한국산업정보학회논문지 제9권 제3호(2004. 9)

토큰기반 변환중심 한일 기계번역을 위한 변환사전†

(Transfer Dictionary for A Token Based Transfer Driven Korean-Japanese Machine Translation)

양 승 원*
(Seungweon Yang)

요 약 한국어와 일본어는 동일한 어족에 속하며 비슷한 문장구조를 가지고 있어 변환중심 기계번역 방법이 효율적이다. 본 논문에서는 토큰 단위의 변환중심 한일 기계번역 시스템을 위한 변환 사전을 생성하는 방법에 관하여 기술하였다. 변환 사전이 잘 구성되면 구문분석 단계에서는 대역어를 선정하기에 적합한 정도까지의 의존트리를 생성하는 간이 파싱 만을 함으로써 필요 없는 노력을 경감시킬 수 있다. 게다가 구문해석 시에 최종의 결과 트리를 만들지 않아도 되므로 문어체 문장은 물론 입력 형태가 비정형적인 대화체 문장에서 더욱 큰 효과를 볼 수 있다. 본 논문의 변환 사전은 한국전자통신 연구원이 수집한 음성 데이터베이스로부터 추출한 말뭉치를 사용해 구성하였다. 구현한 시스템은 여행 계획영역에서 수집된 900여 발화 안의 문장을 대상으로 시험하였는데 제한된 환경에서 92%, 아무런 제약이 없는 환경에서는 81%의 성공률을 보였다.

핵심주제어 : 변환중심 기계번역시스템, 변환사전, 토큰

Abstract Korean and Japanese have same structure of sentences because they belong to same family of languages. So, The transfer driven machine translation is most efficient to translate each other. This paper introduce a method which creates a transfer dictionary for Token Based Transfer Driven Koran-Japanese Machine Translation(TB-TDMT). If the transfer dictionaries are created well, we get rid of useless effort for traditional parsing by performing shallow parsing. The semi-parser makes the dependency tree which has minimum information needed output generating module. We constructed the transfer dictionaries by using the corpus obtained from ETRI spoken language database. Our system was tested with 900 utterances which are collected from travel planning domain. The success-ratio of our system is 92% on restricted testing environment and 81% on unrestricted testing environment.

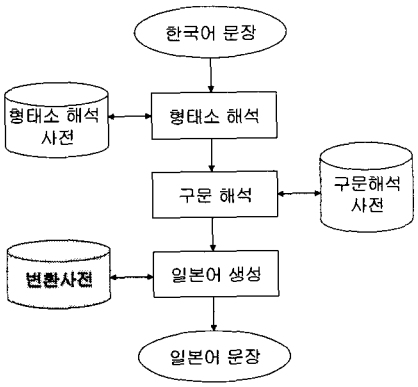
Key Words : Transfer Driven Machine Translation, Transfer Dictionary, Token

1. 서 론

한국어와 일본어는 동일한 어족에 속하며 비슷한 문장구조와 문자수준을 가지고 있어 변환중심 기계번역(TDMT: Transfer Driven Machine Translation) 방법이 자동번역에 효율적일 수 있다. 변환중심 기계번역은 이전에 성공적으로 번역된 예문들을 저장해 둔 데이터베이스와 시소러스(thesaurus)를 이용하여 가장 확률이 높은 예문을 번역의 해(goal)

† 이 논문은 한국전자통신연구원의 지원에 의하여 연구되었음.
* 우석대학교 컴퓨터공학과 부교수

로 선택하는 예제기반 기계번역(EBMT: Example Based Machine Translation)에서 비롯되었다[1,2]. 변환중심의 번역은 예제의 변환방법에 해석적인 지식을 포함시켜 규칙기반번역의 특성을 결합함으로써 문장을 번역해 낼 수 있도록 설계되었다. 이 방법의 기계 번역기는 해석모듈과 변환모듈을 가지고 있다. 이 두 모듈은 서로 완전히 독립적으로 맡은 일을 수행한다. 해석 모듈은 자연언어 처리의 기반인 형태소해석과 구문해석으로 이루어져 있으며 이 모듈의 출력은 변환 모듈의 입력이 되어 대상 언어를 생성하게 된다. 변환중심 번역에서는 변환모듈이 중심이 되어서 번역이 이루어지므로 원문에 대한 적절한 대역문을 찾아내는 것이 가장 중요하므로 기타의 모듈들은 대역어 변환 작업을 돕는 변환 지식들을 생성하여 제공한다. <그림 1>은 변환중심 번역 시스템의 구성도이다.



