

## 프로그램 재사용을 통한 한/영 기계번역시스템의 구현에 관한 연구

김 형근, 양 기철\*, 최 기선  
한국과학기술원 전산학과  
\*목포대학교 전산학과

### On Implementation of Korean-English Machine Translation System through Program Reuse

Kim Hiongun, Yang Gi-Chul\*, Choi Key-Sun  
KAIST  
\*Mokpo National Univ.

#### 요약

본 논문에서는 범용 영어 생성기인 펜맨을 이용한 한/영 번역기에 대해 소개한다. 펜맨은 영어문장 생성기로서 그 입력 언어는 문장생성을 위해 특별히 고안된 문장계획언어이다. 문장계획언어는 문장을 생성하기 위한 다양한 구문적, 의미적인 사양들을 포함하고 있다.

본 연구에서 우리는 한국어 분석을 위해서 의존문법에 기반한 한국어 분석기를 사용하고, 영어의 생성을 위해서 펜맨을 사용한다. 그리고 한국어 해석의 결과인 문장의 의존구조를 펜맨의 입력인 문장계획언어로 변환하는 모듈을 작성하였다.

#### Abstract

In this article we present a rapid development of a Korean to English translation system, by the help of general English generator, PENMAN. PENMAN is an English sentence generation system, of which input language is a language specially devised for sentence generation, named Sentence Planning Language(SPL). The language SPL has various features that are necessary for generating sentences, covering

both syntactic and semantic features.

In this development we integrated a Korean language parser based on dependency grammar and the English sentence generator PENMAN, bridging two systems through a converting module, which converts dependency structures produced by Korean parser into SPL for PENMAN.

## I. 서론

이 논문은 현재 한국과학기술원에서 진행중인 한국어/영어 기계번역 프로젝트인 MATES K/E에서 실험적으로 쓰고 있는 방법론에 대해 기술한다. 일반적으로 번역이라 함은 원시언어의 이해와 목적언어의 생성이라는 두과정이 합해진 종합적인 작업이다. 어느정도의 이해를 할 것인지, 어느 정도에서 생성을 시작할 것인지에 따라 여러가지 다른 접근방법들이 제시될 수 있다.

이해의 정도를 극도로 높여서, 임의의 언어에 대해서 생성이 가능하도록 하는 것이 중간언어사용방식의 번역이고, 이해의 정도가 극히 낮게 해서 원시언어의 패턴으로 부터 목적언어의 패턴으로 직접변환을 시도하는 것이 직접변환방식이다. 그 중간정도에 있는 것이 변환방식인데, 이것은 원시언어에 대한 구문적, 의미적인 이해를 하기는 하지만, 그것이 임의의 언어로 변환될 수 있을 정도로 일반성은 가지지 못하고, 다만 어떤 두 언어쌍에 대한 변환사전에 의존적인 경우이다.

이 프로젝트에서는 기본적으로 변환방식을 취한 기계번역을 시도하였다. 변환방식의 번역기를 크게 세 부분으로 나누자면, 원시언어로 된 문장의 분석부분과 원시언어에 대한 분석의 결과를 목적언어에 적합한 구조로 바꾸는 변환부분과, 최종적으로 변환된 구조로부터 목적언어의 문장을 생성해내는 3단계를 거치게 된다.

한국어는 영어에 비해 상대적으로 분석이 무척 어렵기 때문에 훨씬더 많은 노력이 한국어 분석쪽에 투자되어야 한다. 한국어 문장에 대한 분석쪽에 많은 노력을 들일 수 밖에 없어서, 상대적으로 영어의 생성쪽에는 별다른 연구결과가 없기 때문에 한/영 번역기를 만듦에 있어서 영어의 생성기쪽은 이미 널리 알려진 생성기를 사용하게 되었다. 이미 만들어진 생성기를 사용하는 것은 짧은 개발기간으로 인한 임시방편이라기보다는 영어 생성기에

대한 구체적인 개발에 앞서서 잘 개발된 시스템의 조사의 성격을 띠기도 한다.

