핵심개념 기반의 강건한 한국어 대화체 파싱

노 서 영† · 정 천 영†† · 서 영 훈†††

요 약

부분 자유어순 특성을 가지는 한국어를 CFG형태의 문법으로 기술했을 때 문법이 방대해지고 CFG형태의 문법을 파서가 이용할 때는 자연발화문의 특징인 간투어, 중복발화 등 때문에 불필요 성분을 따로 처리해야 하므로 파서의 부담이 커진다. 이러한 문제점을 해결하기 위해 본 논문에서는 발화문에서 중요한 의미를 가지는 요소를 '핵심개념'이라 정의하고 핵심개념만을 문법에 기술하여 문법이 방대해지는 것을 막고 문법에 기술된 핵심개념을 파싱요소로 선택함으로써 불필요 성분 처리에 대한 파서의 부담을 줄였으며 이렇게 단순화된 문법만으로도 정확한 파싱결과를 내줄 수 있음으로 보인다. 실험결과 '여행안내'영역 자연발화문에 대해서 평균 98%이상의 올바른 파싱결과를 얻어낼 수 있었다.

A Robust Korean Spoken Language Parsing Based on Core Concept

Seo-Young Noh† · Cheon-Young Jung†† · Young-Hoon Seo†††

ABSTRACT

The partial free order feature of Korean makes grammar size represented by CFG too big and that's why grammar has to contain all the ordered words. There are some problems to parse spoken language, because spontaneous spoken language has special features such as meaningless words, repetitious speech, etc. So, in this paper, we define 'Core-Concept' as the necessary element for parsing and we describe grammar only using Core-Concept. And we can prevent grammar from becoming very large and reduce an additional parsing burden as we select Core-Concept described in grammar as parsing element. Through this strategy, we present that the simplified grammar can give us more efficient method to get right results. Experiments show that our parsing strategy has an average of 98% or over success rate in correct parsing results.

1. 서 론

자연언어처리 연구분야에서 파싱은 크게 정형화된 입력문장의 처리와 인간이 일상생활에서 사용하는 자연 발화된 문장을 처리하는 메커니즘으로 분류할 수 있는데, 전자의 경우는 입력문장에 오류가 거의 없는 완벽한 문장이고 후자인 대화체는 문어체와 달리 대부분의 발화가 비문법적이고, 간투어나 중복발화 등 불필요한 성분을 많이 포함하고 있다[1,2].

이러한 이유는 발화자의 심리적 상태, 처해있는 환경 등 외부 요소에 의해 영향을 받기 때문이다. 기존의 문어체 파싱에서는 입력 문장이 문법적으로 옳다는 가정 하에 분석을 하기 때문에 비문법적인 현상이 빈번히 발생하는 자연발화 문장을 처리하기에는 많은 문제점을 내포하고 있다[3,4,5].

본 논문에서 제시하는 핵심개념 기반 파싱은 개념기

* 본 논문은 정보통신부의 정보통신 우수시범학교 지원사업에 의하여 수행된 것입니다.

† 준 회 원 : 충북대학교 대학원 컴퓨터공학과
†† 정 회 원 : 구미1대학 전자계산과 교수
††† 종신회원 : 충북대학교 컴퓨터공학과 교수

논문접수 : 1999년 3월 11일, 심사완료 : 1999년 6월 25일

반 파싱 방법론을 확장한 방법이다. 개념기반의 분석 기법[6,7,8]은 언어 공통원리에 기반하여, 강건성을 가장 큰 장점으로 가지며, 비문법적인 요소를 많이 포함하고 있는 자연발화 처리에 가장 유리한 기법 중 하나로 평가되고 있다. 1994년 ARPA ATIS 평가결과는 AT&T, BBN, CMU, MIT, MITRE, SRI, UNISYS에서 개발한 음성 언어처리 시스템 중 CMU와 AT&T에서 개발한 시스템을 가장 우수한 것으로 평가하고 있고 이들은 모두 개념 기반의 분석 기법을 이용하고 있다.

기존의 개념기반 분석 기법은 불필요한 개념으로 인한 파싱의 오버헤드와 한국어 부분 자유어순 특성을 CFG(Context Free Grammar) 형태의 문법으로 기술함으로써 문법이 방대해지는 문제점이 있었다.

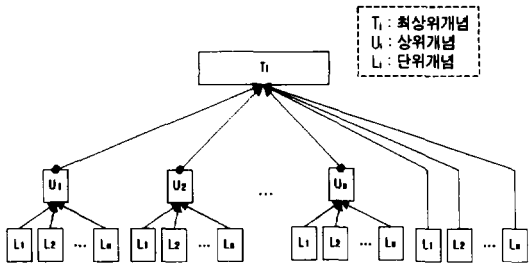
따라서 본 논문에서는 CFG 형태를 따르는 문법기술이 아닌 단순화된 핵심개념 기반 문법만으로도 성공적인 파싱을 수행할 수 있고 문법이 갖는 간결성이 문법의 오버헤드를 줄일 수 있음과 자연 발화문이 갖는 여러 가지 성질 중에서도 파싱에 불필요 성분들에 대한 처리 방법을 다른 메커니즘을 구현하지 않고 단순화된 문법 자체를 이용하여 제거할 수 있음을 보인다.

2. 핵심개념 기반 문법

2.1 개념의 구성

본 논문에서 제시하는 핵심개념 기반 분석 시스템의 문법은 '여행안내 (travel arrangement)' 영역 ETRI Corpus, 1575개의 발화문을 기반으로 하여 작성되었다. 여행안내 영역에서의 개념은 단어를 중심으로 하는 단위개념과 하나 이상의 단위개념이 형성하는 상위개념 그리고 단위개념이나 상위개념이 이루는 최상위 개념으로 개념 레벨을 3단계로 구분한다. 단위개념들은 대상 영역 Corpus를 분석하여 발화자가 의도하는 의미를 전달하기 위해서 사용되는 단어나 연속된 단어들에 부여하는 개념인데, 본 논문에서 구현한 시스템에서 형태소 분석과 전처리를 거친 토큰[1]이나 토큰열로 구성된다. 예를 들어 시간개념 [temporal]을 고려해 볼 때 단위개념은 year(년), month(월), day(일), dayofweek(요일), hour(시), minute(분)이 되고 이러한 단위개념은 시간개념 [temporal]이라는 상위개념을 이루게 된다. 이렇게 형성되는 개념들은 최상위 개념으로 발화자의 의도를 형성해 나간다. (그림 2-1)은 개념들의 레벨 계층을 나타낸 것이다. 이는 최상위 개념을 형성하기 위해서는 상위개념과 단위개념들의 조합으로 구성될 수 있다는 것을 보여주고 있다.

