문장 길이와 단어 정렬에 기반한 한-영 문장 정렬

임재수[0] 서희철 이상주 임해창
고려대학교 컴퓨터학과
{jslim, hcseo, zoo, rim}@nlp.korea.ac.kr

Korean-English Sentence Alignment Based on Sentence Length and Word Alignment

Jae-Soo Lim[0] Hee-Cheol Seo Sang-Zoo Lee Hae-Chang Rim
Dept. of Computer Science and Engineering, Korea University

요 약

말뭉치를 통한 통계적인 자연 언어 처리에 관한 연구가 다국어 처리 분야에서도 활발히 진행되고 있는 가운데, 본 논문에서는 병렬 말뭉치 구축 및 활용의 기본이 되는 문장 정렬을 위한 효과적인 방법을 제안한다. 먼저, 기존의 문장 길이를 이용한 방법을 한-영 문장 정렬에 적용해 보고, 길이 정보만을 이용했을 때의 한계점을 지적한다. 그리고, 사전과 품사 대응 확률을 이용한 단어 정렬을 통하여, 길이 기반의 정렬 방식이 갖는 문제점을 보완할 수 있는 방법을 제시한다. 실험을 통하여 제안한 방법이 길이에 기반한 방법에 비하여 높은 성능을 나타냄을 알 수 있었다. 또한 한-영 문장 정렬에의 어휘 정보 활용에 있어서 문제가 될 수 있는 요소가 어떤 것들이 있는지 알아본다.

1. 서론

컴퓨터의 성능이 꾸준히 향상됨에 따라, 최근 말뭉치에 의한 통계적인 자연 언어 처리에 관한 연구가 활발히 진행되고 있다. 다국어 처리 환경에서도 이러한 통계적 언어 처리 연구가 병렬 말뭉치를 통하여 이뤄지고 있다. 병렬 말뭉치란 같은 내용에 대하여 둘 이상의 언어로 기술하여 대응시킨 말뭉치를 말한다. 이러한 병렬 말뭉치는 기계 번역, 다국어 정보 검색, 사전 편찬 등의 연구 분야에서 기초적인 언어 자료로써 활용되고 있다.

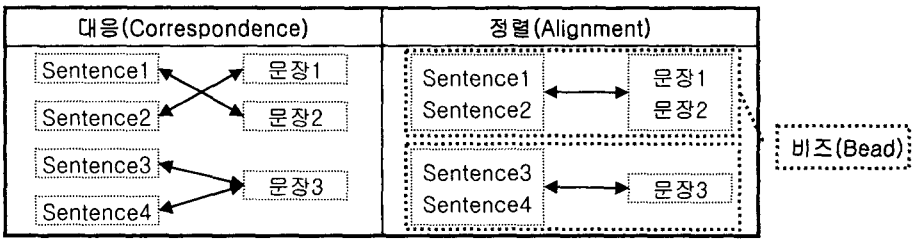
정렬(alignment)은 병렬 말뭉치에서 서로 대응되는 요소들을 찾는 작업을 말한다. 정렬의 대상이 되는 요소들로는 크게 문서에서부터 단락, 문장, 구, 단어 등이 있다. 이 중에서 본 논문에서는 한국어와 영어로 된 양국어(bilingual) 말뭉치에서의 문장 정렬에 관한 연구에 대하여 설명한다. 문장 정렬은 병렬 말뭉치의 구축 및 활용을 위한 가장 선행적이고도 기본적인 작업이라고 할 수 있다.

문장 정렬의 문제에 있어서, 한국어의 한 문장이 반드시 영어의 한 문장과 대응되는 것은 아닌데, 이것은 한국어와 영어간의 언어적 차이, 번역자의 특성 등의 이유로 그렇다. [그림1]은 문장 정렬의 여러 형태들이다. 그림에서 보는 것처럼 정렬의 단위는 하나 이상의 문장이 이루는 세그먼트(segment)이다. 두 문서는 각각 n개의 세그먼트로 분할되고, 각각의 세그먼트들은 순서를 가지고 서로 번역관계로 대응된다. [그림1]의 한국어 문장 K8과 대응되는 영어 세그먼트의 경우에서처럼, 때로는 빈 문장이 하나의 세그먼트를 이루기도 한다[1].