## II. 펜맨이란 무엇인가.

이 프로젝트의 초기시도로 영어문장의 생성부분을 이미 영어문장 생성 시스템으로서 널리 알려진 펜맨(PENMAN)을 사용한다. 펜맨은 미국 남캘리포니아대학의 정보과학 연구소(ISI: Information Sciences Institute)에서 개발한 영어문장 생성 시스템이다. [PENMAN89] [MANN83]

### 1. 펜맨은 범용 영어문장 생성기이다.

자연언어 생성이 필요한 시스템은 상당히 다양한 범위에 걸쳐 있는데, 데이터베이스 시스템이나 전문가시스템처럼 내부적인 자료들을 자연언어를 통해서 보여준다는 단순한 입출력의 필요만을 요구하는 경우도 있고, 인공지능적인 언어연구나 이론언어학과 같이 매우 미세한수준까지 요구하는 경우도 있다. 펜맨은 범용 시스템이기 때문에 사용자가 원하는 수준까지의 문법적, 어휘적 선택을 가능하게 한다.

### 2. 펜맨은 기능조직문법에 기반을 두고 있다.

기능조직문법(Systemic-Functional Grammar)이란 언어를 일련의 규칙들로 설명하기보다는 언어를 문맥속에 사용된 자원이라고 보고 모든 언어적인 현상들을 그것들이 그 문맥내에서 어떻게 구조적으로 역할을 하는지에 대한 '기능'으로 판단한다. 따라서 기능조직문법은 어휘론, 형태론, 음성론, 구문론, 의미론, 화용론까지를 광범위하게 포괄하는 문법이 된다. 문맥상에서의 기능에 중점을 두는 것은, 한글의 특성상 문장을 이루는 각 요소들이 그 위치보다는 형태에 의해 역할이 결정되고, 어순이 비교적 자유롭기 때문에 의존문법에 의한 구문분석을 하려고 하는 시도와 일치하는 면이 있다.

펜맨의 이러한 문법체계는 펜맨의 일부인 니겔문법(Nigel Grammar)으로 잘 정리되어 있다. 기능조직문법에 대해 간단히 설명하자면, 기능조직문법은 Michael Halliday등에 의해 개발된 기능문법의 일종으로, 문법적인 형식을 조직(system)이라고 불리우는 문법적 선택점(point of grammatical choice)들로 이루어진 네트워크로 설명한다. 하나의 조직

에는 여러가지의 문법 자질 (grammatical features)들이 있다. 따라서 어떤 언급을 실현하는 것은 (realization statements) 조직들로 이루어진 네트워크를 따라다니면서 각조직에서 자질들을 선택 하고, 그 결과로 나타나는 특정한 경로를 선형적으로 펼쳐서 문장으로 만드는 행위인 것이다.

### III. 왜 펜맨을 사용하는가.

본연구에서는 펜맨을 한/영 기계번역 시스템의 일부분으로 끌어들이어서 사용하고 있는데, 그것은 단순히 "성급하게 완성으로 치닫는 것"이상의 이유가 있다. 그 이유를 나열하자면 다음과 같다.

#### 1. 시간이 없다.

물론 사실이다. 프로젝트를 훌륭하게 진행하는 데에는 재빠른 시제품의 제작은 필수적인 요건이다.

#### 2. 펜맨은 범용이며 완성도가 높다.

이미 만들어진 영어 생성기가 없거나 있어도 우리가 쓸 수 없는 상태에 있다면 아마 우리 역시 다른 선택의 여지가 없이 영어 생성기를 만들어야만 했을 것이다. 하지만 펜맨은 널리 알려진 영어 생성기로서, 데이터베이스의 인터페이스에서부터 언어학적인 연구에 까지 광범위하고 다양하게 응용과 사용이 가능하며, 그것 자체로 훌륭한 시스템이 될 정도로 완성도가 높아서 상당한 기간동안은 우리의 영어생성에 관한 연구가 그것에 근접할 만큼 충분하지 않을 것이다.

