韓國 컴퓨터情報學會 論文誌
第11卷 第4號, 2006. 9.

2006-11-4-1-21

# Needleman-Wunsch 알고리즘을 이용한 유사예문 검색

김 동 주*, 김 한 우*

# Searching Similar Example-Sentences Using the Needleman-Wunsch Algorithm

Dong-Joo Kim*, Han-Woo Kim*

## 요 약

본 논문에서는 번역지원 시스템을 위한 유사예문 검색 알고리즘을 제안한다. 유사예문 검색이란 질의문에 대하여 구조적, 의미적으로 유사한 예문을 찾는 것으로 번역지원 시스템의 핵심 요소이다. 제안하는 알고리즘은 생물정보학 분야에서 두 단백질의 아미노산열의 유사성을 판별하기 위한 Needleman-Wunsch 알고리즘에 기반하고 있다. 표면정보만 이용하는 Needleman-Wunsch 알고리즘을 그대로 문장 비교에 적용하였을 경우 단어 굴절요소에 민감하여 의미적으로 유사한 문장을 발견하지 못할 가능성이 높다. 따라서 표면 정보 외에 단어의 표제어 정보를 추가적으로 이용한다. 또한 문장 구조의 유사성 정도를 반영하기 위해 품사 정보를 이용한다. 즉, 본 논문에서는 단어의 표면 정보, 표제어 정보, 품사 정보를 융합한 문장 비교 척도를 제안한다. 그리고 이 척도를 이용하여 유사 문장을 검색하고, 유사성에 기여하는 부분쌍을 파악하여 결과로 제시한다. 제안하는 알고리즘은 전기통신 분야의 데이터에 대해 매우 우수한 성능을 보였다.

## Abstract

In this paper, we propose a search algorithm for similar example-sentences in the computer-aided translation. The search for similar examples, which is a main part in the computer-aided translation, is to retrieve the most similar examples in the aspect of structural and semantical analogy for a given query from examples. The proposed algorithm is based on the Needleman-Wunsch algorithm, which is used to measure similarity between protein or nucleotide sequences in bioinformatics. If the original Needleman-Wunsch algorithm is applied to the search for similar sentences, it is likely to fail to find them since similarity is sensitive to word's inflectional components. Therefore, we use the lemma in addition to (typographical) surface information. In addition, we use the part-of-speech to capture the structural analogy. In other word, this paper proposes the similarity metric combining the surface, lemma, and part-of-speech information of a word. Finally, we present a search algorithm with the proposed metric and present pairs contributed to similarity between a query and a found example. Our algorithm shows good performance in the area of electricity and communication.

▷ Keyword : computer-aided translation(번역지원시스템), Needleman-Wunsch algorithm, translation memory(번역 메모리), alignments(정렬)

• 제1저자 : 김동주
• 접수일 : 2006.08.12, 심사일 : 2006.09.15, 심사완료일 : 2006.09.20
* 한양대학교 컴퓨터공학과

# Ⅰ. 서 론

번역 지원(CAT: Computer-Aided Translation) 시스템[1][2]이란 인간 번역가가 번역을 수행하는데 필요한 각종 도구들을 내장하여 번역이 용이하도록 번역 과정을 지원하는 시스템으로, 번역 보조(*computer-assisted translation*) 시스템이라고 부르기도 한다. 1950년대에 시작된 기계번역에 관한 연구가 1960년대에 들면서 한계에 부딪혀 실현성이 떨어진다는 결론에 이르자 대안으로 번역의 주체를 기계가 아닌 인간으로 하고 기계의 역할을 인간 번역가를 지원하는 보조도구로 축소한 반자동 번역에 관심을 두게 되면서부터 번역지원 시스템이 등장하기 시작하였다. 이 시스템은 완전한 자동번역을 수행하지 않는다는 점에서 기계번역과 근본적인 차이점을 갖는다. 일반적으로 번역 지원 시스템은 맞춤법 검사기, 전문용어 관리기, 용례 검색기, 대역어 사전, 등의 인간이 직접 번역을 수행하는데 도움이 되는 많은 도구들을 내장하고 있을 뿐만 아니라, 비록 제한적일지라도 심지어는 자동번역기를 내장하기도 한다. 이 시스템에서 무엇보다 핵심이 되는 부분은 번역 메모리(translation memory)[2][3][4]라는 기술에 기반을 둔 유사예문 검색(혹은 매칭)이다.

