현 중 어미의 활용으로 인한 변화와 무관한 부분만 모아둔 것으로 해석문법 작성시에 다양한 어미변화로 인해 문법이 복잡해 지는 것을 막기 위해 사용되었다. 다양한 어미변화는 '*'를 이용하여 생략 가능으로 처리한다. 예를 들면, 〈표 1〉의 기준단어 중 "곤란"이 있으므로 "곤란하다, 곤란한데요, 곤란하구나, ..." 등 "곤란" 뒤에 어떤 어미가 오더라도 해석문법에 "곤란 *하다" 라는 표현만 있으면 모두 처리 가능하다.

토큰분리기와 비슷한 역할을 하는 것으로 문법자동 수정기가 있다. 토큰분리기가 입력 문장을 대상으로 한 것이라면, 문법자동 수정기는 해석문법과 생성문법을 대상으로 한다는 차이만 있을 뿐, 기본적으로 둘은 동등하다. 문법자동 수정기는 개념의 추가, 삭제, 수정으로 인해 기준단어에 변화가 왔을 때, 변화된 기준단어사전을 이용하여 각각의 해석문법과 생성문법을 수정해 준다. 개념의 수가 280여개이고 해석과 생성문법 파일이 모두 4개 그룹이므로 최대 1120개의 문법파일을 수정해야 하는데 수작업으로 하기에는 너무 많은 시간이 소요된다. 문법자동 수정기는 그 수고를 덜어준다.

4.3 생성시스템의 개선 및 후처리기

기존 Phoenix Parser의 생성시스템은 한 발화에서 생성된 모든 개념을 하나의 문장으로 생성했다. 그래서 한 발화 내에 여러 개의 최상위개념이 존재할 경우, 각 개념 간의 구분이 되지 않아 생성문의 이해가 어려웠다. 이를 해결하기 위해서 본 시스템에서는 최상위개념을 구분할 수 있도록 하여 생성문의 이해도를 높였다.

기존의 생성시스템이 만남, 약속 영역이라는 보다 한정된 영역에서 구현되었기 때문에 영어의 대소문자를 구분하지 않아도 별 문제가 없었다. 그러나 여행계획영역에서는 지명이나 인명 등 대문자로 표기해야 하는 경우를 소문자로 표기하면 생성문의 이해가 어려워진다. 본 시스템에서는 대소문자를 구분하기 때문에 생성문법에 대소문자를 구분하여 나타낼 수 있다. 그래서 관용적으로 대문자로 표기하는 요일 이름, 지명, 인명 등을 대문자로 표기할 수 있어 생성문의 이해도를 높였다. 그리고 생성문법에 숫자를 사용하면 보다 편하게 기술할 수 있고, 생성문의 이해도 쉬운 연도, 번호, 숫자와 함께 쓰이는 표현(5일, 6개, 8인 등) 등에서는 숫자를 사용할 수 있도록 생성시스템을 수정하였다.

영어 후처리기는 문장의 첫 문자를 대문자로 바꿔주고, 각 문장의 끝에 구두점을 찍어주고, 생성문 중 전치사가 연속하여 두 개 이상 나오는 경우는 하나로 만들어 주는 등의 처리를 하고 있다. 일본어 후처리기에서는 활용을 고려하여 분리된 단어들을 한 문장으로 만들어 주어야 하나 현재는 단순히 한 문장으로 만들어 주는 처리만 하고 있다. 한국어 후처리기는 조사의 이형태를 처리하여 올바른 문장이 생성되도록 하고 있다.

5. 평가 및 개선방안

5.1 평 가

평가는 전사된 문장과 음성인식의 결과 두 가지 종류의 입력에 대해 번역된 영어문장과 비교하여 토큰단위의 의미전달률과 발화단위의 의미전달률을 계산하였다. 발화단위의 의미전달률은 발화전체의 의미가 어느 정도 전달되었는지를 평가하고, 토큰단위의 의미전달률은 입력 발화 중 중심어(key-word)가 어느 정도 번역되었는지를 평가한다. 전사된 문장은 여행계획영역의 전사된 텍스트 1500발화 중 임의의 300발화(그림 3)를 대상으로 하였다. 음성인식 결과는 인식시스템의 테스트 결과인 283발화(그림 4)를 대상으로 하였다. 여기서 번역성공률이란 용어 대신 의미전달률을 사용한 이유는 개념기반 번역시스템이 발화자의 의도를 정확히 번역했는지 아닌지에 초점을 두고 있기 때문이다.