#### 3. 펜맨의 좋은 본보기이다.

펜맨은 그 설계당시부터 범용성을 고려했으며 여러 문장이 들어 있는 문단을 일관성있게 생성해주시기도 하는 상당히 훌륭한 생성기이므로 그것을 이용해서 영어를 생성하는 경험은 우리자신의 영어생성기의 고안이나 제작에 충분히 좋은 모범이 될 것이다.

#### 4. 독립적인 모듈의 연구로 다국어환경에 더 쉽게 접근할 수 있다.

현재는 구체적인 계획은 없지만, 장래를 생각해보면 분명히 번역에 있어서 다국어의 수용이라는 요구가 생겨날 것임이 분명한데, 그때를 생각해 보면 이와같은 독립적인 모듈을 참고함으로써, 독립적인 한국어 생성기, 독립적인 한국어 분석기등의 시스템의 개발에 도움을 주며 그렇게 만들어진 시스템이 제목표화에 효율적이다.

#### IV. 펜맨의 내부구조

구체적으로 펜맨의 구조에 대해 설명하기에 앞서 생각해 봐야 하는 것은, 일반적으로 문장생성기가 문장을 생성할 때 어떤 일을 하는 지를 살펴보는 것이다. 먼저 펜맨으로 생성한 두 문단의 글을 예로 보자.

The system is faulty, if all of the expected values of its input terminals equal their actual values and the expected value of one of its input terminals does not equal its actual value.

Knox, which is C4, is en route in order to rendezvous with Task Group 70.1. It will arrive 4/24. It will perform excersises for four days.

위와같이 문장이나 문단으로 표현되어야 할 정보가 주어지면 그것을 텍스트로 변환하는 시스템은 다음과 같은 일을 해야 한다. 큰눈으로 본 경우로부터 점차 눈을 좁혀가면서 살펴보면,

- ㄱ. 여러문장이 들어 있는 문단의 계획
- ㄴ. 문장의 길이와 전체적인 구조의 계획
- ㄷ. 명사그룹의 계획
- ㄹ. 문법적인 영어문장의 생성
- ㅁ. 적당한 단어나 기호의 선택

이제 다시 작은 눈으로부터 크게뜨면서 자세히 살펴보자.

1. 문장을 생성하기 위한 가장 기본적인 정보는 어디에서 오는가?

펜맨으로 문장을 생성하기 위해 필요한 정보는 사용자의 데이터, 상위모델(Upper Model), 분야모델(Domain Model), 어휘집(Lexicon), 문장계획언어(SPL: Sentence Plan Language), 니겔문법(Nigel Grammar) 등이 있다.

상위모델 (the Upper Model)이란 펜맨에서 기본적으로 주어지는 세상의 모델로서 펜맨의 우주관(Ontology)라고 할 수 있는 것으로, 세상을 아주 일반적인 객체들로 계층적인 구분을 해서 제공하는 것이다. 펜맨의

상위모델에는 약 200개의 개념(concept)들이 계층적으로 구분되어 있는데 그 최상위에는 사물(THING)이 있고 사물(THING)은 다시 객체(OBJECT), 양(QUALITY), 과정(PROCESS)으로 분류되어 있다. 과정(PROCESS)은 다시 정신적과정(MENTAL-PROCESS), 발화과정(VERBAL-PROCESS), 물질적과정(MATERIAL-PROCESS), 상관과정(RELATIONAL-PROCESS)로 나누어진다. 이렇게 미리 세상을 보는 관점을 나누었는데, 계층적으로 정의되어 있으므로 당연히 상위 개념이 갖는 모든 양태(aspect)나 역할(role)들은 아래계층으로 상속된다. 양태(aspect)라면 어떤 관계(relation)을 이루는 정의구역(domain)과 치역(range), 물질적과정(MATERIAL-PROCESS)에서의 행위자(agent)가 되는 행위자(actor), "이다"(IS-A)관계나 "일종이다"(A-KIND-OF)관계로 대변될 수 있는계급설명자(class-ascription), 어떤 대상의 특성을 설명하는 특성설명자(property ascription) 등이 있다.