<그림 1> 변환중심 번역시스템의 구성도

이 방법에서 입력문장에 대한 번역 문장을 찾을 때, 통계적인 지식에 의존하여 가장 근접한 번역문을 찾아내므로 대역어 정보를 수록한 변환 사전의 역할이 가장 중요하다. 변환 사전은 해석모듈의 형태소해석이나 구문해석에서 사용하는 사전과는 다른 역할을 하는 사전이며 내용 또한 다르다. 이 사전의 주된 내용은 양국어 사이에 서로 대응되는 원소들을 엔트리로 가지고 있다. 변환 사전을 잘 구축하면 번역기 전체 시스템이 상당히 간결해 진다. 왜냐하면 해석 모듈의 구문해석기가 적절한 대역어를 찾아 낼 수 있을 정도의 파싱만을 수행하면 되므로 해석모듈 단계의 모호성이 현저히 감소하기 때문이다.

본 논문에서는 변환중심의 패러다임을 채용하고 한국어 대화체 문장의 문장성분들의 자유도(degree of freedom)가 매우 높은 점을 감안하여 고안된 토큰(Token)이라는 새로운 문장 해석의 기본 단위를 기반으로 한 한·일 기계 번역기인 TB-TDMT (Token Based Transfer Driven Machine Translation) [3]를 위한 변환 사전 구성 방법에 대해 기술한다. 또한, 구성된 변환 사전을 TB-TDMT에 장착하여 실험한 결과를 제시한다.

2. 관련연구

2.1 변환중심 기계번역(TDMT)

변환중심 기계번역은 예제중심 기계번역(EBMT)을 골격으로 발표되었다. EBMT는 번역된 예제와 시소러스의 중요성을 강조한 방법이다[4,5,6,7]. 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<그림 1>. 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 해석부, 구문 해석부, 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.

$$SE \rightarrow TE_1(E_{11}, E_{12}, \ldots)$$
$$\vdots$$
$$TE_n(E_{n1}, E_{n2}, \ldots) \quad (1)$$

각 TE는 조건으로서 여러 보기를 가지고 있다. E_{ij} 는 TE_i의 j번째의 보기를 의미한다. 입력이 SE 일 때에 그 입력과 보기들의 거리를 계산하여 가장 적절한 TE가 선택된다. 입력(I)과 보기(E_{ij})의 의미적인 거리 즉, $d(I,E_{ij})$는 식 (2)와 같이 계산된다.

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

여기에서 $0 \le d(I,E_{ij}) \le 1$이고 거리를 계산하는 방법으로는 사례기반 추론(case-based reasoning)에서 이용하는 Most Specific Common Abstraction (MSCA)[8]을 사용한다. MSCA는 두 패턴이 의미적으로 유사하면 유사할 수록 작아진다. w_k는 식 (3)과 같이 정의하며, 각 I_k에 대한 TE의 분포를 나타낸다[9].

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

식 (3)에서 f는 말뭉치로부터 얻어 낸 것으로 I_k가 E_{ijk}로 번역된 빈도수이다. w_k를 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 E_{ij} 가 I 에 가장 가까울 때 TE_i가 가장 가능성 있는 TE로 선택된다. TDMT에서는 보기가 많으면 많을수록 더 정확한 TE를 결정할 수 있다. 왜냐하면 TE를 결정하는 조건이 더 세부적이기 때문이다.

2.2 토큰 기반 번역 모델

[3]에서는 어절과 문장의 중간 개념인 토큰을 다음과 같이 정의하고 이를 해석과 변환의 기본 단위로 삼았다.

「정의 1」 토큰(TOKEN)

① 하나의 의미상의 내용어와 그에 수반된 의미상의 기능어로 구성되는 문장의 구성단위.

② 토큰의 범주는 명사구토큰(NP), 동사구토큰(VP), 부사구토큰(AP), 관형사구토큰(DP) 등의 기본토큰과 대화체 문장에 적응하기 위하여 간투사토큰(IG), 구분자토큰(SP) 등의 변형토큰을 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 2> 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 해석으로는 얻을 수 없는 것이다. 즉, 3번 토큰은 원래에는 각각의 의미를 갖는 '영상'과 '축전'의 두 개의 형태소로 되어있으나 이 둘은 모여서 단일 의미를 가지므로 하나의 의미상 내용어로 분리된다. 게다가, 10번의 '알고싶어요'는 일정한 패턴을 갖는 숙어 성분이 하나의 토큰으로 분리된 예이다. 원래의 형태소 해석에 따로 떨어진 상태(알고+싶습니다)로는 본 논문에서 목적언어(target language)로 삼고 있는 일본어에서 적절한 변환 해(goal)를 찾기가 곤란하나 토큰으로 분리된 상태에서는 '知りたいんです'로 자연스러운 번역이 가능하다.

