대화체 문장 번역을 위한 토큰기반 변환중심 한일 기계번역

(A Token Based Transfer Driven Koran -Japanese Machine Translation for Translating the Spoken Sentences)†

양 승 원*
(Seung-Weon Yang)

요 약 본 논문에서는 음성언어 자동 통역시스템의 일부 모듈로 구현한 한일 기계번역 시스템을 소개하였다. 이 번역시스템은 예제중심 기계번역(EBMT)에 기초를 둔 변환중심 기계번역(TDMT) 방법을 기반으로 구현하였다. 본 시스템에서는 토큰(TOKEN)이라는 새로운 번역단위를 정의하여 사용하였다. 토큰단위의 번역방법을 사용함으로써 한국어 문장의 매우 비 정형적인 점을 해결하고 번역의 질을 높일 수 있다. 본 시스템의 구문분석 단계에서는 대역어를 선정하기에 적합한 정도까지의 의존트리를 생성하는 간이파싱만을 함으로써 필요없는 노력을 경감시켰다. 대역어 사전은 한국전자통신 연구원이 수집한 음성 데이터베이스로부터 추출한 말뭉치를 사용해 구성하였다. 구현한 시스템은 여행 계획영역에서 수집된 600 발화 안의 문장을 대상으로 시험하였는데 제한된 환경에서 87%, 아무런 제약이 없는 환경에서는 71%의 성공률을 보였다.

Abstract This paper introduce a Koran-Japanese machine translation system which is a module in the spoken language interpreting system. It is implemented based on the TDMT(Transfre Driven Machine Translation). We define a new unit of translation so called TOKEN. The TOKEN-based translation method resolves nonstructural feature in Korean sentences and increases the quaity of translating results. In our system, we get rid of useless effort for traditional parsing by performing semi-parsing. The semi-parser makes the dependency tree which has minimum information needed generating module. We constructed the generation dictionaries by using the corpus obtained from ETRI spoken language database. Our system was tested with 600 utterances which is collected from travel planning domain. The success-ratio of our system is 87% on restricted testing environment and 71% on unrestricted testing environment.

1. 서 론

컴퓨터를 이용하여 번역을 수행하는 기법으로는 규칙기반 기계번역(RBMT: Rule Based Machine Translation)과 변환중심 기계번역(TDMT: Transfer Driven Machine Translation)을 들 수 있다.

규칙기반 기계번역은 일반적으로 사용하는 방법으로 번역에 필요한 지식을 규칙으로 표현하며 번역은 이렇게 표현된 규칙들을 반복적으로 적용하면서 진행된다. 이 방법의 가장 큰 문제점은 확장이 어렵다는 것이다. 따라서, 규칙기반 기계번역 방법으로 실용적인 기계번역시스템을 만들기는 매우 어려운 일이다. 이러한 확장성 문제는 변환중심 기계번역 방법에서 상당부분 해소 될 수 있다. 이 개념은 기존에 성공적으로 번역된 예문들을 저장해 둔 데이터베이스와 시소러스(thesaurus)를 이용하여 가장 확률

† 이 논문은 한국전자통신연구원의 지원으로 연구되었음.
* 우석대학교 정보통신컴퓨터공학부 조교수

이 높은 예문을 번역의 해(goal)로 선택하는 예제기반 기계번역(EBMT: Example Based Machine Translation)에서 비롯되었다[1,2]. 변환중심의 기계번역에서는 많은 예문들을 효율적으로 찾기 위해서는 예문들의 저장방법이 새로워야 하고 이들을 저장하기에 충분한 저장공간이 필요하다. 또한, 보기들이 많아지면 정확한 해를 찾는 데에 많은 시간이 소요된다. 그렇지만 번역의 메카니즘이 명료하기 때문에 실용적인 시스템을 구현하는 데에는 규칙기반 기계번역보다 적합한 점이 많다. 특히, 동일한 어족에 속하며 비슷한 문장구조와 문자수준을 가지고 있는 한국어와 일본어 간의 번역에는 변환중심의 번역 기법이 다른 기법들에 비하여 효과적으로 적용될 수 있다.