유사예문 검색 시스템은 사전에 번역된 번역 예문 쌍의 집합으로부터 입력문장에 대하여 구조적, 의미적으로 가장 유사한 해당 언어의 예제 문장들을 검색하고, 검색된 예문과 예문에 상응하는 대역 문장 쌍들을 사용자에게 제시해 줌으로써 번역을 용이하게 하도록 도와주는 시스템이다. 이는 예제기반 기계번역(example-based machine translation)[5]에서도 필수적인 요소이다. 번역 지원 시스템을 위한 유사 예문 검색에서 중요한 요소는 동일 언어로 작성된 문장 간의 유사성 정도의 계산이다. 이와 더불어 유사성 정도에 기여하는 부분이 두 문장에서 어느 부분인지도 제시할 수 있어야 한다. 이를 정렬(alignment)이라고 부른다. 또한 번역 대상이 되는 원어 문장(source language sentence)과 이에 상응하는 대역 문장(target language sentence)의 예문 쌍에서 대역관계에 놓인 부분의 파악, 즉 대응관계(correspondence relation) 파악 또한 사전에 이루어져야 한다. 기계번역 분야에서 일반적으로 정렬이란 원어 문장과 대역 문장 사이에서 대역관계에 놓인 부분을 파악하는 것을 의미하나 본 논문에서는 동일 언어로 된 질의문과 예문 사이의 '유사성에 기여하는 부분의 파악'과 구분을 위해 '대역관계에 놓인 부분을 파악'하는 일은 '대응관계 파악'이라고 할 것이다.

번역 메모리를 포함하여 예제기반 기계번역, 유사기반(analogy-based) 기계번역, 사례기반(case-based) 기계번역, 통계기반 기계번역 등과 같은 코퍼스기반(corpus-based) 기계번역[6] 방식에서 한 언어의 문장이 다른 언어로 번역되는 경향을 살피고, 그 경향을 학습하기 위해 선행되어야 할 작업이 대응관계 파악이다. 병렬 코퍼스로[2]부터 대역사전의 자동 구축과 동일한 과정으로 생각할 수 있는 대응관계 파악은 병렬코퍼스에서 한 언어로 작성된 문서의 문단, 문장 혹은 단어와 같은 단위 요소에 대해 다른 언어로 작성된 문서 내에서 대역관계에 놓인 요소를 찾는 것이다. (1)은 대응관계의 단위가 단어인 병렬문장쌍의 예를 보이고 있다.

(1) a. He eats vegetables
그는 채소를 먹는다

b. She bring me potatoes
그녀는 내게 감자를 가져왔다

본 논문에서는 (1-a), (1-b)와 같이 대응관계가 파악된 영한 병렬코퍼스가 준비되어 있다고 가정한다. 영한 번역을 위한 유사예문 검색시스템을 구축하기 위해 영어 입력 문장과 영어 예문의 유사성 정도를 계산하면서 유사성 정도에 기여하는 부분들을 찾아내 정렬을 수행하는 알고리즘을 제시한다. 제시하는 알고리즘은 Needleman-Wunsch 알고리즘[7]에 기반한 알고리즘으로 문장 비교에 적합하도록 개량한 것이다.

# Ⅱ. 관련 연구와 연구의 범위

CAT는 문서편집기, 맞춤법 검사기, 대역어 사전, 전문용어(terminology)[8] 관리기, 번역 메모리, 용례(concordance)[9] 검색기 등과 같은 인간 번역을 돕기 위한 많은 도구들을 내장하고 있다. 이러한 도구들

2) 대역관계에 있는 예문 쌍들의 집합

중에서 간단한 몇 가지 도구들을 제외한 맞춤법 검사기, 대역어 사전 구축, 전문용어 구축, 용례 검색은 또 다른 연구 범주로 간주될 만큼 광범위할 뿐만 아니라 본 논문에서의 핵심 쟁점인 유사예문 검색과는 다른 문제이므로 논의 대상에서 제외한다.