토큰단위의 의미전달률 평가는, 입력된 한국어 문장의 중심어가 번역된 영어문장에 어느 정도 나타나는지를 수치로 나타낸다. 예를 들면 한 발화에 5개의 중심어가 들어 있는데 이 중 4개가 제대로 번역이 되었다면 80%의 의미를 전달했다고 본다. 중심어는 각 발화의 의미 전달에 중요한 요소가 되는 의미단위를

말한다. 예를 들면 지명, 인명, 날짜, 시간, 의도를 표현하는 단어들(있다, 없다, 알고 싶다, 있습니까 등)이다. 번호나 날짜는 그 자체로 한 의미단위가 된다. 즉, 전화번호 일곱 내지 아홉자리가 통째로 한 의미단위가 된다. 위에 언급하지 않은 부분은 주관적으로 판단하여 점수를 매기도록 하였다. 평가에는 5명이 참가하였다.

발화단위의 의미전달률 평가는 아래의 세 가지 평가기준에 의해 하였다. 평가자는 평가지의 각 발화에 대해 아래의 세가지 중 하나로 점수를 매긴다. 번역결과가 없을 경우는 C로 하고, A와 B를 번역성공으로 보았다.

A: 의미전달이 잘 된 경우
B: 의미전달에서 부분적인 오류가 있지만 발화의 취지는 전달된 경우
C: 의미전달이 실패한 경우

평가의 결과는 〈표 2〉와 같다. 음성인식결과보다는 전사문장에 대한 의미전달률이 높게 나타난 것은 인식결과에 오류가 있기 때문이다. 인식률이 높아질수록 전사된 문장의 결과에 가깝게 나타날 것이다. 특히 전사문장의 토큰단위 의미전달률이 높게 나타난 것은 전사문장의 중요단어가 해석문법에 잘 반영되었기 때문이다.

I: 아 안녕하세요 저 노병선입니다 그 동안 잘 지내셨어요
O: Hello. I'm Noh Byeongsun. Hello.
I: 예 아주 저도 좋아요 어 이번에 전화한 게 다른 게 아니라 어 이버 우리 세째 딸이 이번에 웅변대회에서 수상을 했길래 올랜도 주변을 관광을 시켜 줄려고 하거든요 그래서 부탁이 있어서 전화 드린 거예요
O: Yes. That's good. Telephone number. I'm planning to make a trip Orlando. I'd like to get some information please.
I: 아 잘 됐군요 그러면 차액 요금은 얼마나 됩니까
O: That's good. How much is the difference in the fare?
I: 아 잘 됐군요 팩스 번호를 알려 드리죠 어 팩스 번호는 팔 이 에 삼 오 육 삼 팔 사 일 공 칠 입니다
O: That's good. I'll send the fax number. The fax number eight two three five six three eight four one zero seven.
...

(그림 3) 전사된 문장의 평가지
(Fig. 3) The evaluation sheets of the transcription data

I: 네 내용이 갈려고 하는데요 호텔방을 예약할려고 되나요
O: Yes. I'd like to reserve a room do you have?
I: 예 저는 어 정인섭이구요 예 천구백구십육년 오월 사일부터 오월 십일까지 구박을 할려고 합니다
O: Yes. I'm Einseop Cheong. I'm planning to stay from the fourth of May nineteen ninety six to the tenth of May during nine night.
I: 예 가족용으로 하구요 호텔에서 가 공항에서 호텔까지 리무진을 이용할 수 있습니까
O: Yes. I'll take a family room do you have?
I: 어 그러면 유월 지불은 어떻게 하면 되죠
O: In June, how can I pay for it?
...

(그림 4) 음성인식 결과의 평가지
(Fig. 4) The evaluation sheets of the recognized data by speech recognizer

발화단위의 의미전달률과 토큰단위의 의미전달률이 차이를 보이는 이유는 평가기준이 다르기 때문이다. 토큰단위는 전체의 의미보다는 중심어라 불리우는 토큰만이 대상이므로 발화단위의 의미번역률보다는 더 높게 나타나는 것이 당연하다. 그러나 발화에서도 중요한 역할을 하는 것은 역시 중심어이므로 크게 차이가 나지는 않는다.

〈표 2〉 전사 문장과 인식결과에 대한 평가 결과
〈Table 2〉 Coverage of transcribed vs. recognizer-decoded speech.