분야모델(Domain Model)이란 생성될 문장에 특별하게 필요한 세상을 모델화하여 상위모델에서와 같이 계층적으로 정의한 것을 말하는데, 분야모델에서의 가장 높은 개념들은 반드시 상위모델의 어느것과 연관지어져야 한다. 왜냐 하면 문장생성시에 필요한 수많은 특성들이(한두단어로 된 한 문장을 표현하기 위해서라도, 수십에서 수백가지의 특성들을 알아야 한다.) 미리 정의되어 있어서 그것들을 상속받을 필요가 있기 때문이다. 이런 분야모델을 만들기 위해서는 우선 우리가 관심을 쏟고 있는 분야에서의 대상들, 행동들, 사건들, 관계들 등등을 모두 나열하고, 그것들을 적당히 묶어서 그룹으로 나누고, 그리고 나서 상위모델에서의 구분에 맞춰서 상위 그룹의 적당한 위치에 종속시켜야 한다. 상위모델의 개념과 분야모델에서의 어느 특정한 개념사이의 연결은 UpperMost라는 프로그램에 의해 종속적으로 연결하는 것이 가능하다. 다음은 어느 특정한 분야모델에서 GIVE라는 개념을 정의한 것이다.

```
(defconcept GIVE
  :is (:and Dispositive-Material-Action :Primitive))
```

이것은 물질적 이동행위(Dispositive-Material-Action)의 일종이라는 뜻으로, 물질적 이동행위는 이미 상위모델에 정의되어 있는 것이다.

어휘집(Lexicon)이란 위에서처럼 분야를 정리하고 구분해서 만든 분야모델에서의 개념들에 이름을 붙인 것들을 모아둔 것으로서, 실제로 1차원적인 문장을 생성할 때 최종적으로 필요한 자료이다. 어휘의 문법적인 변화

형들도 여기에 저장되어야 한다. 펜맨에서는 LapItUp이라는 도구가 준비되어 있어서 분야모델의 개념과 그에 상응하는 어휘의 연결을 쉽게 할 수 있도록 도와준다.

## 2. 문장계획언어

문장계획언어(SPL: Sentence Plan Language)는 펜맨에서 생성해야 하는 어떤 문장의 세부적인 사항을 결정할 수 있는 언어로서 매우 정교하기 때문에 이 언어를 쓰면 상당히 미묘한 표현까지 가능하다. 다음은 문장계획언어로 "Knox sails."라는 문장을 표현한 것이다.

```
(S1 / SAIL
  :actor (KNOX11 / SHIP
    :name KNOX)
  :tense PRESENT
  :speechact ASSERTION)
```

이것은 행위자가 KNOX이며 현재에 일어나고 있으며, 그 사실에 대해 단순히 긍정을 하는 SAIL이라는 사건에 대해 이야기 하고 있음을 알 수 있다.

두단어로 된 간단한 문장을 만들기 위한 문장계획언어 표현으로 부터 우리는 많은 것을 살펴볼 수 있는데 그것은 문장계획언어는 네트워크구조를 가지고 있으며, 네트워크의 노드들은 (여기서는 S1, KNOX11등이 그것이다) 다양한 문법적인 자질들을 가지고 있고, 그 자질들에 들어갈 수 있는 값들은 이미 상위모델에서 정의 된것을 쓸 수도 있고, 사용자가 자신만의 분야 모델에서 정의한 것일 수도 있고, 혹은 다른 노드가 그 값으로 올수도 있다(여기서는 KNOX11이 그예.)

그러나 잘 생각해보면 "Knox sails"라는 문장을 만들기위해서는 위에서 나타난 것 이외의 수많은 다른 정보들이 필요하다는 것을 알수 있는데 예를 들어 "Sail to Knox", "Sail from Knox", "Does Knox sail?", "Sail, Knox", "Sailing Knox", "Knox sailed", "Knox which sailed"등등의 문장들과 구분되는 특징들을 담고 있어야 한다. 이런 구체적이고 미묘한 것까지 표현할 수 있도록 문장계획언어는 상당히 많은 구문적, 의미적인 면모를 갖추고 있다.