[그림2]는 엄밀한 의미에서의 '대응(correspondence)'과 '정렬'의 차이점을 나타낸 것이다. 그림에서처럼 '대응'은 교차 관계를 허용하며, 각각의 문장들이 모두 다른 쪽 언어의 문장과 하나 이상의 관계를 맺고 있지만, '정렬'은 같은 수의 세그먼트가 순서대로 관계를 맺고 있다. 이러한 '대응'의 문제는 서로 다른 언어로 된 두 문장에서 대응되는 단어나 구를 찾는 어휘 정렬의 문제에서 빈번히 나타난다. 특히, 한국어와 영어 같이 어순이 판이하게 다른 경우에는 더욱 그러하다. 그러나, 문장 정렬에서 교차 대응은 극히 드물어서, 보통 '정렬'의 개념으로 문제를 풀어나간다. 정렬에서 대응된 두 세그먼트를 본 논문에서는 비즈(bead)라는 용어를 사용한다[2].

정렬	영어	한국어
1 : 1	E1 The most common type of alignment.	K1 정렬의 가장 일반적인 형태.
2 : 1	E2 Now a different case. E3 Two sentences are aligned to one	K2 이번에는 다른 경우로써, 두 문장이 하나의 문장으로 정렬되었습니다.
1 : 3	E4 Various configurations are possible, with different numbers of sentences involved, depending on the translators habits.	K3 다양한 배치가 가능합니다. K4 문장의 개수도 다양합니다. K5 이것은 번역자의 습관에서 비롯된 것입니다.
2 : 2	E5 Not very frequently, alignments can be rather complex. E6 For example, they can be 2 to 3, or 3 to 4	K6 때로는 복잡한 정렬도 존재하는데, 예를 들면 2 대 3, 3 대 4 등이 있습니다. K7 다행히도 이런 것은 매우 드뭅니다.
0 : 1		K8 어떤 경우에는 번역이 빠져있기도 합니다.
1 : 0	E7 Of course, the translators can make mstakes.	

[그림1] 문장 정렬의 예



[그림2] '대응'과 '정렬'

2. 관련 연구

문장 정렬에 대한 연구는 먼저 문장의 길이에 기반한 통계적 접근 방법으로부터 시도되었다. 이것은 "긴 문장은 긴 문장으로, 짧은 문장은 짧은 문장으로 번역할 것이다."라는 단순하고도 명료한 직관에서부터 출발하였다. 문장의 길이는 단어의 수[2] 또는 문자의 수[3]를 이용하여 계산하였다. 이러한 방법은 실제로 별다른 언어적 지식을 필요로 하지 않으면서도 높은 성능을 나타냈으며, 무엇보다 빠른 시스템으로써 대량의 말뭉치를 처리할 수 있도록 하는데 기여하였다. 그 중에서도 본 논문에서 실험한 시스템의 기반이 되는 W. A. Gale과 K. W. Church의 방법[3]에 대해서는 다음 2.1 절에서 자세히 다룬다.

이에 반해, 문장을 이루는 단어들, 즉 어휘 정보를 이용하여 문장 간의 유사도를 측정하고, 이를 바탕으로 정렬을 시도한 연구도 진행되었다. S. F. Chen은 두 문장 내 단어들이 오직 1 대 1로 대응되도록 제약한 간단한 번역 모델을 이용하여 문장 간의 유사도를 측정하고, 이를 통하여 정렬을 수행하였으며[4], M. Kay와 M. Roscheisen은 별도의 어휘 데이터 혹은 그것을 얻기 위한 사전 작업이 없이, 병렬 말뭉치 내부의 어휘 정보만을 이용하여 정렬을 하였다[5].

그밖에도, 불어와 영어 같이 동족의 언어들 간에는 문자 체계와 단어가 서로 유사한 점(cognate)을 이용하여 정렬을 수행하기도 하였으며[1], 단지 품사 정보만을 이용한 통계적 정렬 방법이 제안되기도 하였다[6].

한국어 및 영어의 문장 정렬에 관한 연구로는 양주일 외 논문[7]이 있다. 이들은 웹으로부터 수집한 문서를 대상으로 HTML 태그 정보를 이용하여 번역문서 쌍을 판별하였고, 이 문서를 바탕으로 Gale과 Church의 문장의 길이에 기반한 방법을 적용하였다. 이 때, 문장의 길이는 영어의 관사와 전치사를 제외한 단어의 개수와 한국어의 어절의 개수를 이용하여 측정하였다.