번역 메모리에서 핵심인 유사예문 검색, 혹은 매칭 문제는 검색된 예문의 질(quality)과 관련된 정확성 문제[10][11][12]와 검색 시간과 관련된 효율성 문제[10][12][16]로 크게 두 가지 쟁점으로 나뉜다. 효율성 문제와 관련하여서 최근 접근 방법들은 여과(filtering)나 군집화(clustering)를 통한 검색 공간 축소[12], 그리고 중복계산 회피 방식[10]으로 이루어지는데, [10]에서는 [11]의 방법에 대한 중복 계산을 없애 검색 속도를 향상시키기 위해 예문을 그래프 자료구조로 저장하는 방법을 제안하였다. [12]에서는 각각의 쟁점에 대해서 두 가지 방법을 제시하였다. 정확성 문제에 대해 몇 가지 언어학 정보를 이용한 2단계 동적 프로그래밍(two-level dynamic programming) 방법을 제안하였고, 검색 공간을 축소하여 검색 시간을 단축하기 위해 변형된 k-평균군집화(modified k-means clustering) 기술을 이용한 군집화 방법을 제안하였다.

본 논문에서는 정확성 문제에만 다루고 있으며, 따라서 비교 대상이 되는 방법은 [10]의 2단계 동적 프로그래밍 방법과 [11]의 DP-matching Driven transDucer($D^3$) 방법이다. 그러나 [10]의 방법은 사용된 언어학적 정보 추출이 항상 성공하지 않는다는 단점을 지니고 있으며 [11]의 방법은 모든 매트릭스 요소에 사용된 모든 언어학적 정보를 유지해야 하기 때문에 계산 량이 많다는 단점을 지니고 있다. 또한 [10]에서 사용된 코퍼스는 그리스어와 영어로 된 병렬코퍼스를 사용하고 있으며, [11]에서는 일본어와 영어로 된 병렬코퍼스를 사용하고 있어 유사 문장 검색에서 검색 대상이 되는 언어는 각각 그리스어와 일본어이다.

본 논문에서 제안하는 알고리즘은 영한 번역을 위한 번역지원시스템에서 사용될 유사문장 검색 알고리즘이다. 따라서 목표로 하는 영한 번역을 위해서는 유사문장 검색에서 검색 대상은 영어이어야만 한다. [10]과 [11]에서 제안하는 방법론 자체에 대한 심도 있는 논의는 필요하겠지만 각 논문에서 제시하고 있는 성능 평가 결과의 직접 비교는 불가능하다. 따라서 본 논문에서는 다른 방법과의 직접적인 비교 평가는 배제하고 사람 판단에 의한 투표(voting) 방식으로 영어에 대한 유사문장 검색 정확성만을 평가한다.

## III. 유사문장 검색

입력 문장에 대하여 가장 유사한 예문들을 병렬 코퍼스 예문들로부터 검색하기 위해서는 동일 언어 문장들 간의 유사도(similarity), 또는 거리(distance)에 대한 척도를 정의해야 한다. 또한 일치하는 부분에 대한 대역 예문을 선택적으로 가려내어 유사성 정도에 기여하는 부분들을 파악하기 위해 번역하고자 하는 문장 언어 예문으로부터 일치하는 위치 정보도 알아내야 한다. 이를 위해 본 논문에서는 Needleman-Wunsch(NW) 알고리즘에 기반을 둔 전역적 문자열 비교 알고리즘을 사용한다.

### 3.1 Needleman-Wunsch 알고리즘

NW 알고리즘은 그림 1과 같이 생물정보학 분야에서 두 단백질에서의 아미노산열의 유사성을 판별하기 위한 알고리즘으로 편집거리(edit distance)[13] 척도의 가중치 집합을 일반화한 것이다.

$$A(0,0)=0$$

$$A(i,0)=i\times w_d,\ A(0,j)=j\times w_d$$

$$A(i,j)=\max\begin{cases}A(i-1,j-1)+\alpha & \text{[case 1]}\\ A(i-1,j)+w_d & \text{[case 2]}\\ A(i,j-1)+w_d & \text{[case 3]}\end{cases}$$

$$\text{where}\qquad \alpha=\begin{cases}w_m & \text{if}\quad x_i=y_j\\ w_s & \text{if}\quad x_i\neq y_j\end{cases}$$

$$Ptr(i,j)=\begin{cases}Diag & \text{[case 1]}\\ Up & \text{[case 2]}\\ Down & \text{[case 3]}\end{cases}$$