$L_1, \cdots, L_n$까지 일련의 단위개념이 상위개념 U_i를 형성하고, $U_1, \cdots U_n$과 $L_1, \cdots, L_n$의 조합으로 최상위 개념 T_i를 형성한다.



(그림 2-1) 개념 레벨 계층도

현재 핵심개념 기반 문법에 정의되어있는 최상위 개념은 7개로 구성되어 있다. 실험에 사용된 '여행안내' 영역 Corpus에서 7개의 최상위 개념만으로도 발화자가 의도하는 바를 모든 발화문에 대해 최상위 개념 결정이 가능하기 때문이다. 화자는 발화하고자하는 내용을 직설법으로 표현하는 경우가 대부분이어서 문어체와 같이 복잡한 구조로 형성되지 않는다는 것에 초점을 맞추면, 발화의 목적은 문장의 종결형에 따라 대부분이 구분될 수 있다. 발화문의 마지막 동사에 대해 decl(종결형), quest(의문형), please(청유형), will(의지형) 등으로 구분되어 나오는 정보를 이용하여 화자의 의도를 구분하였다. 최상위 개념과 이에 대한 내용은 <표 2-1>에 나타나 있다.

〈표 2-1〉 핵심개념 기반 문법 최상위 개념

최상위개념	핵심개념 기반 최상위 개념의 의미
[give_info]	청자에게 발화자가 정보를 주는 최상위 개념
[i_want]	발화자의 희망을 나타내는 최상위 개념
[i_will]	발화자의 의지를 나타내는 최상위 개념
[nicety]	인사말을 나타내는 문장에 최상위 개념
[query]	발화자의 질의를 나타내는 최상위 개념
[request]	발화자의 요구를 나타내는 최상위 개념
[respond]	짧은 응답을 나타내는 최상위 개념

1) 대화체 문장이 형태소 분석과 전처리를 거쳐서 파서의 입력으로 토큰열이 전달되는데, 토큰은 <@1 @2>의 형태를 갖는다. @1 필드는 토큰명을 나타내면 @2필드는 토큰명이 갖는 속성으로 구성된다.

최상위 개념은 단위개념과 상위개념의 조합으로 구성되는데 핵심개념 기반 문법에 정의된 상위개념을 <표 2-2>에 일부를 보였다.

〈표 2-2〉 핵심개념 기반 문법의 일부 상위 개념

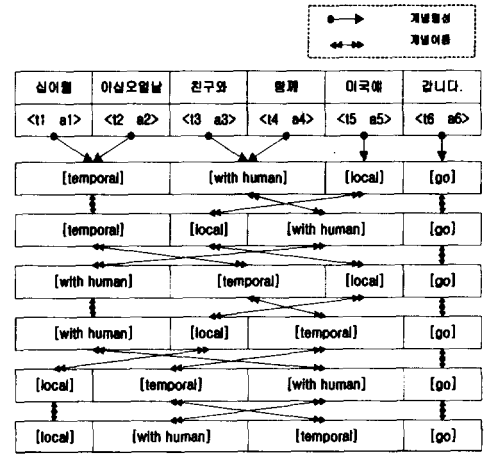
상위개념	핵심개념 기반 상위개념의 의미
[go]	'가다'의 의미를 내포한 개념
[apply]	'신청하다'의 의미를 내포한 개념
[know]	'알다'의 의미를 내포한 개념
[book]	'예약하다'의 의미를 내포한 개념
[offer]	'제공하다'의 의미를 내포한 개념
[attend]	'참석하다'의 의미를 내포한 개념
[meet]	'만나다'의 의미를 내포한 개념
[temporal]	시간의 의미를 내포한 개념
[local]	장소의 의미를 내포한 개념
[how]	방법의 의미를 내포한 개념

최상위 개념 [give_info]의 경우 [give_info]를 형성하는 상위 개념은 약 50여개 정도로 구성되어 있으며, 이를 구성하는 방법은 상위개념을 형성하는 방법과 동일하다.

2.2 핵심개념 기반 문법 구성

핵심개념 기반 문법은 대상 Corpus를 조사하여서 개념으로 묶일 수 있는 요소들을 부분 매칭 부분열들로 나열한 것이 아니라 한국어의 부분 자유어순[9,10]의 특성을 고려한다. 이는, 부분 매칭된 부분열들이 하나의 개념이든 토큰이든 간에 상위개념을 이룰 수 있는 개념의 어느 위치에도 올 수 있음을 의미한다. 이러한 특성을 이용하게되면 발화문에 대한 두 개 이상의 개념들이 의미의 변화를 갖지 않고 자유롭게 위치할 수 있는 경우에 위치 변화에 상관없이 문법을 기술할 수 있는 장점을 갖게 된다.

(그림 2-3)은 문장 "십이월 이십오일날 친구와 함께 미국에 갑니다."에 대한 부분 자유어순이 적용된 경우를 보여주고 있는데 동사 '가다'라는 개념 [go]에 대해서 [go]가 취할 수 있는 개념들은 입력문장에 대해서 [temporal], [with human], 그리고 [local] 개념을 취한다. 그러나 한국어의 부분 자유어순 특성으로 인해서 [temporal], [with human], [local] 개념들이 개념 [go]를 형성하기 위해서는 순열 ${}_3P_3$가지 이상으로 위치가 바뀔 수 있다.



(그림 2-3) 부분자유 어순이 적용된 경우 개념의 위치 변화

(그림 2-3)이 나타내고 있는 내용을 CFG형태의 문법으로 작성 할 경우에 (그림 2-4)와 같이 문법이 구성된다.

```
[go]
    (*+[temporal] *+[with human] *+[local])
    (*+[temporal] *+[local] *+[with human])
    (*+[local] *+[temporal] *+[with human])
    (*+[local] *+[with human] *+[temporal])
    (*+[with human] *+[temporal] *+[local])
    (*+[with human] *+[local] *+[temporal])
```

(그림 2-4) 부분 자유어순이 적용된 문법

문법의 구성 성분 중에서 '*'은 선택적 성분을 나타내며 '*'이후의 개념은 나올 수도 있고 나오지 않을 수 있음을 나타내고, '+'은 반복적 성분으로 '+'이후의 개념은 한번 이상 나올 수도 있음을 나타낸다. 선택적 성분과 반복적 성분이 합쳐진 문법의 표기는 '*+'로 나타내는데 '*+' 이후의 개념은 나오지 않을 수 있고 또한 한번 이상 발화문에서 개념의 형태로 존재할 수 있음을 의미한다[11].