한편, 변환중심의 번역에서는 변환모듈이 중심이 되어서 번역이 이루어지므로 원문에 대한 적절한 대역문을 찾아내는 것이 가장 중요하며 기타의 모듈들은 대역어 변환 작업을 돕는 변환 지식들로 제공된다. 따라서, 본 시스템의 구문분석 단계에서는 완전한 분석이 아니라 대역어를 선정하기에 적합한 정도까지의 의존트리를 생성하는 간이파싱만을 수행한다. 또한, 한국어 대화체 문장은 문장을 이루는 문장성분들의 자유도(degree of freedom)가 매우 높기 때문에 구문분석 단계까지의 결과가 많은 모호성과 오분석을 포함하고 있다. 이러한 점은 번역의 질을 현저히 떨어뜨리는 원인이 되므로 본 논문에서는 한국어문장에서 의미상의 내용어와 기능어를 묶은 결합체를 토큰(TOKEN)이라 새롭게 정의하고 이 토큰을 문장의 기본 단위이자 분석 단위로 삼아 변환중심의 번역을 수행하는 번역시스템을 구현하였다. 본 논문에서는 이 번역 방법을 TB-TDMT(Token-Based Transfer Driven Machine Translation)라 부르며 구현한 시스템은 한국전자통신 연구원에서 카네기 멜론 대학등과 공동으로 연구를 진행 중인 CSTAR과제 중 음성언어 번역시스템에서 한국어를 입력으로 받아 일본어 문장을 생성해 내는 한일 번역 모듈로 사용되고 있다. 본 논문의 제 2장에서는 연구에 도입된 개념에 대해 기술하고, 제 3장에서는 토큰 단위의 분석에 대하여 설명하며, 제 4장에서는 시스템의 구현을 그리고 제 5장에서는 결론을 맺는다.

2. 연구에 도입된 개념

2.1 변환중심 기계번역(TDMT)

변환중심 기계번역(TDMT)은 예제중심 기계번역(EBMT)을 골격으로 발표되었다. EBMT는 번역된 예제와 시소러스의 중요성을 강조한 방법이다[3,4,5,6]. 이 번역 방법은 사람들이 번역 작업을 할 때, 사전에 있는 예문을 직접 이용하여 번역을 하는 것과 같은 개념이다. TDMT는 예제의 변환방법에 해석적인 지식을 포함시켜 규칙기반번역의 특성을 결합함으로써 여러 가지 다양한 입력 문장을 번역해 낼 수 있도록 설계되었다. 해석모듈과 변환모듈은 서로 독립적으로 동작하는데 먼저 해석 모듈에서 해석 지식을 적용하고 그 결과를 다시 변환 모듈에 전달한다. 만약 해석 모듈을 적용할 필요가 없을 경우에는 변환 지식만 이용해서 번역한다. TDMT는 변환을 중심으로 많은 다른 종류의 지식을 서로 협조적으로 사용함으로써 번역이 이루어진다. 변환 지식은 여러 종류의 양국어 정보로 구성된다. 또한, 변환부가 중심이 되어 처리되기 때문에 형태소 분석부, 구문분석부, 생성부, 문맥처리부 등은 변환부가 정확한 번역 결과를 생성할 수 있도록 도와주는 역할을 한다. TDMT에서 중요한 점은 입력된 문장과 가장 유사한 예문을 준비된 데이터베이스로부터 어떻게 찾아낼 것인가 하는 것이다. 이를 위해서 입력과 보기들 사이의 거리를 계산하여 가장 적절한 보기를 선택한다. 두 단어(입력과 보기)사이의 거리는 시소러스 상에서 의미 속성들의 거리로 정의된다. 변환 지식은 특별한 의미 단위를 원어 표현(Source Expression)과 목적어 표현(Target Expression)들 사이의 대응관계를 표현한 것이다. 이것은 보기 기반 골격에 따라서 식(1)과 같이 표현된다.