2.1. Gale과 Church의 방법[3, 8]

병렬 말뭉치에서 원본 문서(S)와 대상 문서(T)가 있을 때, 확률적으로 최대가 되게 하는 정렬(A)은 다음의 수식 1)과 같이 표현된다.

$$\arg\max_A \Pr(A|S,T) = \arg\max_A \Pr(A,S,T)$$

$$= \arg\min_A dist(A,S,T) \qquad 1)$$

$$dist(A,S,T) \approx \sum_k cost(B_k)$$

여기서 Pr(A|S,T)는 원본 문서와 대상 문서가 주어졌을 때 정렬의 확률을 나타내고, dist(A,S,T)는 원본 문서와 대상 문서에 정렬을 할당했을 때의 문서간 거리를 나타낸다. cost(B_k)는 하나의 정렬 비즈 B_k에 대한 할당 비용을 나타낸다. 즉, 원본과 대상 문서에 대한 정렬이 주어졌을 때, 정렬에 포함된 모든 정렬 비즈의 할당 비용을 합한 것이 바로 그 정렬을 할당했을 때의 문서간 거리가 된다.

정렬 비즈의 할당 비용을 계산하기 위해서 두 문서의 길이가 사용되는데, 아래 수식 2)와 같다.

$$cost(B_k) = -\log(\Pr(cat(B_k)) \times pnorm(\delta))$$

$$\delta = \frac{|l_t - l_s c|}{\sqrt{l_s s^2}} \qquad 2)$$

$$pnorm(\delta) = 2\left(1 - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\delta} e^{-z^2/2} dz\right)$$

여기서 cat(B_k)는 정렬 비즈 B_k의 범주(1 대 0, 1 대 1, 1 대 2 등)를 반환하는 함수이다. 즉 Pr(cat(B_k))는 정렬 비즈의 범주 확률이 된다. δ는 두 세그먼트간 길이의 차이 값을 정규화한 값이며, l_s와 l_t는 각각 원본 및 대상 언어 세그먼트의 길이를 나타내고, c와 s^2는 세그먼트간 길이의 평균 비율과 이 비율을 고려한 길이 차의 분산을 나타낸다. 마지막으로 pnorm(δ)는 정규분포에서 δ의 확률 밀도이며, 이 값은 0에서 1 사이의 값을 가지는 두 세그먼트의 유사도라고 볼 수 있다. 여기서 Gale과 Church는 두 세그먼트의 길이는 서로 밀접한 관련이 있으며, 길이의 차이는 정규분포를 따를 것이라는 가정을 하였다. 따라서 평균 길이 비율을 고려한 길이의 차이가 0이 되면 두 세그먼트의 유사도가 최대인 1의 값을 가지게 된다[1].

원본 언어와 대상 언어 단락이 주어졌을 때, 두 단락의 거리를 최소로 하는 정렬을 찾는 방법은 아래 수식 3)과 같은 동적 프로그래밍(dynamic programming) 기법을 사용한다.

$$D(i,j) = \min \begin{cases} D(i,j-1) + d(0,0;t_j,0) & \rightarrow 0;1 \\ D(i-1,j) + d(s_i,0;0,0) & \rightarrow 1;0 \\ D(i-1,j-1) + d(s_i,0;t_j,0) & \rightarrow 1;1 \\ D(i-1,j-2) + d(s_i,0;t_j,t_{j-1}) & \rightarrow 1;2 \\ D(i-2,j-1) + d(s_i,s_{i-1};t_j,0) & \rightarrow 2;1 \\ D(i-2,j-2) + d(s_i,s_{i-1};t_j,t_{j-1}) & \rightarrow 2;2 \end{cases} \qquad 3)$$

여기서 D(i;j)는 각각 i 및 j 개의 문장을 갖는 두 단락간 최소 거리이며, d(s;t)는 정렬 비즈의 할당 비용을 나타낸다. Gale과 Church는 총 여섯 가지의 정렬 비즈 범주를 사용한다.

3. 문장 길이에 의한 방법

본 논문에서는 기본적으로 Gale과 Church의 방법을 따른다. 문장의 길이를 측정하기 위한 방법으로 문자, 단어, 형태소, 내용어 등을 단위로 사용한다.