### 3. 니겔 문법

앞에서도 이미 언급한 바와 같이 펜맨에서는 언어라는 것을 문맥에 따라 기능적인 필요에 의해 발화된 언어적인 자원으로 생각하는 기능조직문법(Systemic-Functional Grammar)을 사용한다. 기능문법에서는 어휘론, 형태론, 구문론, 의미론, 화용론까지를 모두 폭넓게 문법이라는 것으로 묶어서 비교적 단순한 '상황내에서의 기능'으로 설명하는데, 그런 관점에서 본다면 펜맨시스템은 크게 문법을 다루는 문법기인 니겔과 그이외의 모든 환경적인 자원으로 양분될 수 있다. 니겔은 앞에서 설명한 문장계획 언어로 표현된 네트워크적인 문장에 대한 계획을 입력으로 받아서 그것을 문장발현(realization statements)이라는 과정을 통해, 다시말해 문법네트워크를 따라다니면서 여러가지 판단을 하여, 일차원적인 문장으로 만들어 내는 역할을 한다.

니겔에서 어떤 문장이(혹은 문단)이 생성되기 위해서는 여러개의 서로다른 조직들이 조화를 이루며 협동해야 하는데 그것들에는 이미 언급한 상위모델, 분야모델, 어휘집이외에도 문단계획자, 문장구조계획자, 명사그룹계획자등이 있다.

문단계획자는 수사구조이론(RST: Rhetorical Structure Theory)라는 어절들 사이의 관계를 따지는 이론을 통해서, 문장계획언어로 된 입력을 적당히 가공하여 어절들의 길이, 순서, 관심의 촛점등을 조절하는 역할을 한다. 비슷하게 문장구조계획자도 입력의 특성등을 고려하여 입력을 어떤 트리형태로 표현한다음 그 트리상의 구조적인 변화를 기획함으로써 문장의 구조를 결정한다음 그 트리를 따라 다니면서 촛점을 제어하고 문장의 크기한계등을 결정하는 일을 한다.

여러문장이 가능한 문단단위의 생성을 지원함에 있어서 중요한 부분중에 하나가 바로 명사그룹의 처리인데, 명사그룹의 처리에서는 관사(determiner), 대명사(pronouns), 명사화(nominalization), 양화사(quantifiers), 복합명사(compound nominals)등을 처리한다.

### 4. 사용자의 역할

위에서 펜맨의 다양한 면모를 살펴보았는데 과연 사용자가 자신이 원하는 특정의 문장만을 만들어내기 위해서는 어떤 일들을 해주어야 하는가.

일단 사용자는 자신이 생성하고자하는 문장들이 속해있는 분야를 조사할 필요가 있다. 대상, 객체에는 어떤 것이 있고 어떤 사건이 일어나며 어떤 행동이 가능하고 그들간의 관계는 어떠한가등에 대한 나열이 필요하다. 데이터베이스에 자연언어 인터페이스를 지원하고자 하는 사람이라면 이미 그런 것들이 준비되어 있을 것이다.

그 다음에는, 이렇게 조사된 분야에 관한 정리를 일목요연하게 분야모델이라는 것과 그에 상응하는 어휘집이라는 것으로 조직화함으로써 기본적인 준비가 끝난다. 이때 분야모델에서의 최상층에 있는 개념들은 상위모델의 개념들과 어떤식으로든 연결되어야 한다.

그다음에는 실제로 문장의 생성에 필요한 문장계획언어의 구사가 가능한데, 사용자의 요구수준에 따라, 만약 사용자의 요구가 가장 기본적인 수준에 머문다면 미리 준비된 문장계획언어 생성기(SPL-Constructor)를 사용할 수도 있고, 사용자가 만약 충분히 자세한 수준까지 문장을 계획하고 싶다면 사용자가 직접 문장계획언어를 써서 문장을 만들 수도 있고, 만약 사용자가 언어학적으로 깊이 관심이 있다면 니겔문법이나 어휘선택규칙같은 것까지 미세하게 다룰수도 있다.