$$\begin{aligned} SE \rightarrow\ & TE_1(E_{11}, E_{12}, \ldots) \\ & \cdot \\ & \cdot \\ & \cdot \\ & TE_n(E_{n1}, E_{n2}, \ldots) \end{aligned} \qquad (1)$$

각 TE는 조건으로서 여러 보기를 가지고 있다. Eij는 TEi의 j번째의 보기를 의미한다. 입력이 SE일 때에 그 입력과 보기들의 거리를 계산하여 가장 적절한 TE가 선택된다. 입력(I)과 보기(Eij)의 의미적인 거리 즉, d(I,Eij)는 식 (2)와 같이 계산된다.

$$\begin{aligned} d(I,E_{ij}) &= d((I_i,\ldots I_n),(E_{ij1},\ldots E_{ijn})) \\ &= \sum_{k=1}^{t} d(I_k,E_{ijk}) * w_k \end{aligned} \qquad (2)$$

여기에서 $0 \le d(I,Eij) \le 1$이고 거리를 계산하는 방법으로는 사례기반 추론(case-based reasoning)에서 이용하는 Most Specific Common Abstraction(MSCA)[7]을 사용한다. MSCA는 두 패턴이 의미적으로 유사하면 유사할 수록 작아진다. wk는 식 (3)과 같이 정의하며, 각 Ik에 대한 TE의 분포를 나타낸다[8].

$$w_k = \sqrt{f^2} \qquad (3)$$

식 (3)에서 f는 말뭉치로부터 얻어 낸 것으로 Ik가 Eijk로 번역된 빈도수이다. wk를 계산하기 위해서는 많은 계산이 필요하지만 실제는 예제 데이터베이스의 번역 패턴의 빈도수에 의존하므로 데이터 베이스 구축 시 미리 계산해 둘 수 있다. 따라서, 실행 시에는 상수를 참조하는 정도의 시간 밖에 요구되지 않는다. 입력 I로부터 모든 보기들의 거리를 먼저 계산한다. 그리고 나서 거리가 가장 가까운 보기를 선택하여 TE를 선택하는데 Eij 가 I 에 가장 가까울 때 TEi가 가장 가능성 있는 TE로 선택된다. TDMT에서는 보기가 많으면 많을수록 더 정확한 TE를 결정할 수 있다. 왜냐하면 TE를 결정하는 조건이 더 세부적이기 때문이다. 만약 보기가 하나만 존재하거나 I에 충분히 가까운 Eij를 선택할 수 없을 때에는 변환 지식의 적용을 거절할 수도 있다.

3. 토큰 기반 번역 모델

한국어는 기능어가 별도의 성분으로서 내용어의 뒤에 나타나는데, 이 같은 유형의 언어를 언어학에서는 교착어로 분류한다. 내용어는 명사나 동사의 어간처럼 그 자체가 어휘요소로서 의미적 내포를 갖는 성분이고 기능어는 조사나 어미처럼 자신은 의미 내포를 가지지 않는 대신 내용어가 문장에서 차지하는 문법적인 역할을 제시하는 성분이다. 예를 들면, "철수가 밥을 먹었다"라는 문장에서 '철수'와 '떡' 그리고 '먹(다)'은 자신의 의미를 분명하게 갖는 내용어이고 '가'와 '를'은 각각 '철수'와 '밥'을 문장 상에서 주어와 목적어로 역할을 결정해주는 기능어이다. 이처럼 기능어는 실체에 대응하는 성분이라기 보다는 내용어의 문법적 역할을 보완하는 성분에 불과한 것이다. 그럼에도 불구하고 기존의 많은 연구에서는 기능어를 내용어와 대등한 문법 성분으로 취급하고 분석을 수행함으로써 오분석과 비효율의 원인이 되어 왔다. 대역어의 변환을 중심으로 하는 본 논문의 번역 시스템에서는 이러한 난점을 해소하기 위해서 어절과 문장의 중간 개념인 토큰을 다음과 같이 정의하고 이를 분석과 대역의 기본 단위로 삼는다.