• **문자 길이 기반** : 문장에 포함된 문자의 수, 즉 바이트(byte) 길이를 이용한 측정 방법은 Gale과 Church가 사용했던 방법을 그대로 적용한다. 문자를 기준으로 하면, 한국어와 영어 세그먼트의 평균 길이의 비율은, 영어를 기준으로 했을 때 약 0.85 정도로 나타나는데, 이 값은 Gale과 Church가 한 것처럼 1.0으로 간주한다. 이렇게 했을 때 한국어와 영어 단락의 길이 차는 약 51의 분산을 갖는다. 이 두 값은 수식 2)에서의 c와 s^2에 해당한다.

• **단어-어절 길이 기반** : 영어 문장의 단어와 한국어 문장의 어절 수를 이용하여 길이를 측정한다. 이때의 평균 길이의 비율은 약 0.6 정도로 나타난다. 이 값은 비교적 큰 차이를 보이는 것으로 판단하여 문서간 길이 차를 구할 때 정확히 적용한다. 이때에는 단락의 길이 차에 대한 분산이 약 2.88 정도로 작게 나타난다. 이것은 단어의 수를 이용한다는 점에서 단위가 작고, 평균 길이의 비율을 정확히 적용했기 때문에, 바이트 단위에 비해 비교적 작은 값을 가지게 된다.

• **단어-형태소 길이 기반** : 영어 문장의 단어와 한국어 문장의 형태소의 수를 이용하여 길이를 측정한다. 이때의 평균 길이의 비율은 단어와 어절을 사용할 때와는 반대로 약 1.34로 나타난다. 이 값도 비교적 큰 차이를 보이는 것으로 판단하여 역시 문서간 길이 차를 구할 때 정확히 적용한다. 이때의 단락 길이 차에 대한 분산은 약 3.03으로서 단어-어절 길이 기반 방법과 비슷한 수치를 보인다.

• **내용어 길이 기반** : 영어 문장에서 기호 및 관사와 한국어와 의미적 대응 관계가 없다고 판단되는 6 가지 품사[2]를 갖는 단어들을 제외한 나머지 단어들의 수와, 한국어 문장에서는 22 개의 실질 형태소[3]의 수를 이용하여 길이를 측정한다. 이렇게 할 때의 평균 길이의 비율은 약 0.84 정도로 나타났으며, 이 값은 문자 길이 기반 방법과 같이 1.0으로 간주한다. 단락 길이 차에 대한 분산은 약 6.67의 값을 가진다. 이때 사용한 자동 품사 태거는 영어의 경우 E. Brill의 태거[9]이며, 한국어의 경우 김진동 외의 태거[10]이다.

위 네 가지 방법으로 측정된 문서의 길이를 이용하여, Gale과 Church가 사용한 여섯 가지 정렬 비즈 범주에 몇 가지를 추가하여 적용한다. 추가한 정렬 비즈 범주는 1 대 3, 3대 1, 2 대 3, 3 대 2, 3 대 3으로 다섯 가지이다. 이것은 한국어와 영어의 경우 영어권 언어들 간의 정렬에 비해 비교적 복잡한 형태를 띄는 경우가 많아, 최적의 정렬 해를 위한 탐색 공간을 늘리기 위해서이다.

문장 길이에 의한 방법은 근본적인 한계를 지니고 있다. 두 단락의 거리를 계산함에 있어서 길이의 차이를 최소로 하는 정렬 해를 찾기 때문에, 어휘적인 관계를 고려한다면 명백한 정렬 해를 얻을 수 있는 부분에서 그

1) 수식 2)는 원본 언어의 세그먼트를 기준으로 한 '단방향 거리'라고 볼 수 있으며, 두 번째 수식의 분모로 쓰인 l_s는 0의 값을 가질 경우 심각한 문제를 발생한다. [3]의 부록으로 포함된 프로그램 소스 코드에서는 l_s를 대신하여 '$(l_s+l_t/c)/2$'를 사용하였고, 이를 '양방향 거리'라고 정의한다.

2) PennTreebank의 품사 집합에서 기호 태그를 제외한 36 개의 태그 중에서, CD, LS, POS, RP, SYM, TO를 제외하고, DT 태그 중에서 a, an, the 세 가지 관사만을 제거한다.

3) 김진동 외[10]의 태거에서 사용하는 51개의 품사 집합 중에서, 체언, 용언, 독립언, 수식언에 해당하는 22개의 실질 형태소만을 사용한다.