3. 변환 사전 구축

3.1 양국어 대응 말뭉치 정렬

한국어와 일본어가 구조적으로 매우 유사하기는 하지만 미묘한 부분에서는 서로 다른 부분이 존재한다. 이 차이는 번역의 질을 떨어뜨리는 주요 요인이며 심지어는 이 차이로 인하여 번역에 실패하기도 한다. 우리는 변환사전을 구축하기에 앞서서 이러한 차이점을 최대한 줄이기 위하여 말뭉치의 정렬작업을 선행하였다. 정렬에는 다음과 같이 세 가지의 메타기호를 사용하였다.

① + : 이 기호는 기본적인 형태소를 토큰단위로 연결할 때 사용한다.
예) 이름+은 이근호+입니다
名前+は イグンホ+です

② @ : 이 기호는 한국어의 경우 띄어쓰기가 되어 있는 원소에 대응하는 일본어의 특성상 띄어쓰기를 무시하고 붙여 써야 할 때 원래의 띄어쓰는 곳을 표시한다. 즉, 한국어 문장의 어절 간 표시기호이다.
예) 종류에@대해서
種類について

③ * : 한일 상호간에 대응되는 토큰 상에 부족한 형태소가 있는 경우에 사용한다.
예) 방*종류를 알고싶어요
部屋の種類を 知りたいんです

다음은 이러한 기호를 사용하여 정렬한 말뭉치의 원문 중 일부이다.

De001 Se001 m01 DATe01 00004-00330 Agnt
01-EK : 안녕하십니까. 아메리칸투어+입니다.
통 : こんにちは. アメリカンツアー+です.
02-한 : 안녕하세요. 미국 패키지투어+의 종류+를 알고@싶은데요.
통 : こんにちは. アメリカ パッケージツアー+の 種類+が 知りたいんですが.
03-EK : 네, 저희+는 두@가지+의 투어+가 있는데요. 하나+는 하와이+로 가는 거 하나+는 동부+로 가는 것+입니다.
통 : はい, 私ども+は 二つ+の ツアー+が ありますが. 一つ+は ハワイ+へ 行く もの 一つ+は 東部+へ 行く もの+です.
04-한 : 예, 그러면 각 패키지투어+의 기간+과 요금+을 좀 알@수@있습니까.
통 : はい, では 各 パッケージツアー+の 期間+と料金+が ちょっと 分りますか.

3.2 변환 사전의 구성

본 논문에서 구성한 변환사전의 종류는 token_dic, examj_dic, forbid_dic, name_dic, place_dic이 있다. 이들 중 가장 기본이 되는 token_dic은 3.1절에서와 같이 정렬된 말뭉치(aligned corpus)로부터 생성한다. 정렬한 말뭉치를 사전의 엔트리로 변환하기 위하여 정렬시 추가한 메타 기호들을 삭제한다. 기호를 삭제하고나면 서로 대응하는 한국어와 일본어 문장은 같은 수의 토큰을 가지게 되므로 <그림 3과>과 같은 간단한 알고리즘으로 기본 변환사전을 생성할 수 있다.

1. 필요없는 문장을 스킵; //대화의 종류 번호
2. if (한국어 키가 사전에 없으면) {
3. 대역어 레코드를 사전에 추가하고 1로 감;.
}
4. elseif (대역어 필드에 이미 동일한 대역어가 있으면) 1로 감;
5. 대역어 필드에 추가 하고 1로 감;

<그림 3> token_dic 생성 알고리즘

<그림 4>는 token_dic의 사전 항목에 관한 구체적인 예이다. 이 기본 변환 사전은 하나의 토큰에 대응하는 여러 개의 대역어를 가질 수 있으며, 다중 대역어에 대한 모호성은 참조 시에 함수에 의해서 해결된다. 우리는 이외에도 인명 변환 사전(name_dic)과 지명 변환 사전(place_dic)을 보조 사전을 추가하였다.

토 큰	번역 정보	대 역 어
알고싶은데	1 2 0 4024	知りたいん 知りたいんですが
알고싶은데요	1 3 0 4025	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 4026	しるため
알기위해서	1 1 0 4027	しるために
알려드리겠고요	1 1 0 4028	お知らえいたします
알려드리겠습니다	1 2 0 4029	お教えいたします お知らせいたします

<그림 4> token 변환 사전

<그림 4>의 필드 중 가운데의 숫자들은 번역의 질을 높이기 위하여 사용하는 변환외적인 정보들이다. 각각은 처음부터 토큰의 범주, 대역어의 개수, 대응되는 명사가 2개인 경우 스타일, 사전 내의 레코드 번호를 나타낸다. 예를 들면, 토큰 '알고싶은데'는 토큰의 범주가 1로서 VP를 나타내고 대역어의 개수는 2개 그 다음 수자는 명사형이 아니므로 0이며 마지막의 수자는 이 토큰이 저장된 위치가 4024번째의 레코드임을 나타낸다.