「정의 1」 토큰(TOKEN)

가. 하나의 의미상의 내용어와 그에 수반된 의미상의 기능어로 구성되는 문장의 구성단위.

나. 토큰의 범주는 명사구토큰(NP), 동사구토큰(VP), 부사구토큰(AP), 관형사구토큰(DP) 등의 기본토큰과 대화체 문장에 적응하기 위하여 간투사토큰(IG), 구분자토큰(SP) 등의 변형토큰 을 둔다.

여기에서 정의된 토큰은 전통적인 언어학적인 문법체계에서 말하는 어절과는 분명한 차이가 있다. 어절은 '명사 + 조사' 또는 '동사/형용사 어간 + 어미'의 형태이지만 토큰은 그 정의에 의미상의 내용어와 의미상의 기능어라는 용어를 사용했으므로 토큰이 분리되는 구획이 어절과는 다르다. 즉, 지역적인 패턴으로 나타나는 각종 숙어적 표현이나 복합명사 등을 하나의 토큰으로 포착할 수 있다. 다음과 같은 문장을 정의에 맞게 토큰 단위로 분리해 보자.

예문 1) 에 다름이아니라요 이번에 영상축전을 하는데 귀빈들의 식사예약을 어 할 수 있는지 알고 싶습니다.

0 <에 IG>
1 <다름이아니라요 AP>
2 <이번에 AP>
3 <영상축전을 NP>
4 <하는데 VP>
5 <귀빈들의 DP>
7 <식사예약을 NP>
8 <어 IG>
9 <할수있는지 VP>
10 <알고싶습니다 VP>

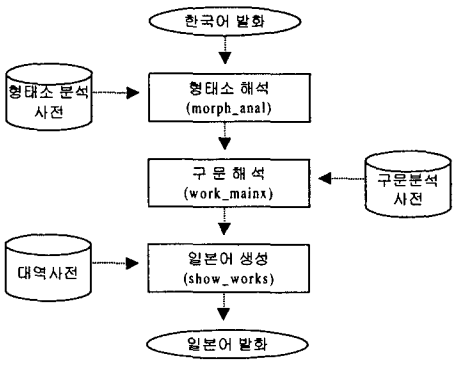
<그림 1> 예문 1에 대한 토큰의 분리

이들 중에서 2,5번 토큰은 일반적인 어절과 일치하지만 1,3,7,9번 토큰은 의미상의 내용어를 중심으로 한 토큰의 특성 때문에 얻어 지는 것들이며 전통적인 형태소 분석으로는 얻을 수 없는 것이다. 즉, 3번 토큰은 원래에는 각각의 의미를 갖는 '영상'과 '축전'의 두 개의 형태소로 되어 있으나 이 둘은 단일 의미를 가지므로 하나의 의미상 내용어로 분리된다. 게다가, 10번의 '알고싶어요'는 일정한 패턴을 갖는 숙어 성분이 하나의 토큰으로 분리된 예이다. 원래의 형태소 분석에 따로 떨어진 상태(알고+싶습니다)로는 본 논문에서 목적언어로 삼고 있는 일본어에서 적절한 대역 해(goal)를 찾을 수 없으나 토큰으로 분리된 상태에서는 '知りたいんです'로 자연스럽게 번역 할 수 있다. 분리된 각 토큰은 내부적으로는 내용어와 기능어의 경계를 명확히 가지고 있어서 기존의 언어학적인 문장소로도 쉽게 환원할 수 있다.