그림 1. Needleman-Wunsch 알고리즘
Fig. 1. Needleman-Wunsch Algorithm

즉, 그림 1의 알고리즘은 아미노산열 $X=x_1\cdots x_m$을 $Y=y_1\cdots y_n$으로 변환하기 위해 필요한 일치, 대치, 삽입/삭제 연산 각각에 가중치 $w_m$, $w_s$, $w_d$를 주어 matrix $A$를 계산하게 된다. 이 가중치는 유사성 정도에 기여하는 일치 연산에 대해서는 높은 값을 주고, 기여하

지 않는 대치나 삽입/삭제 연산에 대해서는 낮은 값으로 설정한다. 따라서 [13]에서의 편집거리 척도가 상이성 척도로 사용된 반면에 NW 척도는 유사성 척도로 사용된다. $X_i$를 $x_1$에서부터 $x_i$에 이르는 $X$의 부분열, $Y_j$를 $y_1$에서부터 $y_j$에 이르는 $Y$의 부분열이라고 했을 때, $A(i,j)$의 값은 부분열 $X_i$를 $Y_j$로 변환하는데 필요한 연산에 가중치를 주어 계산된 부분 유사도 값이다.

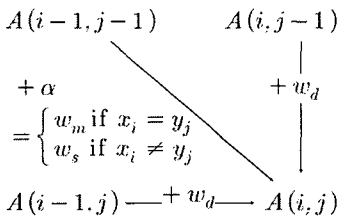


그림 2. $A(i,j)$의 계산

Fig. 2. Calculation of $A(i,j)$

그림 2는 그림 1의 알고리즘에서 $A(i,j)$를 계산하기 위한 세 가지 경로 case 1, case 2, case 3를 도식화한 것으로 case 1은 부분열 $X_{i-1}$와 $Y_{j-1}$의 유사도에 $\alpha$를 더한 값이고, case 2는 $X_{i-1}$와 $Y_j$의 유사도에 $w_d$를 더한 값, 그리고 case 3는 $X_i$와 $Y_{j-1}$의 유사도에 $w_d$를 더한 값이다. $\alpha$의 값은 $x_i$와 $y_j$가 같다면 $w_m$이 되고 같지 않다면 $w_s$가 된다. 이 세 가지 경로로부터 계산된 값 중 제일 큰 값이 $A(i,j)$가 된다. 이는 동적 프로그래밍 기법에 의해 재귀적으로 Matrix $A$가 계산되고 이와 동시에 일치하는 문자를 알아내기 위한 경로 정보, 즉 $A(i,j)$의 값이 세 가지 경로 중 어느 경로로부터 계산된 것인지에 대한 경로 정보 $Ptr(i,j)$를 유지하게 된다.

### 3.2 다층유사도 측정을 통한 유사 예문 검색

NW 알고리즘을 문장 비교 문제에 적용하기 위해서는 단어 단위의 비교를 필요로 한다. 따라서 (2)의 두 영어 문장에 대한 유사성 점수와 정렬을 구하면 그림 3과 같은 결과를 얻을 수 있으며, 이때의 가중치 $w_m$, $w_s$, $w_d$는 각각 1, -1, -2로 설정하였다. 그림 3은 두 문장 (2-a)와 (2-b)의 유사성 점수는 0이며 'reads'를 'buys'로, 'the'를 'a'로 대치하는 것이 최적의 정렬임을 의미한다.

(2) a. He reads the book

b. He buys a book

| | | He | buys | a | book |
|---|---|---|---|---|---|
| | 0 | -2 | -4 | -6 | -8 |
| He | -2 | 1 | -1 | -3 | -5 |
| read | -4 | -1 | 0 | -2 | -4 |
| the | -6 | -3 | -2 | -1 | -3 |
| book | -8 | -5 | -4 | -3 | 0 |

| He | reads | the | book |
|---|---|---|---|
| He | buys | a | book |

그림 3. 문장 (2-a)와 (2-b)에 대한 계산 예

Fig. 3. An Example for Calculation of (2-a) and (2-b)

그런데 문장 (3-a)와 (3-b)의 경우 두 문장이 구조적으로 (2-a)와 (2-b)의 경우보다 더 상이함에도 불구하고 유사성 점수는 같다. 즉, (2-a) 문장은 (3-b) 문장보다 구조적으로 (2-b) 문장이 더 유사함에도 불구하고 유사도 점수가 동일하여 번역에 더 유용한 문장을 명확히 판단하지 못하는 경우가 발생한다. 더욱 극단적인 경우에는 구조적으로 덜 유사한 문장을 유사성이 더 높은 문장으로 잘못 판단할 수도 있다. 이 문제를 해결하기 위해 단어의 표면 정보만 반영하는 것이 아니라 단어의 품사 정보를 추가적으로 반영한다. 즉, 단어의 표면형과는 무관하게 비교하는 단어의 품사가 동일하다면 표면정보 가중치 외에 품사정보 가중치를 추가적으로 준다.