이로서 일반적으로 생성기라 함은 어떤 입력구조를 필요로하며, 펜맨이라는 시스템은 구체적으로 어떻게 그 필요를 만족시키고 있으며, 펜맨을 사용하여 문장을 생성하기 위해서는 어떤 작업들이 필요한지를 살펴보았다. 이제는 다시 한/영번역기로 돌아가서, 한영번역기를 전체적으로 보았을 때, 이 펜맨이 어떻게 한/영번역기에 수용되어 쓰이는지를 살펴보자.

## V. 펜맨의 입력 생성

한/영 번역시스템의 분석기는 한국과학기술원에서 개발한 한국어 파서를 사용하는데, 한국어는 비교적 어순이 자유롭고 문장의 필수성분이 빈번히 생략되며, 문장을 이루는 요소들이 그 위치에 의해서라기보다는 형태에 의해 역할이 결정된다. 이러한 특성 때문에 한국어의 구문 분석에는 의존문법이 바람직하다. [서광준93]

본연구에서는 의존문법에 기반하여 문장 내의 어절 사이의 구문적 관계에 의해 문장을 분석하는 한국어 파서를 이용하는데 파싱 결과로 나오는 출력은 의존구조이다. 파싱의 성공률을 높이기 위해 퍼지개념이 파서에 응용되었고[서광준93] 이는 자연언어 처리시스템의 견고성을 높여(MANF92)

보다 실용적인 시스템이 될수 있도록 하였다. 그렇게 해서 생성되는 의존 구조를 펜맨의 입력형태인 문장생성언어로 변환하는 작업이 이 논문에서 하는 가장 큰 일이다. 구체적으로 설명하기 위해 다음의 예를 보자. 다음은 "적어도 세척의 배가 월요일에 목포항으로 향해한다"라는 문장에 대한 한국어 구문해석의 결과와 문장계획언어의 구조이다.

의존구조 :

[[[[[적어도 ADV] 세척의 MOD] 배가 AGT] [월요일에 ALT]  
[목표항으로 TAR] 향해한다 SNT]

문장계획언어구조 : .

(S1 / SAIL  
:actor (A1 / SHIP  
:at-least 3)  
:destination (D1 / PORT  
:name Mokpo-Hang)  
:temporal-nonordering (T1 / DAY  
name MONDAY)  
:tense PRESENT  
:speechact ASSERTION)

위에서 보는 의존구조나 문장계획언어 구조는 표현형태는 다르나 같은 내용을 말하고 있다. 의존 구조는 직접 영어문장생성기인 펜맨의 입력으로 쓰일 수 없기 때문에 펜맨이 이해할 수 있는 문장계획언어로 변환이 필요하다. 위에서도 언급한 바와 같이 문장계획언어의 표현에서 문장 생성에 사용되는 개념들이 분야모델에 정의되어 있지 않으면 사용되지 못한다. 따라서 펜맨을 사용하기전에 이미 분야모델과 어휘집을 정의하고 문장계획언어로 입력을 만들어야 한다.

문장계획언어와 의존구조 사이에는 구조상의 차이뿐만 아니라 의존구조에서 사용되는 격의 이름과 문장계획언어에서 사용하는 자질의 이름 (Keyword)이 다르고 서로 1:1 대응이 되지 않는 어려움이 있다. 이 문제는 현재의 구문해석기를 개정하여 직접 문장계획언어의 입력형태로 출력하게 할 수도 있고, 키워드의 대응표를 만들어 둘사이를 연결할 수도 있다. 본연구에서는 대응표를 만드는 방법을 사용한다. 이는 펜맨이나 한국어 구문해석기의 독자성을 계속 유지하면서 문제를 해결하는 것이며, 특히 키워드를 마음대로 새로 정의하여 쓸 수 있는 펜맨의 융통성에 신축적으로

대응할 수 있기 때문이다. 키워드의 대응표에는 구체적으로 의존구조의 AGT를 문장계획언어의 actor로 대치하는 등의 정보가 들어 있다.