4. 시스템 구현

한일 번역기는 크게 형태소 분석, 구문분석, 일본어

생성으로 구성되어 있고 그 구조는 TDMT의 구조와 유사하며 <그림 2>와 같다. 번역기의 각 모듈은 서로 다른 사전 구조를 사용하고 있다. 형태소 분석기와 구문 분석기는 기존에 독립적으로 개발된 것을 모체로 사용한다. 따라서, 각 모듈에서 사용하는 정보도 또한 서로 다른 정보를 사용하며 각 사전 역시 독립적으로 개발되었다. 각 부분에 대한 설명은 다음의 각 절에서 구체적으로 설명한다.



<그림 2> 시스템 구성도

4.1 형태소 및 구문 분석

본 번역기에서 형태소 분석기는 음절정보를 이용한 형태소 분석 방법[9]을 사용하였다. 이 서브루틴은 한글공학연구소의 홈페이지(http://ham.hansung.ac.kr)에 가면 HAM이라는 이름의 라이브러리(libham.a) 형태로 제공되고 있다. 형태소 분석을 위한 사전 역시 HAM에서 제공하는 내부 코드 형태소로 되어있다.

구문 분석기는 본 번역기에서 완전한 구문분석을 수행하지 않고 다음 단계인 일본어 생성 단계의 입력정보를 제공하는 정도의 구문분석을 수행하는 간이분석을 사용하였다[10]. 이 방법은 한국어 대화체 문장을 분석하기 위한 여러 가지의 경험적 지식(Heuristic information)을 사용한다. 구문 분석을 위해 사용하는 사전은 6개로 나뉘어져 있다. 이들은 각각 부사사전(adv.dic), 명사사전(noun.dic), 어미사전(suf.dic), 관형어사전(det.dic), 조사사전(post.dic), 용언사전(verb.dic)이고 각 사전은 Trie구조를 채택해 구성하였다. 이들 사전은 ASCII 파일이며 <그림 3>과 같이 사전 엔트리의 리스트를 LISP코드 형태로 구성되어 있다.

<그림 3>에서는 보이지 않으나, 실제의 사전의 정보 필드에는 여러 가지의 경험 정보를 포함하고 있다. 이 경험 정보들은 사전의 key값 뒤에 나열되어 있다. 현재는 각 key에 대한 품사정보만이 주로 사용되고 있고 계속 설계되어 추가되고 있으므로 본 논문에서는 생략하기로 한다.

```
(
(가 (능한한빨리 )
                (장 ))
                    ....
            (각 (자 ))
            (갑 (자기 ))
            (같 (이 ))
            (거 (기 )
                (서 )))
)
```

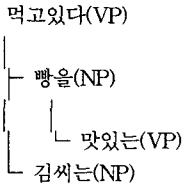
<그림 3> 부사사전의 내용

하나의 문장(예문2)이 본 시스템에서 구문분석 되기까지의 수행되는 과정은 <그림 4>와 같다.

예문 2) 김씨는 맛있는 빵을 먹고 있다.

```
4 5 11
    김씨는,NP ([철수, N] [가, P])
    맛있는,VP ([맛,N] [있,SF] [는,F])
    빵을  ,NP ([빵,N] [을,P])
    먹고있다,VP ([먹,V] [고,F]) ([있,V] [다,F])
```

a) 토큰의 분리



b) 의존트리

<그림 4> 구문 분석 과정

<그림 4>의 a)는 토큰이 분리된 결과를 보여준다. 여기에서 분리된 토큰들 위의 숫자는 각각 토큰의 수(4), 어절 수(5), 형태소 수(11)를 나타낸다. 그리고 비록 표층으로는 토큰의 형태로 나타나지만 내부의 자료구조에는 개개의 어절이나 형태소 정보가 정확히 포함되어 있는 것을 볼 수 있다. b)는 구문 분석의 결과로 생성된 의존트리이다. 그 의존트리의 각 노드는 토큰이고 에지에는 토큰들 사이의 관계를 표현한다. 토큰은 하나의 의미를 가지고 있는 기본적인 단위로서 번역의 기본 단위가 되며 중심어와 그 중심어에 따른 기능어로 구성되어 있다.