(3) a. He reads the book

b. He became the spokesman

추가적으로 고려해야 할 것은 형태론적 변형에 관한 것이다. 파생접사류와는 달리 시제, 수의 일치, 단복수의 구분을 위한 굴절접사류는 어간(stem)의 의미를 변화시키지 않아 한국어로의 번역시에 번역되지 않거나 번역하지 않아도 의미상 문제가 없는 경우가 많다는 것이다. 따라서 형태론적 변형에 따른 급격한 유사도 변화를 완

화하기 위해 해당 단어의 표제어(lemma)를 비교하여 표면 정보가 같지 않더라고 표제어가 동일하면 가중치를 준다. 또한 한정사(determiner) 중 일반적으로 번역이 되지 않는 관사류는 유사도 계산에서 제외한다.

이와 같이 본 논문에서는 문장간 유사성 검사와 정렬의 정확성을 높이기 위해 표면정보 외에 단어의 어간 정보, 품사 정보를 함께 반영하는 다층 유사성 척도를 사용한다. 즉, 다층 정보는 그림 4와 같은 표면층(SL), 어간층(TL), 품사층(PL)으로 이루어진다.

| POS Layer (PL) | NNP | VB | DT | NN |
|---|---|---|---|---|
| Stem Layer (TL) | SHE | USE | THE | PENCIL |
| Surface Layer (SL) | She | uses | the | pencil |

그림 4. 다층 정보
Fig. 4. Multi-layered Information

표 1. 세분화된 가중치 집합
Table. 1. A Proposed Weights Set

| | $w_s$ (SL) | $w_t$ (TL) | $w_p$ (PL) |
|---|---|---|---|
| $w^m$ (match) | $w_s^m$ | $w_t^m$ | $w_p^m$ |
| $w^s$ (mismatch) | $w_s^s$ | $w_t^s$ | $w_p^s$ |

표면층에서 질의문과 예문의 한 단어는 각각 $s_i^x$, $s_j^y$, 어간층에서 어간은 $t_i^x$, $t_j^y$, 품사층에서 품사는 $p_i^x$, $p_j^y$이고 $1 < i < m$, $1 < j < n$이라고 했을 때, NW 알고리즘의 $\alpha$에서 문자 일치여부에 대한 가중치 $w_m$, $w_s$를 각각 $w^m$과 $w^s$으로 표기하여 표 1과 같이 세분한다. 즉, 단어의 표층형에 관한 가중치 $w_s$ 외에 단어의 어간이나 품사의 일치여부에 관한 가중치 $w_t$와 $w_p$를 추가하게 된다. 따라서 $A(i,j)$를 계산할 때, 대각 방향의 계산 $A(i-1,j-1)+\alpha$에서 $\alpha$는 식 1과 같다.

$$\alpha = w_s + w_t + w_p$$

$$w_s = \begin{cases} w_s^m & \text{if } s_{i-1}^x = s_{j-1}^y \\ w_s^s & \text{if } s_{i-1}^x \neq s_{j-1}^y \end{cases}$$

$$w_t = \begin{cases} w_t^m & \text{if } t_{i-1}^x = t_{j-1}^y \\ w_t^s & \text{if } t_{i-1}^x \neq t_{j-1}^y \end{cases}$$

$$w_p = \begin{cases} w_p^m & \text{if } p_{i-1}^x = p_{j-1}^y \\ w_p^s & \text{if } p_{i-1}^x \neq p_{j-1}^y \end{cases}$$