맹점을 보인다. [그림3]은 그러한 단점을 잘 나타내 주는 예이다.

영어	E1 " Trust Me , ma'am , " said the dwarf .
	E2 " Please trust me and let take care of it . "
한국어	K1 「 아주머니, 저를 믿어 보세요. 」
	K2 난장이가 말했다.
	K3 「 저를 믿고 맡겨 주십시오. 」

[그림3] 길이에 의한 방법의 한계 예

여기서, 영어의 두 문장과 한국어의 세 문장이 정렬되어야 하므로, 확률을 최대로 하는 정렬은, 영어는 각각 한 문장씩 두 세그먼트를 이뤄야 하며, 한국어는 두 문장이 하나의 세그먼트를 이루고 나머지 한 문장이 다른 세그먼트를 이뤄야 한다. 길이 정보만을 이용할 경우 K2와 K3가 하나의 세그먼트를 이뤄 E2와 대응되는 것이 문서간 거리를 최소로 하는 정렬이 된다. 그러나, K1과 K2가 하나의 세그먼트를 이뤄 E1과 대응되는 것이 바른 정렬이다. '난장이가'와 'dwarf', '말했다'와 'said'가 서로 대응된다는 어휘 정보를 이용하지 않고 길이 정보만을 이용해서는 바른 정렬을 얻을 수 없다.

4. 단어 정렬에 의한 방법

어휘 정보를 이용하여 문장간의 유사도를 구하기 위해서는 우선, 단어 혹은 구 단위의 대응 정보가 필요하다. 본 논문에서는 대역 사전과 품사 대응 확률을 이용한 단어 대 단어 정렬 모델을 통해 문장간 유사도를 측정한다. 앞서 설명한 길이에 의한 문장간 거리에, 단어 정렬을 통한 문장 유사도를 가중치로 하여, 길이에 의한 방법이 지닌 오류를 정정하고자 한다.

4.1. 단어 정렬

정렬의 대상이 되는 단어는, 앞서 길이에 의한 방법에서 설명했던 영어의 기호, 관사 및 6 가지 품사를 제외한 30 가지의 품사와, 한국어의 22 가지 실질 형태소로 선택한 내용어이다. 1 대 1 대응만을 고려하며, 실험 결과, 영어를 기준으로 평균 41%, 한국어를 기준으로 평균 50%의 단어가 정렬된다. 이렇게 적용 비율이 낮은 것은 한국어와 영어의 언어 구조가 매우 다르기 때문이며, 보다 정확한 어휘 정렬과 통계적 대응 확률을 구하기 위해서는 구 단위로 확장되어야 할 것이다. 그러나 본 논문은 길이에 기반한 방법의 오류를 보정하는 데 초점을 두고 있으므로, 간단한 1 대 1 대응만으로도 충분한 정보로서 활용할 수 있다고 본다.

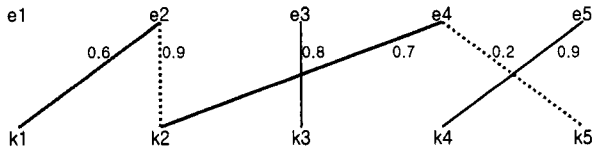
단어의 대응 정보는 대역어 사전을 이용한다. 전자 사전으로부터 한국어, 영어 모두 단일 어절로만 구성된 단어 쌍을 추출하여 대역어 사전을 구축한다. 영어의 경우 원형을 복원하여 정규화하고, 한국어의 경우 실질 형태소만을 추출한다. 이렇게 구성된 영-한 단어 쌍의 개수는 229,010 개이며, 영어 단어는 69,807 개, 한국어 단어는 55,062 개이다. 영어 한 단어에 대한 평균 대역어의 개수는 약 3.28 개이다.