4.2 일본어 생성

일본어 생성 모듈은 한국어에 대한 구문 분석 결과를 입력으로 받아서 일본어 문장을 생성한다. TB-TDMT는 2장에서 설명한 TDMT를 기본 번역 방법으로 채택하고 있으며, 그 번역의 기본 단위는 토큰으로 하고있다.

4.2.1 대역 사전의 구성

일본어로 번역하기 위해서 사용되는 대역 사전의 종류는 token_dic, examj_dic, forbid_dic이 있다. 이 대역 사전들은 한국전자통신연구원의 여행영역 말뭉치 중에서 일본어 번역 표현과 1대1로 정렬된 말뭉치(aligned corpus)로부터 생성하였다. 여기서 token_dic은 기본적인 토큰의 대역어를 수록한다. 이 기본 대역 사전은 하나의 토큰에 대응하는 여러 개의 대역어를 가질 수 있으며, 다중 대역어에 대한 모호성은 함수에 의해서 해결된다. <그림 5>는 token_dic의 사전 항목에 관한 구체적인 예이다.

토 큰	번역 정보	대 역 어
알고싶은데	1 2 0 3115	知りたいん 知りたいんですが
알고싶은데요	1 3 0 3116	知りたいんですが 知りたいですが しりたいんですが
알기위해	1 1 0 3117	しるため
알기위해서	1 1 0 3118	しるために
알려드리겠고요	1 1 0 3119	お知らえいたします
알려드리겠습니다	1 2 0 3120	お教えいたします お知らせいたします

<그림 5> 토큰 대역 사전

<그림 5>의 필드 중 가운데의 숫자들은 번역의 질을 높이기 위하여 사용하는 변환외적인 정보들이다. 각각은 처음부터 토큰의 범주, 대역어의 개수, 대응되는 명사가 2개인 경우 스타일, 사전내의 레코드 번호를 나타낸다. 예를 들면, 토큰 '알고싶은데요'는 토큰의 범주가 1로서 VP를 나타내고 대역어의 개수는 3개 그 다음 수자는 명사형이 아니므로 0이며 마지막의 수자는 이 토큰이 저장된 위치가 3116번째의 레코드임을 나타낸다.

examj_dic은 의존트리의 한 에지를 기본 단위로 하는 번역 보기들이다. 즉, 토큰들이 모여지면서 문장을 이루어 가는 과정의 번역 예들을 수록해 둔 사전이다. examj_dic는 <그림 6>과 같이 구성되어 있다. 이 사전 안에서 수자들의 의미는 대역필드의 개수와 레코드 번호이다.

forbid_dic은 음성인식 결과를 번역하는 데에서 오는 불확실함을 해소하는 데 사용하는 사전이다. 음성인식 결과에는 실제 의미전달에는 불필요한 여러 가지의 간투어들이 섞여있다. 이와같은 간투어들을 일일이 번역할 경우에는 의미전달이 부자연스러울 뿐 아니라 엉뚱한 문장으로 번역되는 경우가 자주 발생한다. 본 시스템에서는 이와 같은 문제를 해결하기 위한 가장 간단한 방법으로 의미전달에 큰 영향을 주지않는 간투어는 번역문에서 제거해 버린다. 이를 위해서 간투어에 대한 정보를 저장해두는 사전이 forbid_dic이다.

의존트리의 에지	번역 정보	대역어
가격을싸게하고싶은데	1 69	値段を安くしたいと思うんです
가고싶은데요	1 70	行きたいんです
가는게있습니까	1 71	行くことがありますか
가려구하는데요	1 72	行きたいと思っておりますけど
가르쳐주시겠습니까	1 73	教えていただけますでしょうか

<그림 6> examj_dic

4.2.2 일본어 생성

위에서 설명한 대역 사전을 이용하여 TB-TDMT에서 입력 문장에 대응하는 일본어를 생성하는 과정을 개략적으로 살펴보면 [알고리즘1]과 같다.