	토큰단위 의미전달률	발화단위 의미전달률
전사 문장	80	77
음성인식 결과	57	73(55)

인식 결과에 대한 발화단위의 의미전달률 73%는 인식의 결과 중 번역의 입력으로 부적격하다고 판단된 25%의 문장을 제외한 나머지를 가지고 평가한 것이다. 인식의 결과 중 의미를 추출할 수 없는 경우는 부적격한 문장으로 간주하였다. 괄호안의 55%는 부적격한 문장으로 판단된 인식결과를 번역실패로 간주했을 때의 의미전달률이다. 이렇게 음성인식결과를 반영한 경우 의미전달률은 55%로 크게 떨어진 이유는 인식결과 자체가 잘못되면 그것을 이용해서는 의미를 찾아내지 못하기 때문이다. 따라서 인식의 성능이 번역의 성능에 크게 영향을 미친다는 것을 알 수 있다.

5.2 개선을 위해 고려해야 할 내용

개념기반 시스템은 몇 가지 단점이 있다. 첫째는 같은 영역에서의 확장은 쉽지만, 다른 영역으로의 확장이 어렵다. 다른 영역에 적용될 경우 개념의 정의나 종류가 달라지므로 처음부터 다시 개념들을 정의해야 한다. 그리고 영역 간에 비슷한 개념이 적용되더라도 미묘한 차이가 많아 기존의 문법을 그대로 사용하기는 힘들다. 둘째로 부분자유어순처리가 힘들다. 부분자유어순을 개념기반 시스템을 이용하여 표현하려면 가능한 모든 경우를 문법에 기술하여야 한다. 셋째로 의미위주의 번역을 하기 때문에 수의 일치, 시제의 일치 등을 처리하기가 힘들다. 넷째로 한 개념내의 미묘한 차이가 있는 표현들을 그대로 번역하기가 힘들다. 예를 들어 "suggest"라는 개념을 간단하게 만들어 보자.

학교에서 만났으면 좋겠습니다.
학교에서 만나죠
학교에서 철수와 함께 만나죠
...

첫 번째와 두 번째 표현은 단순히 만나자는 것이지만, 세 번째 표현은 다른 사람과 같이 만날 것을 제안하고 있다. 위의 세가지 표현은 한 가지 개념으로 표현되어 있다. 이 경우 다른 사람과 함께 만나는 경우를 세분화하지 않고 그냥 한 개념으로 표현하였기 때문에, 생성될 때 다른 사람과 함께 만난다는 의미는 무시되게 된다. 다섯째로 상위개념은 동등한데 그것을 표현하는 하위개념을 두 언어가 동일하게 표현하기 어려울 경우가 있다. 예를 들면 숫자를 셀 때, 한국어는 네 자리가 기본이지만, 영어는 세 자리가 기본이다.

위와 같은 단점들을 보완하고, 시스템의 성능을 높이기 위해 앞으로 연구해야 할 분야는 많다. 부분자유어순을 해결하기 위해서는 한 개념의 하위 개념으로 동일 계층에 있는 개념들이 상호 자리바꿈이 가능하도록 프로그램하여야 한다. 해결방안으로 집합개념을 도입하는 방안과 격틀개념을 도입하는 방안이 검토 중이다. 영어의 수와 시제의 일치, 한국어의 의존명사와 존칭관련 표현 등으로 대표되는 정밀한 번역이 어려운 문제와 개념 내의 미묘한 표현들의 정확한 번역은 개념의 세분화로 해결할 수도 있지만, 개념의 세분화는 개념의 복잡도를 증가시키는 또 다른 문제를 가지고 있기 때문에 신중해야 한다. 상위개념은 동등한데 그것을 표현하는 하위개념을 두 언어가 동일하게 표현하기 어려운 경우는 따로 함수를 만들어 처리하는 방안을 검토 중이다.

현재의 기준단어를 이용한 토큰분리기에는 다양한 어미변화를 수용하기 힘들다는 단점이 있다. 이 단점을 해소하기 위해 형태소 해석기를 이용하는 방안이 연구 중이다. 형태소해석기가 보다 정확한 해석을 하지만, 처리시간이 늘어나고 많은 수의 후보를 출력하는 경향이 있어 여러 가지를 신중하게 고려한 후 선

택해야 할 것이다. 수작업에 전적으로 의존하는 개념의 정의와 문법파일의 작성은 시간과 노력이 많이 소요되기 때문에 작업(문법 작성, 검증 등)을 편하고 빠르게 할 수 있도록 도와주는 도구가 필요하다. 이 외에도 미등록어가 나올 경우 미등록어 자체를 번역하지 않고, 그대로 생성해 주는 방법을 이용한 미등록어 처리 방안도 강구 중이다.