(그림 2-4)에서 보여주고 있는 것과 같이 한국어의 부분 자유어순 특성이 문법에서 적용되어 문법으로 모든 순서를 처리해야하는 것은 문법량이 방대해지는 것을 막을 방법이 없다. 이렇게 모든 순서를 나열해서 문법으로 표현하는 것은 효율적이지 못하다

이러한 점에서 한국어의 부분자유 어순 특성을 고려

할 때, 하나의 개념으로 묶일 수 있는 요소들은 개념 안에서 자유롭게 위치이동이 가능하게 된다. 이는 개념을 집합으로 간주하고 개념으로 묶일 수 있는 토큰들을 집합의 요소로서 간주한다면 쉽게 문법을 구성할 수 있게 된다.

단위개념은 토큰이나 토큰열로 구성되는데 단일토큰은 토큰명(token name) t_i와 토큰에 대한 속성(attribute) a_i로 구성되고, 토큰위치 i에서 단일 토큰은 $\langle t_i, a_i \rangle$가 된다. 따라서 개념 $[C_r]$을 구성하는 토큰열이 $\langle t_1, a_1 \rangle, \cdots, \langle t_n, a_n \rangle$이라고 할 경우에 토큰열로만 구성된 핵심개념 기반 문법은 [식 2-1]과 같이 표현할 수 있다.

T : Token name Set
A : Attribute Set
N : Natural number

$$[C_r] = \{\langle t_1, a_1 \rangle, \langle t_2, a_2 \rangle, \langle t_3, a_3 \rangle, \cdots, \langle t_{n-1}, a_{n-1} \rangle, \langle t_n, a_n \rangle\}$$
$$= \{\langle t_i, a_i \rangle : t \in T,\ a \in A,\ i \in N,\ 1 \le i \le n\}$$

[식 2-1] 토큰으로 구성된 핵심개념 기반 문법

여기에서 개념은 단일 토큰 또는 토큰열로 구성되므로 토큰 $\langle t_i, a_i \rangle$나 토큰열 $\langle t_i, a_i \rangle, \cdots, \langle t_k, a_k \rangle$를 개념 $[C_i]$, $[C_k]$등으로 표현할 수 있다. 따라서 [식 2-1]을 일반화시키면 [식 2-2]와 같이 나타낼 수 있으며 최상위 개념은 개념들의 합집합 [식 2-3]과 같이 나타낼 수 있다.

CON : Concept Set

$$[C_r] = \{\langle t_1, a_1 \rangle, \cdots, \langle t_d, a_d \rangle, [C_i], \cdots, [C_k]\}$$
$$= \{\langle t_i a_i \rangle, [C_j] : t \in T,\ a \in A,\ C \in CON,\ i \in N,\ j \in N,\ 1 \le i, j \le n\}$$

[식 2-2] 일반화된 핵심개념 기반 문법

$$[C_{\text{toplevel}}] = \bigcup_{i=1}^{last} [C_i]$$

(단, $[C_{\text{toplevel}}]$는 최상위 개념)

[식 2-3] 최상위 레벨 핵심개념 기반 문법

문법을 구성할 때 대상 Corpus 문장을 동사 단위로 분리하고 i번째 동사 V_i와 $i+1$번째 동사 V_{i+1} 사이에 있는 토큰열을 개념으로 생성한다. 토큰열에서 하나의 토큰이 개념을 형성할 수 있고 하나이상의 토큰이 개념을 형성할 수 있다. 형성된 개념들이 동사 V_{i+1}에 필요한 성분인지를 검사하여 만일 필요한 성분이라면, 동사 V_{i+1}를 나타내는 개념에 대한 문법 G_{i+1}의 핵심개념으로 취해지게 된다. 이러한 절차로 모든 Corpus의 문장을 대상으로 동사 V_{i+1}이 가질 수 있는 개념들을 문법 G_{i+1}에 기술함으로써 동사에 대한 문법이 구성된다. 또한 Corpus에서 수식 받는 대상에 대한 문법을 구성해야 하는데, 예를 들어 '~로 가는 비행기'라는 문장이 있을 경우 개념 [go]는 '비행기'를 수식하여 '비행기'에 대한 개념 [aircraft]에 포함되게 된다. 수식 받는 쪽의 문법 또한, 전체 Corpus를 대상으로 문법을 구성하였다. [알고리즘 2-1]은 문법을 구성하는 방법을 나타낸 것이다.

```
// S : Sentences set
// Cur_sen : Current sentence
// G : Grammar set
// t_i : token
// Gra_curVerb : Grammar of current verb
AlgorithmConstructGrammar
for each Cur_sen in S do
    firstVerb ← 0;
    SetPositionNumber( Cur_sen )
    while(SeekVerb( Cur_sen )) do
      secondVerb ← verbPosition;
      for(i←firstVerb+1; i≤secondVerb-1; i++) do
        if t_i is element for Gra_curVerb in G
        then
          AddCoreConcept( t_i );// Gra_curVerb ← t_i
          if (IsModified( t_i ))
          then
            CreateGrammarFor( t_i );
            AddCoreConcept(firstVerb)
          endif
        endif
      endfor
    endwhile
  endfor
```

[알고리즘 2-1] 문법구성 알고리즘

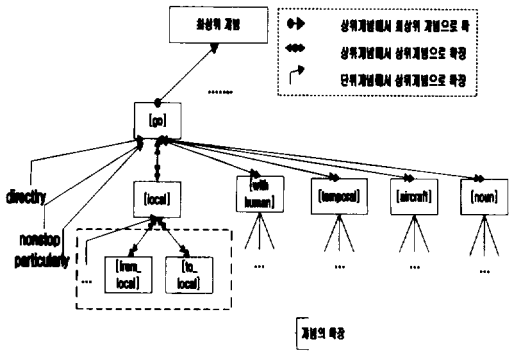
이를 적용하여 동사 '가다'에 대한 '여행안내'영역 Corpus에 대해서 핵심개념 기반 문법을 구성하면 (그림 2-5)와 같이 구성된다.

```
[go]
    directly
    nonstop
    particularly
    [aircraft]
    [local]
    [with human]
    [noun]
    [temporal]
```

(그림 2-5) 동사 '가다'에 대한 핵심개념 기반 문법

(그림 2-5)에서 보이는 directly, nonstop, particularly, [aircraft], [local], [with human], [noun], [temporal]은 대상 Corpus를 통해서 얻어진 개념 [go]에 대한 핵심개념들이다.