[알고리즘1] 일본어 생성 과정
단계1. 의존 구조 입력
단계2. 토큰에 대한 대역어 결정
단계3. 문장부호의 생성과 무의미의 제거
단계4. 모호성 해결
단계5. 미등록 토큰에 대한 처리
단계6. 일본어 출력

단계2에서는 입력되는 각 토큰에 대한 대역어를 결정하는데 의존트리를 따라가면서(travel) 대역어를 적재한다. 즉, 모든 token에 대해서 token_dic으로부터 대응하는 일본어를 찾는다. 찾아진 토큰들은 의존트리를 따라 올라가면서 복합토큰으로 모아서 examj_dic에서 적중하는 대역어를 결정한다. 단계3에서는 마침표(period)등 일본어 문장에 적합한 문장 부호를 생성한다. 그리고, 무의미어를 입력된 의존 트리로부터 지운다. 무의미어에 해당하는 리스트는 forbid_dic과 경험적인 정보를 포함하고 있는 내부 테이블에 등재되어 있다. 단계4에서는 단계2에서 적재해 둔 대역어들 중에 모호성이 있는 것들을 골라 모호성을 제거한다. 여기에서는 다음과 같은 세 가지 정보를 이용해서 모호성을 해결한다. 첫째 동사에 포함된 경험 정보, 둘째, 모호성을 가지고 있는 토큰에 대한 함수 정보 셋째, 번역 예제

(examj_dic)를 이용하는 것이다. 단계5에서는 대역 사전에 등록되어 있지 않은 토큰에 대한 처리를 한다. 토큰에 대한 대역어를 발견할 수 없을 경우에는 형태소 단위로 번역하여 그 결과를 결합한 후 token_dic에서 최적의 대역어를 결정하고 모호성이 존재하면 단계4로 간다. 생성의 마지막 함수인 단계6에서는 일본어 출력을 하는데 옵션에 따라 KS, SJIS, EUC-JIS의 코드로 출력할 수 있다.

5. 실험 및 평가

구현한 시스템은 여행계획 영역의 전사된 텍스트 1500 발화 중 임의로 선택한 600 발화를 대상으로 평가되었다. 자연언어의 번역 결과에 대한 평가에서는 정확히 일치하지는 않아도 의미가 통하는 문장을 틀린 번역 결과라고 단정지을 수만은 없다. 따라서 정량에 의한 객관적인 평가 방법을 사용할 수는 없고 평가자들의 주관적인 평가에 의존 할 수밖에 없다. 본 논문에서는 한국인으로서 일본어 실력이 우수한 사람 4명과 일본인이면서 한국어 실력이 보통인 사람 1명에게 평가를 의뢰해서 그들의 주관적인 점수를 받아 집계하였다. 평가의 기준은 다음과 같이 세 가지의 등급으로 나누었다.

A: 번역 결과가 일치하는 문장
B: 일치하지는 않지만 발화의 취지가 전달된 경우
C:: 의미전달에 실패한 경우

평가자들은 원문과 결과문장을 비교하여 세 가지 중의 하나의 점수를 매겼고 우리는 이 점수 중 A와 B는 번역성공으로 C는 실패로 간주하였다.

<표 1> 번역 성공률 평가 결과

제한된 환경	일반 환경
87%	71%

평가 결과는 <표 1>과 같다. 표에서 제한된 환경이라 함은 음성인식의 결과 중 사전을 만드는 데 사용된 문장들을 중심으로 입력문장을 선택한 경우를 말한다.

번역에 실패한 경우의 63%는 지명과 인명이 포함되어 이를 대치할 만한 대역어를 사전에서 찾아내지 못한 문장들이었다.