6. 마무리

본 논문에서는 여행계획 영역의 대화체 음성언어 번역시스템의 번역부로, 한국어를 해석하여 영어, 일본어, 한국어로 생성해 주는 시스템에 대해 기술하였다. 개념기반 번역시스템은 간투사가 빈번히 나타나고 비정형문이 많은 대화체 문장과 오인식된 결과를 내보내는 음성인식의 결과를 처리하기에 적합한 시스템이다.

개념기반 번역시스템은 한국어 처리와 생성문의 이해를 높이기 위해 여러 부분에서 개선이 되었다. 여행계획 영역에 대한 280여개의 새로운 개념과 개념의 계층구조 구축, 한국어 해석문법과 영어, 일본어, 한국어 생성문법의 구축, 기준단어 사전과 토큰분리기의 개발로 인해 한국어를 해석하여 영어로 생성해 주는 시스템의 경우 75% 정도의 번역성공률을 보이고 있다. 이 밖에도 각 언어의 생성기에 후처리기를 두어 한국어 생성의 조사 붙이기, 일본어 생성의 공백 없애기, 영어 생성의 구두점 표시 및 대소문자 구별 등을 가능하게 하여 보다 자연스러운 생성문이 가능하도록 하였다. 최상위개념을 구분함으로써 개념간의 경계를 알 수 있게 하여 생성문의 이해를 높였다. 또한 생성문법에서 숫자가 무시되는 것을 수정하여 숫자 처리가 가능하도록 함으로써 숫자를 사용함으로써 이해가 쉬운 일본어 표현이나 숫자관련 표현들의 이해를 높였다. 마지막으로 영어의 대소문자 구분이 안 되는 것을 수정하여 영어에서 일반적으로 대문자로 표현하는 요일, 인명, 지명 등을 대문자로 표기함으로써 생성문의 이해를 높였다.

대화체의 다양한 언어현상과, 음성인식 결과의 오류 때문에 여행계획 영역에서 정의된 280여개 개념으로는 영역 전체를 다루기에는 부족한 점이 있다. 그러나 개념의 보충, 다양한 어미변화를 수용할 수 있는 토큰분리기의 개선이 이루어진다면 보다 나은 성능의 번역시스템이 가능할 것이다.

감사의 글

본 연구는 정보통신부 출연 "다중매체 환경하에서의 대화체 음성번역 통신기술 개발" 과제로 수행되었습니다.

참 고 문 헌

[1] 최운천, 한남용, 김영섬, "개념파서를 이용한 대화체 음성언어번역", '95 한국정보처리학회 학술발표논문집, pp. 159-163, 1995.

[2] 한남용, 최운천, 김영섬, "대화체 음성인식 및 기계번역을 위한 언어처리 연구," '95 신호처리합동학술대회 논문집, pp. 80-83, 1995.

[3] 이영직, 양재우, "다중매체 통신을 이용한 대화체 음성언어번역 시스템," 제13회 음성통신 및 신호처리 워크샵 논문집, pp. 101-106, 1996.

[4] A. Waibel *et al*, "JANUS-II : Translation of Spontaneous Conversational Speech," Proc. of the 1996 International Conference on Acoustics, Speech, and Signal Processings, vol. I, pp. 401-404, 1996.

[5] M. Rayner *et al.*, "Estimating Performance of Pipelined Spoken Language Translation Systems," Proc. of 1994 International Conference of Spoken Language Processings, Yokohama, Japan, vol III, pp. 1251-1254, 1994.

[6] M. Woszczyna, N. A. Waibel, F. D. Buo, N. Coccaro, *et al*, "JANUS93:Towards Spontaneous Speech Translation," Proc. of the 1994 International Conference on Acoustics, Speech, and Signal Processings, vol. 1, pp. 345-348, 1994.

[7] L. Mayfield, M. Gavalda, W. Ward, and A. Weibel, "Concept-Based Speech Translation," Proc. of the 1995 International Conference on Acoustics, Speech, and Signal Processings, vol. 1, pp. 97-100, 1995.