식 1. 대각 방향(일치, 대치) 가중치
Eq. 1. Weights for Diagonal Direction

삽입, 혹은 삭제 단어는 질의문이나 예문에서 해당 단어를 경계로 문맥적 단절의 가능성이 높아 두 문장의 유사성 정도에 기여하지 못하고 오로지 비유사성 정도만을 나타낼 뿐이다. 따라서 식 1에서의 삽입, 삭제에 대한 가중치 $w_d$는 일치에 대한 가중치 $w^m$: $w_s^m$, $w_t^m$, $w_p^m$에 비해 작아야만 한다. 뿐만 아니라 일치 사이에 발생하는 몇몇의 대치 연산은 문장의 구조적 유사성 정도를 크게 해하지 않을 가능성이 높기 때문에 일치에 대한 가중치는 대치에 대한 가중치 $w^s$: $w_s^s$, $w_t^s$, $w_p^s$에 비해 또한 커야 할 것이다. 대치에 대한 가중치는 일치에 대한 가중치에 비해 더 크거나 같게 설정할 수도 있겠지만 이렇게 하면 구조적으로 유사하지 않음에도 불구하고 단순히 문장의 길이가 비슷하여 유사도 점수가 높을 수가 있다. 대치에 대한 가중치가 구조적 유사성에 기여하도록 하기 위해서는 일치 연산의 수보다 작아야만 한다. 이를 반영하기 위해 대치에 대한 가중치는 일치에 대한 가중치보다 낮게 설정한다.

정렬은 유사도 계산과 동시에 저장되는 역추적 정보($Ptr(i,j)$)를 통하여 이루어지는데, 이는 번역 단계에서 대역어 선택에 있어 중요한 역할을 수행한다. 그런데 일치와 대치에 대해 역추적 정보는 모두 대각 방향으로 둘을 구분하지 못한다. NW 알고리즘에서 정렬 관계는 표면어의 일치여부에 따라 단 두 가지 $\alpha$만을 갖기 때문에 일치와 대치 관계의 결정은 두 가지 값에 따르면 된다. 반면에 제안하는 척도에서 $\alpha$의 값은 세 가지 정보에 대한 가중치의 합이므로 $2^3$가지 수가 존재한다. 그런데 표면어, 어간, 품사 정보의 계층 관계로 인하여 4가지 경우는 발생하지 않는다. 따라서 계층관계에 부합하는 4가지 경우만 발생하는데, 정렬 관계를 결정하기 위해 이 4가지 경우를 일치와 대치로 구분하여야 한다. 본 논문에서는 4가지를 다음과 같이 일치와 대치로 구분한다.

- 일치(M): 정렬 관계에 있는 단어의 어간과 품사가 동일한 경우

$$A(i,j) = A(i-1,j-1) + w_s^m + w_t^m + w_p^m$$

$$A(i,j) = A(i-1,j-1) + w_s^s + w_t^m + w_p^m$$

- 대치(S): 표층어와 어간이 일치하지 않는 경우

$$A(i,j) = A(i-1,j-1) + w_s^s + w_t^s + w_p^m$$

$$A(i,j) = A(i-1,j-1) + w_s^s + w_t^s + w_p^s$$

또한 길이 $m$인 질의문과 길이 $n$인 예문에서 하나의 정렬 관계 $r_l(1 \le l \le \max(m,n))$은 일치(M), 대치(S), 삽입(I), 삭제(D)로 정의한다.

(4) a. This annex defines rules
b. This package defines the behavior

이렇게 정의된 유사성 척도를 이용하여 질의 문장 (4-a)에 대해 예제 문장 (4-b)와의 유사성 점수 계산하면 그림 5의 a)와 같이 4가 된다. 또한 역추적 정보 $Ptr(i,j)$를 이용한 정렬 관계는 그림 5의 b)와 같다. 여기서 일치에 대한 가중치는 $w_s^m = w_t^m = w_p^m = 1$로 설정하였으며 대치에 대한 가중치는 $w_s^s = w_t^s = w_p^s = -1$로, 그리고 삽입/삭제에 대한 가중치는 $w_d = -3$으로 설정하였다.

| | | DT this This | NN package package | VB define defines | NN behavi behav |
|---|---|---|---|---|---|
| | 0 | -3 | -6 | -9 | -12 |
| DT this This | -3 | 3 | 0 | -3 | -6 |
| NN annex annex | -6 | 0 | 2 | -1 | -4 |
| VB define defines | -9 | -3 | -1 | 5 | 2 |
| NN rule rules | -12 | -6 | -4 | 2 | 4 |

a) Similarity Matrix

| This | annex | defines | rules |
|---|---|---|---|
| \| | \| | \| | \| |
| M | S | M | S |
| \| | \| | \| | \| |
| This | package | defines | behavior |

b) Alignments

그림 5. 유사 점수와 정렬 관계 계산의 예
Fig. 5. An Example for Similarity Calculation and Alignment Identification