영어 단어에 대해 한국어 대응 단어 후보가 여러 개일 경우, 품사 대역 확률을 이용하여 전체 문장의 품사 대역 확률을 최대로 하는 단어 정렬을 선택하게 된다. 4)는 최대 확률을 갖는 단어 정렬을 표현한 수식이다.

$$\arg\max_{\alpha} \prod_{\beta_i \in \alpha} \Pr(\beta_i) \qquad 4)$$

$$\Pr(\beta_i) = \Pr(pos_k | pos_e)$$

여기서 α는 단어 정렬을 나타내고, β_i는 α에 포함된 1 대 1 대응 단어 정렬 비즈를 나타낸다. pos_e와 pos_k는 각각 영어 및 한국어 품사를 나타내고, $\Pr(pos_k|pos_e)$는 영어 품사에 대한 한국어 품사의 조건부 발생 확률이 된다. 예를 들어, 아래 [그림4]과 같은 중의성을 갖는 단어 정렬의 경우,

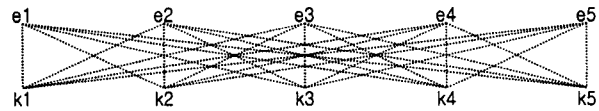


[그림4] 중의성을 갖는 단어 정렬의 예

영어 단어 e2의 대역어 후보는 k1과 k2이고, e4의 대역어 후보는 k2와 k5이다. e3과 e5는 각각 k3과 k4에 대응되고 있으며, e1은 가능한 대역어 후보가 없다. 이 경우 가능한 단어 정렬은 {(e1;·), (e2;k1), (e3;k3), (e4;k2), (e5;k4), (·;k5)}, {(e1;·), (e2;k1), (e3;k3), (e4;k5), (e5;k4), (·;k2)}, {(e1;·), (e2;k2), (e3;k3), (e4;k5), (e5;k4), (·;k1)}로 모두 세 가지이다. 이 중 첫 번째 정렬이 0.3024로 최대 확률을 가지는 정렬 해이다. 단어의 대역 확률을 추정하기 위해서는 대량의 병렬 말뭉치가 요구되지만, 품사 대응 확률의 경우 소량의 병렬 말뭉치로부터 얻을 수 있어, 자료부족(data sparseness) 문제가 다소 적다는 장점이 있다.

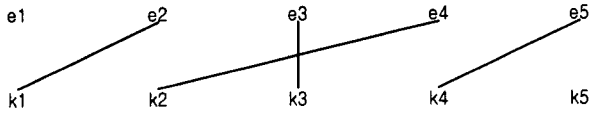
품사 대응 확률은 다음 두 단계를 통해 추정한다.

• **초기 단계** : 정렬된 한-영 문장 쌍으로부터 가능한 모든 단어 쌍의 빈도를 얻고, 이 값을 MLE (maximum likelihood estimation)로 추정한다. 아래 [그림5]는 초기 단계에 사용되는 단어 쌍을 나타낸다.



[그림5] 초기 단계

• **반복 단계** : 전 단계에서 추정한 확률을 이용하여 단어 정렬을 먼저 수행한다. 그리고, 최대 확률을 갖는 정렬의 단어 쌍만을 이용하여 MLE로 재 추정한 다음, 이전 확률을 갱신한다. 아래 [그림6]은 반복 단계에 사용되는 단어 쌍을 나타낸다. 전체 품사 대응 확률의 엔트로피(entropy)가 수렴할 때 반복을 종료한다.



[그림6] 반복 단계

4.2. 단어 정렬에 의한 문장간 유사도

단어 정렬 정보를 이용한 문장간 유사도 측정에는 두 가지 방법을 사용한다. 하나는 단어의 어휘 자체를 이용한 방법이며, 다른 하나는 단어의 품사를 이용한 방법이다.

4.2.1 어휘 유사도

문장을 구성하는 단어들을 출현 빈도에 의해 가중치를 부여하여 문장간 유사도를 측정한다. 출현 빈도가 높은 단어는 인접한 문장에 나타날 가능성이 크며, 이로 인해 여러 문장으로 구성된 세그먼트간의 유사도를 고려해야 하는 문장 정렬의 문제에서 혼란을 야기할 수 있다. 따라서, 출현 빈도가 높은 단어들에 대하여 상대적으로 낮은 가중치를 부여하기 위하여, 정보검색 모델의 벡터 모델에서 사용하는 IDF(inverse document frequency)와 비슷한 개념으로, ISF(inverse sentence frequency)를 이용하여 가중치를 부여한다. 수식 5)는 ISF 가중치를 구하는 수식이다.