[8] M. Suzuki, N. Inoue, F. Yato, K. Takeda, and S. Yamamoto, "A Prototype of A Japanese-Korean Speech Translation System," Proc. of 4th Eur-

opean Conference on Speech Communication and Technology, vol. 3, pp. 1951-1954, 1995.

[9] M. Nagao, "Some Rationales and Methodologies for Example-based Approach," Proc. of Int'l Workshop on Fundamental Research for Future Generation of Natural Language Processing, pp. 82-94, 1992.

[10] O. Furuse and H. Iida, "Transfer-Driven Machine Transfer," Proc. of Int'l Workshop on Fundamental Research for Future Generation of Natural Language Processing, pp. 95-111, 1992.

[11] E. Levin, and R. Pieraccini, "Concept-Based Spontaneous Speech Understanding System," Proc. of 4th European Conference on Speech Communication and Technology, vol. 1, pp. 555-558, 1995.

〈부록 1〉 시스템의 실행 예

다음은 개념기반 번역시스템의 음성인식 결과에 대한 실행 예이다. 첫 번째 예는 "어떤"이 오인식된 것이지만 의미는 제대로 전달되었다. 두 번째 예는 전형적인 대화체 문장의 예로 의미를 전부 전달하지는 못했지만 주된 의미들은 전달되었다.

입력: 네 네명이 갈려고 하는데요 어떤 방을 예약할려고 하거든요

토큰분리기: 네 네 명 이 갈려고 하는데요 어떤 방 을 예약 할려고 하거든요

개념파서:

[response] ([yes] (네))

[give_info] ([no_of_person] ([no_of] (네 명)이 갈려고 하는데요))

[my_want] (방 을 [reserve] (예약)할려고 하거든요)

일본어 생성: はい四人です. 豫約をしたいんです

영어 생성: Yes. The number of people is four persons. I'd like to reserve a room.

한국어 생성: 예 네명입니다. 예약을 하고 싶은데요.

입력: 네 칠월 삼월은 설악산이구요 기간은 칠일입니다 그리고 사월 사일은 용인 자연농원이 아니요 시간은 사일입니다

토큰분리기: 네 칠 월 삼 월 은 설악산 이 구 요 기간 은 칠 일 입니다 그리고 사 월 사 일 은 용인 자연농원 이 아니요 시간 은 사 일 입니다

개념파서:

[response] ([yes] (네))

[give_info] ([temporal] ([point] ([date] ([month_name] (칠 월))

[date] ([month_name] (삼 월)은)))

[city] (설악산)이 구 요)

[give_info] ([period] (기간 은 [temporal] ([point] ([date] ([day_num] (칠 일))))입니다))

[temporal] ([point] ([date] ([month_name] (사 월) [day_num] (사 일)은)))

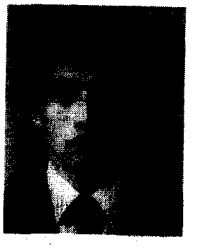
[response] ([no] (아니요))

[give_info] ([number] ([number_of_phone] ([digit] (사)[digit] (일)입니다)))

일본어 생성: はい七月三月は雪岳山です. 期間は七日です. 四月四日. いいえ. 四一です

영어 생성: Yes. Departing on in July. In March that's to Seorag mountain.
We wanna stay the seventh. The fourth of April. No. Four one.

한국어 생성: 예 칠월 삼월은 설악산입니다. 기간은 칠일입니다. 사월 사일. 아닙니다. 사일입니다.



최 운 천

1989년 2월 전남대학교 전산통계학과 (이학사)

1991년 2월 한국과학기술원 전산학과 (공학석사)

1991년 3월~현재 한국전자통신연구원 음성언어연구실 선임연구원

관심분야: 자연어처리, 음성언어번역

한 남 용

1985년 2월 충남대학교 계산통계학과 (이학사)
1987년 2월 충남대학교 대학원 계산통계학과 (이학석사)
1987년 11월~1991년 3월 공군사관학교 전자계산소
1991년 6월~현재 한국전자통신연구원 음성언어연구실 선임연구원
관심분야: 자연어처리, 코퍼스분석, 음성인식과 기계번역의 인터페이스

김 재 훈

1986년 2월 계명대학교 전자계산학과 (이학사)
1988년 2월 한국과학기술원 전산학과 (공학석사)
1996년 8월 한국과학기술원 전산학과 (공학박사)
1988년 2월~현재 한국전자통신연구원 음성언어연구실 선임연구원
관심분야: 코퍼스기반 언어처리, 음성언어번역