괄호 [ ]로 묶이지 않는 directly와 nonstop은 문법에서 단위개념을 나타내고 괄호 [ ]로 묶인 [local], [temporal]은 상위개념을 나타낸다. 예를 들어 [local]의 경우는 다시 [from_local], [to_local]등을 핵심개념로 가질 수 있다. (그림 2-6)은 [go]를 형성하는 개념들의 계층구조를 보여주고 있다.



(그림 2-6) [go]를 형성하는 개념들의 계층구조

3. 핵심개념 기반 파싱

기존의 개념기반 파싱의 문법구성은 코퍼스를 분석하여 개념을 이룰 수 있는 요소들을 CFG로 기술하고 있다. 이러한 방법은 한국어가 부분자유어순 특성을 처리하는데 상당한 문제점을 내포하게 된다. 일상생활에서 사용하는 한국어의 자연 발화문에 일정한 어순을 정해놓고 문법을 기술하기에는 문법의 크기가 한국어의 부분 자유어순 특성으로 인해서 방대해질 수 있기 때문이다. 어순이 일정하게 정의된 문법에 대해서 파서가 파싱을 수행하다가 파싱에 불필요 성분을 만나게 될 경우 파싱을 성공적으로 이끌어 낼 수 없는 경우가 발생하게 되는데, 이러한 경우 불필요 성분 제거에 대한 처리를 해줄 모듈을 부가적으로 파서에 설계해야 한다.

따라서 이러한 문제점을 해결하기 위해서 핵심개념 기반 방법론에서는 각 개념이 Corpus 내에서 취할 수 있는 요소들을 '핵심개념'이라 정의하고 개념을 이루는 요소들만을 선택해서 파싱 요소로 간주하여 파싱을 수행해 나간다.

3.1 핵심개념 기반 한국어 대화체 분석 시스템의 불필요 성분 처리

문법에 모든 어순을 정의하고 기술한 문법을 사용한 시스템의 경우는 파싱에 불필요 성분을 처리하기 위해서 여러 가지 방법을 모색해야 한다. 그 중에서 skip 방식에 의한 처리 방법이 있다[11,12,13].

시스템[13]에서는 비정형 문법요소에 대해서 파서가 파싱을 수행해 나가면서 skip을 하는데 이때 파싱 결과가 하나이상 생성될 때는 skip counter를 두어 최소 개수가 skip된 파싱결과를 옳은 결과로 간주하여 선택한다.

시스템[11]에는 skip의 개수를 1, 2, 3과 같이 달리함으로써 불필요 성분을 처리하는데 이에 대한 성능은 '여행안내' 영역 ETRI Corpus 1,575개의 발화문에 대한 성공률이 <표 3-1>과 같다.

최대 토큰 skip 개수	1	2	3
성 공 률	72%	72%	71.2%

〈표 3-1〉 시스템[11]에서 토큰 skip 개수를 달리한 평가

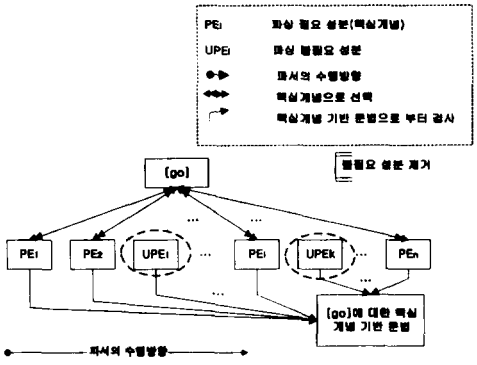
여기에서 중요한 점은 <표 3-1>이 제시하는 것과 같이 skip의 개수가 많아지는 것이 성공률이 높아진다는 것을 보장할 수 없는 것이다. 또한 skip 1, 2, 3에 대해서 시스템이 처리해야 할 추가의 비용이 발생하게 되는데 skip의 개수를 i로 했을 경우 연속된 i개 토큰에서 실패를 하고 $i+1$번째 토큰에서 성공하여 마지막 n번째에 가서 개념으로 묶이지 않는 경우가 된다. 이러한 비용은 가우스 함수(Gauss function)를 이용하여 [식 3-1]로 표현된다.

G : 문법이 포함하는 토큰열의 평균개수
n : 토큰의 수

$$f_i(n) = G \times ([\frac{i}{i+1} \times n])$$

[식 3-1] Skip i에 대한 파서의 추가비용

그러나 핵심개념 기반 문법을 적용한 한국어 대화체 분석 시스템의 경우는 문법을 통해서 불필요 성분이 제거되는 효과를 갖기 때문에 불필요 파싱 성분을 처리하는 별도의 메커니즘을 설계할 필요가 없다. (그림 3-1)은 핵심개념 기반 파서의 불필요 성분에 대한 파서의 처리를 보여준다.



(그림 3-1) 불필요 파싱 성분에 대한 처리

파싱요소 PE(Parsing Element:핵심개념)와 불필요 파싱성분 UPE(Unnecessary Parsing Element)가 n개의 개념열로 구성되어있고 상위개념 [go]를 형성하기 위해 파서는 각 개념을 순차적으로 선택하고 [go]의 문법에 핵심개념으로 정의되어있는지 검사한다. 만일 존재할 경우 현재 선택된 개념은 PE가 되고 그렇지 않은 경우는 UPE가 된다. 이렇게 해서 걸러진 UPE는 제거되고 선택된 $PE_1, \cdots, PE_n$까지가 [go]를 이루는 핵심개념이 된다.

[go]에 대한 문법구성 방법과 마찬가지로 파서는 동사단위로 파싱을 하는데 [go]를 형성하는 동사와 이 전동사 사이의 개념들에 대해서만 PE와 UPE를 구별하게 된다.