6. 결론 및 향후 연구

본 논문에서는 대화 환경에서 한국어의 문장을 입력으로 받아들여 일본어 문장으로 번역해내는 자동 번역 시스템인 TB-TDMT에 관하여 설명하였다. TB-TDMT는 문장의 각 구성요소들의 자유도가 매우 높은 대화체 한국어 문장의 특성을 수용하기 위하여 새로운 분석 및 대역 단위(TOKEN)를 설정하여 사용하였다. 본 시스템은 SUN SPARC 시스템에서 gcc를 사용하여 구현하였다. 구현된 시스템은 여행계획 영역에서 수집된 예문 600 발화를 대상으로 실험하였다. 그 결과 제한된 환경에서는 약87%, 아무런 제약이 없는 환경에서는 약 71%의 번역 성공률을 보였다. 번역에 실패한 경우를 분석해 보니 실패한 문장의 약 63%가 사람의 이름과 지명에 관련되어 있었다.

대화체 문장에서 무시할 수 없는 간투어의 제거를 위하여 본 시스템에서는 단순히 기본 대역사전과 유사한 수준의 forbid_dic을 사용하고 있다. 그런데 이 방법은 간투어를 처리 대상 문장 속에 번역의 마지막 단계에까지 포함하고 있어서 이전 단계에서 많은 모호성을 발생시키는 것은 물론 번역 효율을 떨어뜨리는 요인으로 작용한다. 따라서 형태소 분석의 전 단계에서 간투사를 제거하는 방향으로 개선하고 있다. 또한, 본 논문에서 사용하는 자료구조는 여러 가지 경험정보를 번역에 동시에 사용할 수 있도록 구성되어 있는데 새롭고 잘 정제된 경험정보들을 추가함으로써 번역의 질을 높일 수 있도록 하는 연구를 계속 진행 중이다.

참 고 문 헌

[1] Fruse, O., Iida, H., "An Example_Based Method for Transfer Driven Machine Translation," proc. of 4th Int'l Conf.on Theoretical & Methodologcal Issues in Machine Translation(TMI-92), pp. 139-150, 1992.

[2] Fruse, O., and Iida, H., "Cooperation between Transfer and Analysis in Framework," Proc. of Coling-92, pp645-651, 1992.

[3] Nagao, M., " A Framework of a Mechanical Translatior between Japanese and English by Analcgy Principle,", in A. Elithorn and R. Banerji(ed.), Artificial and human Intelligence, North Holland, pp. 173-180, 1984.

[4] Nagao, M., "Some Rationales and Methodologies

for Example-based Approach," Proc. of int'l workshop on Fundamental Research for Future Generation of Natural Language Processing, pp. 61-81, 1992.

[5] Sumita, E. Iida, H. , and Kohyma, H., "Translating with Examples. A New Approach to Machine Translation," Proc. of The 3rd Int'l Conf. on Theoretical and Methodological Issue in Machine Translation of Natural Language(TMI'90), pp. 203-212, 1990.

[6] Sumita, E. and Iida, H., "Experiments and Prospects of Example-Based Machine Translation," Proc. of the 28th Annual Meeting of the Assoc. for Computational Linguistics(ACL'91), pp. 185-192, 1991.

[7] Kolodner, J., "Case-Based Resoning," Tutorial Textbook of 11th IJCAI, 1989.

[8] Stanfull, C. and Waltz, D., "Toward Memory-Based Reasoning," Commun. of the ACM, vol. 29, no. 12, pp. 1213-1228, 1990.

[9] 강승식, 음절정보와 복수어 단어 정보를 이용한 한국어 형태소 분석, 서울대학교 컴퓨터공학과 박사학위논문, 1993.

[10] 윤덕호, 우요섭, 한국어사전의 확장과 대화체 Tagged Corpus 구성 용역, 한국전자통신연구소, 용역결과보고서, 1996.

양 승 원

1985. 2. 전북대학교 전산통계학과 졸업
1987. 2 전북대학교 대학원 이학 석사
1995. 8 전북대학교 대학원 이학 박사
1999. 12 현재 우석대학교 정보통신 컴퓨터공학부 조교수