## IV. 실험 및 평가

실험을 위해 국제전기통신연합(ITU)에서 전기 통신 업무의 기술, 운용 및 요금 문제를 연구하여 그 결론을 권고로 공표하는 영어로 된 'ITU-T 권고' 문서들과 이 문서들을 한국정보통신기술협회에서 표준 번호 문서 단위로 번역한 한국어 문서를 이용하였다. 본 논문에서는 'ITU-T' 권고 영어로 문서와 표준 번호 단위로 대응되어 한국어 문서쌍들에 대하여 수작업을 통하여 문장단위 대응 관계를 파악하여 구축한 병렬코퍼스를 사용하였다. 구축한 병렬코퍼스에서 검색 대상이 되는 영어 문장들에 대한 특성은 표 2와 같으며, 기호, 구두점, 및 숫자는 표제어 수에서 제외하였다.

표 2. 실험 코퍼스 특성
Table 2. Characteristics of Experiment Corpus

| 항목 | 개수 |
|---|---|
| 문장 | 710 |
| 토큰 | 10,820 |
| 서로 다른 토큰 | 1,380 |
| 표제어 | 827 |
| 문장당 평균 토큰 | 15.2 |

Brill의 태거[14]로 태깅을 수행하였고, 사용된 품사 태그 집합은 Penn Treebank 태그 집합[15]을 기반으로 한국어의 번역에 영향을 비교적 주지 않는 NNS는 NN으로, VBP와 VBZ은 VB로 통합하여 총 33개 태그를 사용하였다. 표제어 추출(Lemmatization)을 위한 도구는 WordSmith를 사용하였다. 그림 5와 동일하게 가중치는 $w_s^m = w_t^m = w_p^m = 1$, $w_s^s = w_t^s = w_p^s = -1$, $w_d = -3$으로 설정하였다. 알고리즘 평가를 위해 ITU-T 권고 710 문장쌍에서 모든 영어 문장을 한 번씩 질의문으로 사용하였으며 질의문으로 선택된 문장은 예문에서 제외하여 반복적으로 실험하였다. 각 질의문에 대하여 유사성 점수 $S$가 0이상인 것에 대한 검색 성공률과 유사성 순으로 상위 3개(T3)에 대해 검색 정확률을 평가하였다. 표 3에서 문장의 길이가 6 이상 10 이하인 문장 수는 107개였으며 S>0 평가 지표에서 99개 질의 문장에 대해 유사 문장 검색에 성공하였으며 T3 평가 지표에서 95개 질의 문장에 대해 정확한 문장을 검색해냈다. 그리고 전체 질의 문장 710개에 대해 평균적으로 약 73.2%의 정확도를 보였다. 15개 이하의 단어수를 갖는

문장에 대한 검색 정확률은 매우 정확하였으나 21개 이상에 대해서는 매우 부정확했다. 통계량으로서 데이터의 양이 충분하지 않아 단정적으로 이야기하기 어렵지만, 긴 문장에 대한 부정확성은 NW 알고리즘의 특성인 전역적 유사성 비교에 따른 자료 부족 문제를 주요 원인으로 꼽을 수 있을 것이다.

표 3. 유사 문장 검색 정확률
Table 3. Experimental Results

| 단어수 | 5이하 | 6-10 | 11-15 | 16-20 | 21이상 | 전체 |
|---|---|---|---|---|---|---|
| 문장수 | 100 | 107 | 298 | 197 | 8 | 710 |
| S)0 성공률(%) | 100 (100) | 92.5 (99) | 83.2 (248) | 68.0 (134) | 37.5 (3) | 82.3 (584) |
| T3 정확률(%) | 98.0 (98) | 89.8 (95) | 74.8 (223) | 55.3 (103) | 12.5 (1) | 73.2 (520) |