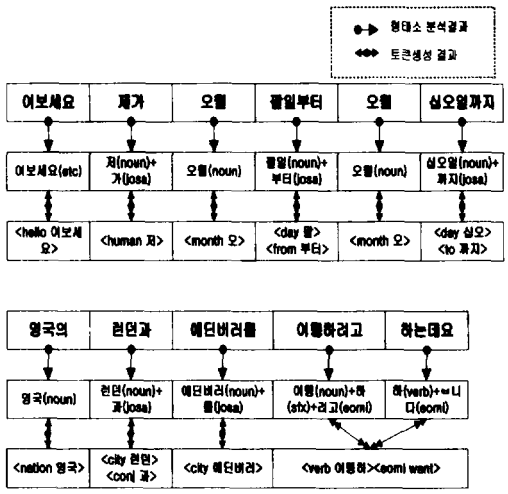
이미 3장 서두에서 밝힌바와 같이 '핵심개념'을 정의함으로써 파서는 문법에 기술된 내용을 파싱요소로 취함으로써 불필요 성분 처리를 위한 비용이 들지 않는다. 이는 파싱결과에 영향을 미치지 않는 간투어나 불필요 파싱 성분에 대해서도 정확한 결과를 제공할 수 있다는 장점을 갖게 된다.

3.2 핵심개념 기반 파싱을 위한 토큰열의 구성

핵심개념 기반 파서는 파싱을 수행하기 위해서 형태소 분석 시스템과 전처리 모듈을 거친 토큰열 정보를 이용해서 파싱을 수행해 나가는데 입력으로 들어오는 토큰열의 구성은 <@1 @2>형태로 구성된다. @1 필드는 개념 필드로서 토큰명(token name)을 나타내고 @2 필드는 속성(attribute)를 나타낸다. 이것은 입력 '오월'에 대해서 형태소 분석과정을 거치면 하나의 명사로 분석되지만 전처리를 통해서 '오'는 숫자를 속성으로 하고 '월'이라는 음절에서 개념 즉, 토큰명 'month'를 인식하여 '<month 오>'와 같이 처리하여 출력해 준다. 토큰열 구성 예를 (그림 3-2)에 나타내었다.

전사한 문장 :

여보세요 제가 오월 팔일부터 오월 십오일까지 영국의 런던과 에딘버러를 여행하려고 하는데요



(그림 3-2) 토큰열 구성

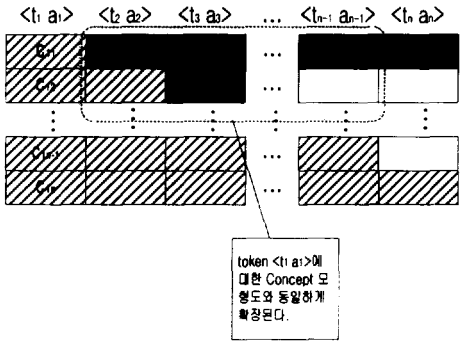
3.3 핵심개념 기반 파싱 알고리즘

하나의 개념은 다른 개념의 핵심요소로서 포함될 수 있으며 단일 토큰들 또한 단위개념으로 묶일 수 있고 연속된 토큰열이 하나로 묶여 상위개념의 핵심요소가 될 수 있다. 따라서 토큰의 현 위치 i가 하나의 개념 $[C_i]$가 되고 i와 $i+1$이 묶여 개념 $[C_{ik}]$를 이룬다. k는 토큰 i에서 k번째 개념을 의미한다.

t_i : i번째 token name

a_i : i번째 attribute

C_{ij} : token i에서 갖을 수 있는 j번째 Concept



(그림 3-3) 토큰에서 개념 구성

(그림 3-3)은 이러한 내용을 나타내고 있는데, C_{21}의 경우는 $C_{11}, C_{12}, \cdots, C_{1n}$과 같이 $C_{21}, C_{22}, \cdots, C_{2n}$으로 확장되고 C_{31}부터 $C_{(n-1)1}$까지도 같은 형태로 확장된다.

이러한 정보를 이용하여 핵심개념 기반 대화체 분석 시스템은 대화체를 입력받아서 형태소 분석기가 내준 결과를 전처리를 통해서 출력되는 토큰열 $\langle t_1, a_1\rangle$, $\cdots, \langle t_n, a_n\rangle$을 입력으로 받는다. 다음 토큰열의 개수를 파악해서 메모리에 토큰정보와 함께 저장하고 동사를 찾은 다음, 찾은 동사에 대한 문법 파일을 메모리에 로드하고 문법파일에 기술된 내용에 대해 이전 동사 다음부터 현재 동사 사이의 요소만 현재 동사의 파싱요소로 취하는데, 취해지는 모든 요소가 단일 개념으로 묶일 수 있는 개념의 핵심 개념이 되는 것이다. 이렇게 해서 묶인 개념이 이전 동사의 개념에 연결될 수 있는지 검사하게 되는데, 만일 연결 가능하면 현 동사 개념의 요소 개념으로 취한다. 이때 개념은 상위 개념의 핵심개념이 된다. 이러한 작업을 파서가 토큰열의 마지막에 위치하게 될 때까지 반복하게 된다. 이에 대한 파싱 알고리즘은 [알고리즘 3-1]과 같다.

```
// Con_cur : 현재 문장에서 형성된 개념 집합
// Gra_curVerb : 현재 동사가 형성하는 개념의 문법
// Con_curMod : 수식받는 개념 집합
//UPE : Unnecessary Parsing Element
//PE : Parsing Element

AlgorithmCoreConceptBasedParsing
m_nOldPosition ← 0;
m_pTokenString ← GetTokenString();
m_nTokenCount
  ← GetNumOfToken(m_pTokenString);
tmpTokenString ← m_pTokenString;
for( i←1; i≤m_nTokenCount; i++) do
  if(IsVerb(tmpTokenString))
  then
    LoadGrammar(m_pVerb);
    for j←m_nOldPosition; j≤i; j++;) do
      m_pElement ← GetToken(j);
      if m_pElement equal PE
                and PE in Gra_curVerb
      then
        Con_curVerb ← AddConcept(m_pElement)
      else
       //m_pElement equal UPE
         Remove(m_pElement)
       endif
     endfor
     m_nOldPosition ← i;
    else
      if(IsModified(tmpTokenString)
      then
        m_pModified ← tmpTokenString;
        LoadGrammar(m_pModified);
        Con_curMod ← CreateConcept(m_pModified)
      endif
  endif
    //토큰열의 위치를 하나 증가 시킴
    tmpTokenString←tmpTokenString+1;
endfor
while(IsThereConcept( Con_cur)) do
   Result ← ConstructTopLevel( Con_cur)
   ShowParsingResult(Result);
endwhile
```

(알고리즘 3-1) 핵심개념 기반 파싱 알고리즘

핵심개념 기반 파싱 알고리즘 [알고리즘 3-1]에는 skip 방식과 같은 파싱 불필요 성분에 대한 메커니즘에 대한 기술을 하고 있지 않다. 핵심개념 기반 문법 자체가 파싱에 불필요한 성분을 제거하는 기능을 제공해주기 때문이다. 간투어나 불필요 성분이 토큰열로 구성될 경우에 문법에 기술되지 않은 성분을 파싱 요소 즉, 핵심개념으로 선정하지 않기 때문에 불필요 성분에 대한 별도의 메커니즘을 들 필요가 없게 된다. 이렇게 문법이 한국어 부분 자유어순의 특성을 제공하고 불필요 성분에 대한 제거기능을 제공함으로 문법의 오버헤드를 줄이고 파서의 부담을 크게 줄일 수 있다.