의존트리의 에지	번역 정보	대역어
가격을싸게하고싶은데	1 128	値段を安くしたいと思うんです
가고싶은데요	1 129	行きたいんです
가는게있습니까	1 130	行くことがありますか
가려구하는데요	1 131	行きたいと思っておりますけど
가르쳐주시겠습니까	1 132	教えていただけますでしょうか

<그림 5> 예제 사전

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 5>와 같이 구성되어 있다. 이 사전 안에서 수자들의 의미는 대역 필드의 개수와 레코드 번호이다.

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

name_dic과 place_dic은 각각 인명과 지명에 대한 변환사전이다. 이 두 개의 사전은 [3]의 연구결과에서 해석 오류의 상당 부분을 인명과 지명을 포함한 문장이 차지하고 있다는 점을 보완하기 위하여 만들었다.

4. 일본어 생성

생성된 변환 사전을 이용하여 TB-TDMT에서 입력 문장에 대응하는 일본어를 생성하는 과정을 개략적으로 살펴보면 <그림 6>과 같다.

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

<그림 6> 일본어 생성 과정

해석부로부터 구문해석의 결과인 의존 트리를

입력으로 받으면 해석부에서는 변환사전을 참조하여 각 노드의 토큰과 대응하는 일본어를 생성한다. 단계2에서는 입력되는 각 토큰에 대한 대역어를 결정하는데 입력된 의존트리를 따라가면서(travel) 대역어를 적재한다. 즉, 모든 token에 대해서 token_dic으로부터 대응하는 일본어를 찾는다. 찾아진 토큰들은 의존트리를 따라 올라가면서 복합 토큰으로 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 변환 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. 단계6은 생성의 마지막 함수로서 선택된 일본어를 출력한다.

5. 실험 및 평가

구현한 시스템은 여행계획 영역의 전사된 텍스트 1500 발화 중 임의로 선택한 900 발화를 대상으로 평가되었다. 이 평가 방법은 [3]에서 실시한 방법과 동일한 것으로 두 시스템의 비교를 위하여 채택하였다. 자연언어의 번역 결과에 대한 평가에서는 정확히 일치하지는 않아도 의미가 통하는 문장을 틀린 번역 결과라고 단정 지을 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 번역 성공으로, C는 실패로 간주하였다.

<표 1> 번역 성공률 평가 결과

제한된 환경	일반 환경
92%	81%

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

실험에서 번역에 실패한 부분은 변환 사전에 복수로 등재된 엔트리 중 잘못된 엔트리를 가져온 경우와 등재되지 않은 인명이나 지명 그리고 숫자 등이었다.

6. 결 론

본 논문에서는 대화 환경에서 한국어의 문장을 입력으로 받아들여 일본어 문장으로 번역해내는 자동 번역 시스템인 TB-TDMT를 위한 사전 구축 방법에 관하여 기술하였다. [3]에서의 실험 결과 인명과 지명을 포함한 문장에서 번역오류가 빈번한 것에 착안하여 본 연구에서는 인명과 지명 변환사전을 추가하였다. 변환사전 구축을 위하여 몇 가지의 자동화 모듈을 구현하였는데 이들은 Windows XP 위에서 Visual C++ 6.0을 사용하여 구현하였다. 구축한 사전을 TB-TDMT에 장착해 예문 900 발화를 대상으로 실험하였다. 그 결과 제한된 환경에서는 약92%, 아무런 제약이 없는 환경에서는 약 81%의 번역 성공률을 보였다. 번역에 실패한 경우를 분석해 보니 변환 사전에 복수로 등재된 엔트리 중 잘못된 엔트리를 가져온 경우와 등재되지 않은 인명이나 지명 그리고 숫자 등이었다.

본 논문에서 인명이나 지명을 위한 변환 사전을 추가 했으나 모든 고유명사를 한정된 사전 안에

등재하는 것은 불가능한 일이다. 따라서, 대역어 찾기에 실패한 경우 이들을 분석해서 적절한 사전에 등재할 수 있는 메카니즘이 필요하다. 숫자 또한 마찬가지인데 이는 변환사전과는 별도로 숫자의 각 디지트(digit) 단위로 대응시켜주는 내부 루틴이 필요하다.

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] 양승원, "대화체 문장 번역을 위한 토큰기반 변환중심 한일 기계번역," 한국산업정보학회논문지, 제4권 제4호, pp. 40-46, 1999.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

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

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

양 승 원 (Seungweon Yang)

우석대학교 컴퓨터공학과 부교수
한국전자통신 연구원 초빙연구원
University of Guelph 방문교수

(관심분야: 자연언어처리, 바이오 인포믹스)