## V. 결론 및 향후 연구 방향

본 논문에서는 전기통신 분야의 제한된 영역의 문장을 사용하여 번역지원 시스템을 위한 유사 예문 검색 알고리즘을 제안하였다. 매우 작은 량의 문장을 사용하였음에도 불구하고 비교적 긍정적인 결과를 얻을 수 있었다. 향후 과제는 제한된 영역의 문장이 아닌 일반 영역에서의 문장에 대해서도 평가가 이루어져야 할 것이다. 또한 해결되어야 할 문제점으로는 긴 문장에 대한 고품질의 번역을 위해서는 NW 알고리즘에서의 전역적 유사성 비교 외에 국부적(Local) 유사성 비교가 동반되어야 할 것으로 생각된다. 이와 더불어 최적의 가중치 집합을 결정하는 문제 또한 많은 연구가 필요할 것이다. 또한 실용화에 있어 무엇보다 우선적으로 해결되어야 검색 속도 문제는 [10][12][16]의 예에서와 같이 검색 대상 예문을 제한하는 방법론이나 적합한 새로운 파일시스템이 고안되어야 할 것이다.

## 참고문헌

[1] D. Wilson and D. Moss, "CAT: a 7090-3600 Computer-Aided Translation," Communications of the ACM, vol. 8, no. 12, pp.777-781, 1965.

[2] L. Bowker, Computer Aided Translation Technology: A Practical Introduction, University of Ottawa Press, 2002.

[3] M. Kay, "The Proper Place of Men and Machines in Language Translation," Research Report CSL-80-11, Xerox Palo Alto Research Center, Palo Alto, Calif., Reprinted in Machine Translation vol. 12, pp.3-33 (1997), 1980.

[4] M. Simard, "Translation Spotting for Translatio Memory," Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond, vol. 3, pp.65-72, 2003.

[5] M. Nagao, "A Framework of a Mechanical Translation between Japanese and English by Analogy Principle," In Artificial and human intelligence, A. Elithorn and R. Banerji (Eds.), Amsterdam: North-Holland, pp.173-180, 1994.

[6] H. L. Somers, "'New Paradigms' in MT: the State of the Play now that the Dust has Settled," In 10th European Summer School in Logic, Language and Information, Workshop on Machine Translation, pp.22-23, 1998.

[7] S. Needleman and D. Wunsch, "A General Method Applicable to the search for similarities in the amino acid sequence of two proteins," Journal of Molecular Biology vol. 48, pp.443-453, 1970.

[8] H. Picht and J. Draskau, Terminology, An Introduction, The Copenhangen School of Economics, Copenhagen, 1985.

[9] J. R. Kohlenberger, The Strongest Tniv Exhaustive Concordance, Zondervan, 2006.

[10] L. Cranias, H. Papageorgiou and S. Piperidis, "A Matching Technique in Example-based Machine Translation," In Proceedings of the 15th International Conference on Computational Linguistics, pp.100-104, 1994.

[11] E. Sumita, "An Example-based machine translation system using DP-matching

between word sequences," Recent Advances in Example-based Machine Translation, Kluwer Academic Publishers, pp.189-209, 2003.

[12] T. Doi, H. Yamamoto and E. Sumita, "Graph-based Retrieval for Example-based Machine Translation Using Edit-distance," Workshop on Example-based Machine Translation, MT Summit X, pp.51-58, 2005.

[13] V. I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals," Soviet Physics - Doklady, Vol. 10 No. 8, pp. 707-710, February 1996, Translated from Doklady Akademii Nauk SSSR, vol. 163 no. 4 pp.845-848, August 1965.

[14] E. Brill, "Some Advances in Transformation-Based Part of Speech Tagging," Proceedings of the Conference of the American Association for Artificial Interlligence, 1994.

[15] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz, "Building a Large Annotated Corpus of English: The Penn Treebank," Computational Linguistics, vol. 19, no. 2, pp.313-330, 1993.

[16] L. Cranias et al., "Clustering: A Technique for Search Space Reduction in Example-Based Machine Translation," Proceedings of International Conference on System, Man, and Cybernetics, Oct. 2-5, pp.1-6, 1994.

## 저 자 소 개

**김 동 주**

2001년: 한양대학교 전자계산학 박사학위 과정 수료.

관심분야: 한국어 형태소 및 구문 분석, 정보검색, 맞춤법 검사, 기계번역.

**김 한 우**

1980년: 한양대학교 전자공학 공학박사.

1981년~현재: 한양대학교 전자컴퓨터공학부 교수

관심분야: 정보처리, 자연언어처리, 기계번역.