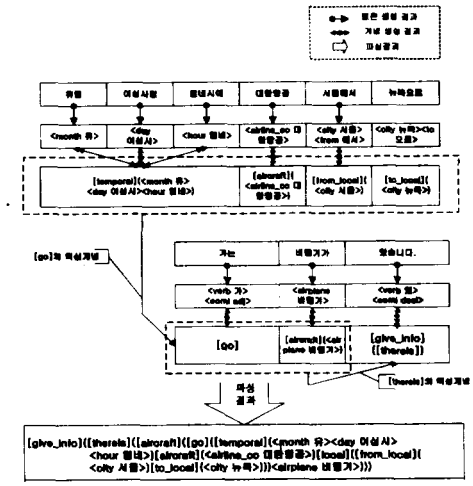
3.4 분석결과

본 논문의 핵심개념 기반 한국어 대화체 분석기의

구현은 Digital ALPHAstation 255 Unix용 C++를 이용해서 구현되었으며 전처리를 통한 토큰열에 대한 파싱 결과는 (그림 3-4), (그림 3-5), (그림 3-6)에 나타나 있다. (그림 3-4), (그림 3-5), (그림 3-6)에 나타난 결과는 최상위 개념 레벨이 정보를 제공하는 [give_info], 질의를 나타내는 [query], 어떤 것을 요구하는 [request]를 최상위 개념으로 하는 파싱 결과들이다.

입력문장 :

유월 이십사일 열네시에 대한항공 서울에서 뉴욕으로 가는 비행기가 있습니다.



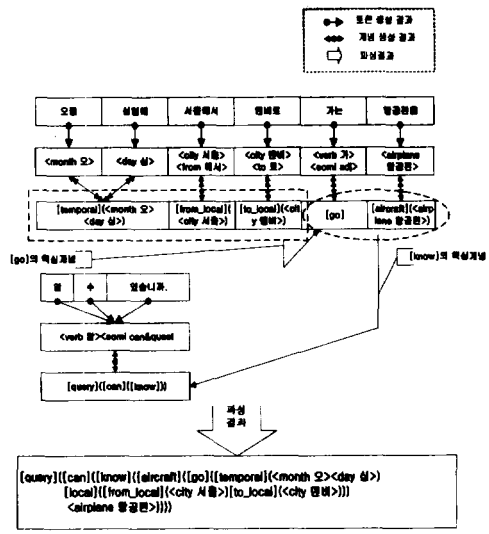
(그림 3-4) [give_info]를 최상위 개념으로 갖는 파싱결과

(그림 3-4)에서 보이듯 토큰열 <month 유><day 이십사><hour 열네>은 상위개념 [temporal]을 형성하고, <city 서울><from 에서>는 [from_local], <city 뉴욕><to 으로>는 [to_local]로 형성된다. [temporal], [aircraft], [local]은 다시 상위개념 [go]의 핵심 개념이 되어 상위개념 [go]를 형성한다. [aircraft]을 이루는 핵심개념 중에 [go]가 존재하여 [aircraft]([go]) 형태를 이루게 된다. 다시 [thereis]는 [aircraft]을 포함함으로서 파서에 의한 결과는 [thereis]([aircraft]([go]))가된다. 발화자의 의도를 나타내는 최상위 개념으로 묶으면 [thereis]는 [give_info]의 핵심개념으로 취해지게 되므로 [give_info]를 최상위 개념으로 하는 파싱결과를 얻게 된다.

(그림 3-5)와 (그림 3-6)에서 보이는 최상위 개념 [query], [request]에 대해서도 (그림 3-4)의 [give_info]와 같은 방법으로 파싱결과가 형성된다.

입력문장 :

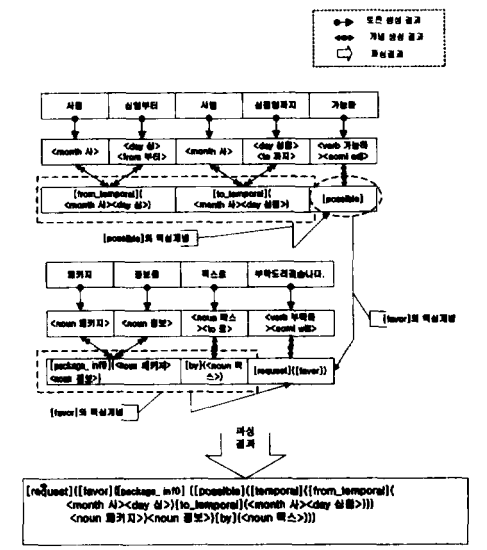
오월 십일에 서울에서 덴버로 가는 항공편을 알 수 있습니까.



(그림 3-5) [query]를 최상위 개념으로 갖는 파싱결과

입력문장 :

사월 십일부터 사월 십칠일까지 가능한 패키지 정보를 팩스로 부탁드리겠습니다.

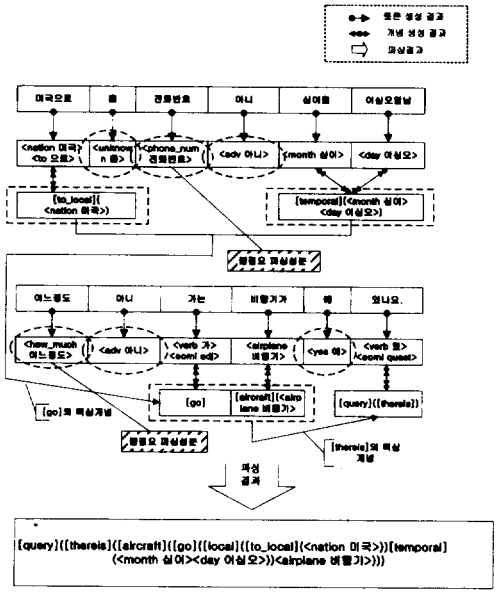


(그림 3-6) [request]를 최상위 개념으로 갖는 파싱결과

대화체 발화문에 파싱 불필요 성분이 들어있는 경우가 있는데 (그림 3-7)은 불필요 성분이 대화체에 들어있을 때 파싱을 수행한 결과이다.