또한 의존구조에 나타난 단어들이 분야모델의 개념으로 정의되어 있는지를 확인하고 기타정보를 이용하기 위해 개념표(Concept Table)이 필요하다. 개념표는 개념구조와 문장계획언어 사이의 변환기에서 펜맨의 상위모델과, 분야모델, 어휘집과 같은 역할을 하기 위해 작성된다. 개념표는 어휘, 정의된 개념, 세상에 관한정보의 집합이다.

위와같은 표가 만들어지면 남는 것은 구조 변환인데, 이를 위해 먼저 의존구조를 역순으로 변환시킨다. 이는 한국어의 특성상 본동사가 문장의 끝에 나오는 성질과 펜맨의 입력은 객체가 아닌 행위나 사건등과 같은 과정을 주축으로 구성되는 특징이 있기 때문이다. 다음은 역순으로 변환된 의존구조이다.

(SNT 항해한다 (목포항으로 TAR) (월요일에 ALT)  
((적어도 ADV) 세척의 MOD) 배가 AGT)))

이렇게 변환된 의존구조의 첫번째 원소로부터 "항해하다"라는 개념어인 SAIL을 찾아 "(S1 / SAIL"이라는 문장계획언어문장의 일부를 만들 수 있다.

다음단계로 의존구조 각각의 원소들은 개념표를 근거로 문장계획언어로 변환될 수 있다. 예를 들어 "(목포항으로 TAR)"는 문장계획언어의 ":destination (P1 / PORT :name Mokpo-Hang)"으로 변환된다.

이렇게 한 원소가 하나의 간단한 원소로만 이루어진 경우는 처리하기가 쉬우나, "((적어도 ADV) 세척의 MOD) 배가 AGT)"와 같이 여러 층으로 이루어진 구조일 경우에는 재귀적(recursive)으로 변환을 시도할 수 있다. 먼저 "(A1 / SHIP (number 3 (at-least ADV)))"로 변환한 다음 다음단계에서 (A1 / SHIP :at-least 3))로 바꿀 수 있다.

## VI. 결론

본 연구에서는 구문정보 뿐만아니라 의미정보까지도 활용하는 한/영 기계번역 시스템을 시도하였다. 이를 위하여 한국어를 분석하는 데 의존문법

에 기반한 한국어 구문분석기를 사용하고 그 결과를 의미정보를 첨가하여 영어 생성기인 펜맨의 입력형식으로 바꿔 영어문장을 생성하도록 하였다. 구문분석의 결과인 의존구조는 개념표에 있는 의미정보를 써서 펜맨의 입력형식인 문장계획언어로 변환된다.

이는 독립적으로 개발된 두 시스템을 상당히 거칠게 연결하는 결과를 가져왔지만, 이로 인해 앞으로 더 연구해야할 과제로 몇가지가 명확히 드러났는데 그것은 다음과 같다.

1. 한국어 구문해석결과인 의존구조에 사용되는 심층격과 문장계획언어에 사용되는 의미자질이름(Keyword)들 사이의 더욱 세분화된 관계를 정립한다.
2. 개념표에 대한 체계적인 정의가 필요하다.
3. 일반적인 자연언어 문장생성기가 갖추어야 할 성질들을 고찰한다.

이것들이 해결되면 보다 양질의 의미정보를 이용하는 한/영 기계번역기를 구현할 수 있을 것이다.

## 참고문헌

[서광준93] 서광준, "어절 사이의 의존관계를 이용한 한국어 구문 분석기", 한국과학기술원, 석사학위 논문, 1993.

[PENMAN89] The Penman Natural language Group, *The Penman User Guide*, The Penman ReferenceManual, 1989.

[MANN83] Mann. w. c, *An overview of the PENMAN text generation system*, Proceedings of The AAI, 1983.

[MANF92] Manfred Stede, *The Search for Robustness in Natural Language Understanding*, AI review 6 P383-414, 1992.