입력문장 :

미국으로 음 전화번호 아니 십이월 이십오일날 어느정도 아니 가는 비행기가 예 있나요.

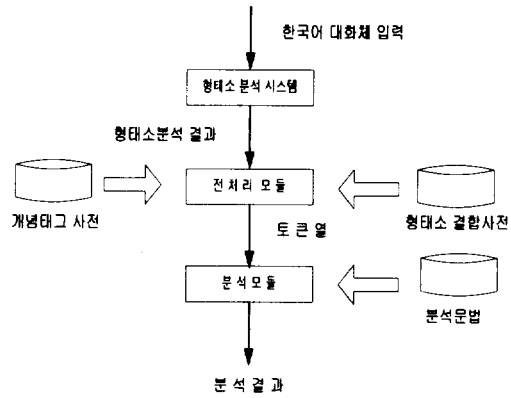


(그림 3-7) 불필요 파싱 성분을 제거한 파싱결과

(그림 3-7)이 나타내는 결과와 같이 간투어 '음'이나 '전화번호' '예' '아니'등은 개념 [go]를 형성하는 문법을 검사하여 UPE로 결정되기 때문에 파싱 대상에서 제거된다. 불필요 파싱성분이 제거되어, 파싱결과는 발화자의 의도와 일치하는 결과를 얻어낸다.

4. 실험 및 평가

핵심개념 기반 시스템의 전체적인 구성은 형태소 분석 시스템, 전처리 모듈, 분석모듈로 구성되어 있으며, 먼저 한국어 대화체가 형태소 분석 시스템으로 입력이 되고 형태소 분석 시스템은 대화체 입력에 대한 형태소 분석 결과를 출력한다. 형태소 분석 결과는 전처리 모듈의 입력으로 전달되는데 전처리 모듈에서는 개념태그 사전과 형태소 결합사전을 이용해서 토큰열을 출력하게된다. 토큰열은 분석모듈의 입력으로 전달되고 분석모듈은 분석문법을 이용하여서 입력된 토큰열을 분석하고 번역을 위한 분석결과를 출력한다. 이러한 처리과정은 (그림 4-1)과 같이 구성된다.



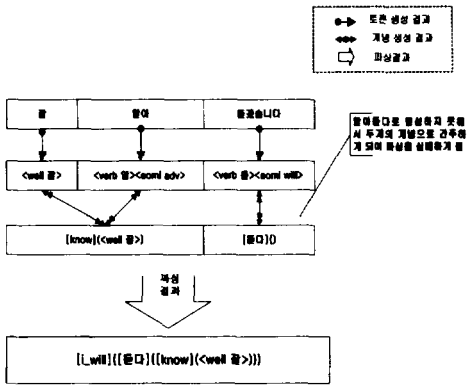
(그림 4-1) 시스템 처리 과정

본 논문의 실험은 전화상의 대화를 전사한 '여행안내'영역의 말뭉치를 대상으로 하였으며 <표 4-1>은 '여행안내'영역 ETRI Corpus 1575개의 문장 중에서 644개의 훈련된 문장에 대한 실험결과이다. 파싱의 성공 여부를 어떠한 형태로든 개념으로 묶여 나오지 않은 경우를 완전실패로 간주하고 올바른 최상위 개념(top-level concept)으로 묶이지 않는 경우나 개념 단위로는 묶이나 의미를 잃어버린 경우를 부분실패로 설정하였다. 그리고 성공은 완전실패와 부분실패를 제외한 파싱결과가 원문의 의미를 담고있고 바르게 파싱된 결과만을 취하였다.

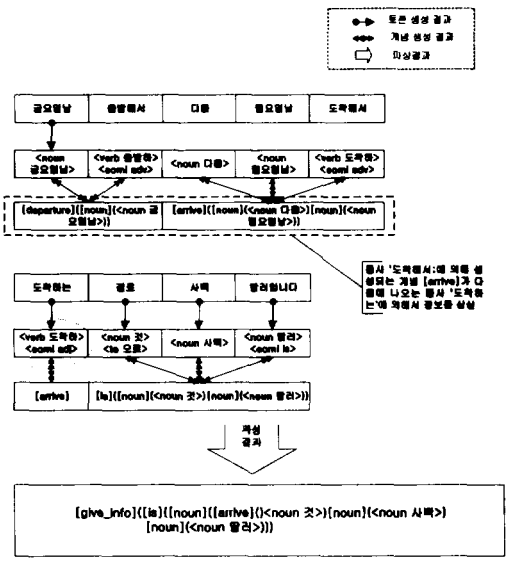
<표 4-1> 훈련된 발화문 644개에 대한 파싱 성공률

문장수	완전실패	부분실패	성공	성공률
644	0	9	635	98.6%

부분 실패문장 9개가 실험결과로 나왔는데, (그림 4-2), (그림 4-3)에 부분실패한 문장의 특성과 실패한 원인을 나타내고 있다. (그림 4-2)에서는 '알다'라는 동사에 '듣다'라는 보조용언이 결합된 형태로 '알아듣다'라는 [know]의 개념을 형성하지 못하고 '알다'와 '듣다'라는 다른 개념으로 간주하여 파싱을 시도하려는 문제가 발생했다. 이는 보조용언에 대한 처리를 하지 못한 전처리 모듈의 오류로 보조용언의 처리를 할 수 있도록 시스템을 보완해야 한다.



(그림 4-2) 파싱에 실패한 예 1



(그림 4-3) 파싱에 실패한 예 2

(그림 4-3)의 경우는 같은 동사가 두 번 나온 경우로 이전의 동사정보가 다음에 나오는 같은 동사 때문에 이전의 정보를 모두 잃어버리는 경우가 발생한 것이다. 이는 파서가 동사를 중심으로 파싱을 수행하기 때문에 아무리 이전에 동사에 대한 개념을 형성하였더라도, 다음에 나오는 똑같은 동사에 대해 개념을 형성해야 하므로 이전 정보가 무시되는 것이다. 이에 대해서는 파서가 같은 동사에 대해서는 다른 메커니즘으로 파싱을 수행할 수 있도록 방법을 간구 해야 할 필요성이 있다. 그러나 이러한 실패의 예가 있음에도 불구하고 핵심개념 기반 파싱 방법으로 파싱했을 때 훈련된 발화문에 대해서 평균 98%이상의 올바른 파싱 성공률을 보였다.

5. 결 론

현재 한국어 대화체 분석기의 문법 구성이 Corpus를 분석해서 가능한 문장을 CFG 형태의 문법으로 구성할 때, 이것의 문제점은 모든 형태의 문장을 CFG로 표현해서 문법으로 기술할 수 없다는 것에 있었다. 한국어와 같이 부분 자유어순이 적용되는 언어적 특징에서는 더욱더 문제가 되는 부분이다.

이에 대한 해결 방법으로 본 논문에는 Corpus를 분석해서 개념이 가질 수 있는 다른 개념 즉, 핵심개념들만을 문법에 기술함으로써 문법의 크기를 줄일 수 있었으며, 핵심개념 기반 문법의 기술이 처리하고자하는 개념에 대해 필요한 개념 성분만을 파싱 하게 됨으로 문법에 기술되지 않은 모든 성분은 불필요 성분으로 간주하게 되어 단순화된 문법만으로도 한국어 자연발화문을 파싱할 때의 문제점인 불필요 파싱 성분을 제거할 수 있고 파서가 불필요성분을 처리하는데 들었던 파싱의 오버헤드를 줄일 수 있는 장점을 얻을 수 있었다. 실험에서 간투어나 파싱에 필요하지 않은 불필요 성분의 입력에 대해서도 핵심개념 기반 파서는 정확한 파싱 결과를 내줄 수 있었으며, 98% 이상의 파싱 성공률을 얻을 수 있었다.

논문에서 제시하는 개념은 동사와 명사를 중심으로 하는 개념이기 때문에, 코퍼스를 대상으로 문법을 작성함에 있어서 새로운 동사가 나오면 이에 대한 개념을 정의해야하고 개념에 대한 문법을 구성해야 한다. 제한된 영역과 코퍼스를 통해 문법이 구성되기 때문에, 다른 영역에 적용될 경우 많은 부분은 재구성해야 하겠으나 동사와 명사를 중심으로 개념을 정의한 것은 영역의 이식성을 높이기 위한 하나의 방법이다.

핵심개념 기반 파싱 방법론을 발전시키기 위해서는 더 많은 자료를 대상으로 실험을 해야겠으며, 또한 현재 구현한 시스템이 사용하는 상위개념들이 대부분 동사를 중심으로 기술되어있기 때문에 파싱에 불필요한 성분이 동사로 나오게 된다면 파싱의 결과는 보증할 수 없는 결과를 낼 수 있다. 이러한 경우는 현재 구현한 ETRI Corpus '여행안내'영역 1575개의 발화문에서는 나타나지 않았지만 핵심개념 기반 파서의 확장을 고려할 때 다른 Corpus에서 이런 현상은 나타날 가능

성이 충분히 있어서 이를 확인하고 처리할 수 있는 메커니즘이 필요할 것이다.

참 고 문 헌

[1] 노서영, 정천영, 서영훈 "개념간 상호정보를 이용한 효율적인 개념기반 한국어 대화체 파싱", 한글 및 한국어정보처리, pp.365-369, 1998. 10.

[2] 정천영, 서영훈, "의미패턴에 기반한 대화체 한영 기계번역", 한국정보처리학회 논문지, Vol.5, No.9, pp.2362, 1998.

[3] 서영훈, "음성 언어 번역을 위한 개념 기반의 한국어 분석 및 생성", 정보과학회 논문지, Vol.23, No.11, pp.1176-1177, 1996.

[4] 최재웅, "대화분석에 있어서의 몇가지 문제: 호텔 예약 전화 대화를 중심으로", 한글 및 한국어정보처리, pp.8-15, 1996.

[5] 이현정, 서정연 "문장의 화행을 반영한 한-영 대화체 기계번역", 한글 및 한국어 정보처리, pp.272-272, 1997.

[6] B.Suhm., et al., "JANUS : TOWARDS MULTILINGUAL SPOKEN LANGUAGE TRANSLATION", Interactive System Laboratories, Carnegie Mellon University, 1995.

[7] Mayfield, L., M.Cavalda, Y-H Seo, N. Suhm, W. Ward, A. Waibel, "Parsing Real Input in JANUS: A Concept-based Approach to Spoken Language Translation," Proceeding of TMI95, 1995.

[8] Levin, E. and R. Pieranccini, "Concept-based Spontaneous Speech Understanding System," Eurospeech '95, pp.555-558, 1995.

[9] 김승렬, "국어 어순 연구", 한신문화사, pp.31-42, 1990.

[10] 김영택, "자연언어 처리", 교학사, pp.330-340, 1994.

[11] 왕지현, "구문 정보를 이용한 개념기반의 한국어 대화체 분석기", 충북대학교 석사학위 논문, pp.34-36, 1998

[12] Levin, A and Tomita, M. "An Efficient Word-Skipping Parsing Algorithm for Context-Free Grammars," 3rd International Workshop on Parsing Technologies(IWPT93) Belguim, 1993.

[13] Woszczyna, M. et al., "Recent advances in JANUS : A speech translation system," Eurospeech '93, pp.1295-1298, 1993.

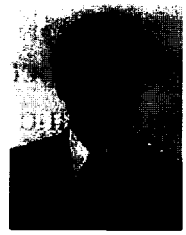
노 서 영

e-mail : rsyoung@dcenlp.chungbuk.ac.kr

1998년 2월 충북대학교 컴퓨터공학과(학사)

1998년 3월~현재 충북대학교 컴퓨터공학과 석사과정

관심분야 : 자연언어처리, 인공지능



정 천 영

e-mail : cyjung@mail.kumi.ac.kr

1986년 충남대학교 계산통계학과(학사)

1992년 충남대학교 전산학과(석사)

1996년 충북대학교 컴퓨터공학과(박사수료)

1986년~1997년 한국에너지연구소 연구원

1997년~현재 구미1대학 전자계산과 전임강사

관심분야 : 기계번역, 자연언어처리, 정보검색

서 영 훈

e-mail : yhseo@cbucc.chungbuk.ac.kr

1983년 서울대학교 컴퓨터공학과(학사)

1985년 서울대학교 컴퓨터공학과(석사)

1991년 서울대학교 컴퓨터공학과(박사)

1988년~현재 충북대학교 컴퓨터공학과 교수

1994년~1995년 미국 Carnegie-Mellon 대학 기계번역 센타 객원교수

관심분야 : 자연언어처리, 음성언어처리, 기계번역