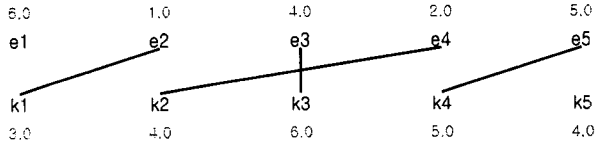
$$ISF(w) = \ln \frac{N}{n_w} \qquad 5)$$

여기서 w는 영어 및 한국어 단어를 나타내고, n_w는 단어가 출현한 문장의 수이며, N은 전체 문장의 수이다. 출현 빈도가 높은 단어들을 구분하는 것이 목적이므로, 실험에서는 6.0을 상한 임계값으로 둔다. 이렇게 ISF를 이용한 가중치 부여 방법은, 특별히 병렬 말뭉치가 아니라 한국어 및 영어 각각의 단일어 원시 말뭉치로부터 빈도에 의해 손쉽게 얻을 수 있다는 장점이 있다.

ISF를 이용한 문장간 유사도는 아래 수식 6)과 같다.

$$sim_{word}(B_k) = 2\frac{MR(E)\times MR(K)}{MR(E)+MR(K)}$$
$$MR(S) = \frac{\sum_{w_i \in S, w_i \in match} ISF(w_i)}{\sum_{w_i \in S} ISF(w_i)} \qquad 6)$$

여기서 E와 K는 각각 영어 및 한국어 문장을 나타내고, sim_{word}는 정렬 비즈 B_k에서의 두 문장간 유사도이며, MR(S)는 각각 문장의 대응률(match ratio)이다. S는 문장을 구성하는 모든 단어의 집합을 나타내고, match는 문장에서 정렬을 통하여 대응을 이룬 단어의 집합이다. sim_{word}는 MR 값의 조화평균으로 표현되었는데, 두 문장 모두에서 MR 값이 높아야만 전체 유사도가 높은 값을 가지게 된다. 이것은 E와 K가 여러 문장으로 구성될 수 있으며, 한 쪽 문장만이 높은 MR 값을 가질 경우, 부분적으로 일치한 경우일 가능성이 크기 때문에, 이 경우에 높은 유사도를 갖지 못하도록 하기 위해서이다. 아래 [그림7]과 수식 7)은 단어 정렬 결과와 ISF 값을 이용하여 유사도를 계산하는 예이다.



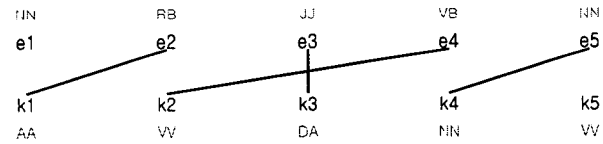
[그림7] 단어 정렬과 ISF 값의 예

$$MR(E) = \frac{1.0+4.0+2.0+5.0}{6.0+1.0+4.0+2.0+5.0} \fallingdotseq 0.6667$$
$$MR(K) = \frac{3.0+4.0+6.0+5.0}{3.0+4.0+6.0+5.0+4.0} \fallingdotseq 0.8182 \qquad 7)$$
$$sim_{word}(B_k) = 2\frac{0.6667\times 0.8182}{0.6667+0.8182} \fallingdotseq 0.7347$$

4.2.2 품사 유사도

영어 문장의 품사 분포를 한국어 품사 분포 벡터로 변환하여 한국어 문장의 품사 분포 벡터와의 유사도를 측정한다. 영어 문장을 한국어 품사 분포 벡터로 변환하는 방법은, 우선 단어 정렬을 수행한 다음, 대응된 단어의 경우 한국어 단어의 품사를 이용하여 빈도 1의 가중치를 부여하고, 대응되지 않는 단어의 경우 위 4.1절에서 설명한 품사 대응 확률을 이용하여 한국어의 모든 품사들에 확률 값으로 가중치를 부여한다.

아래 [그림8]은 한국어 품사 분포 벡터를 구성하기 위한 단어 정렬과 품사의 예이다.



[그림8] 단어 정렬과 품사의 예

영어의 전체 품사는 NN, RB, JJ, VB로 네 가지이며, 한국어의 전체 품사는 NN, AA, DA, VV로 네 가지라고 가정하자. 먼저 한국어 문장의 품사 분포 벡터는 VV 태그를 갖는 단어가 k2, k5로 두 개이고, 나머지 NN, AA, DA는 각각 한 번씩 나타났으므로 <1, 1, 1, 2>이다. 다음 영어 문장의 품사 분포 벡터의 경우, 정렬된 단어 e2, e3, e4, e5에 의해 <1, 1, 1, 1>이 된다. 정렬되지 않은 단어 e1의 품사 NN의 한국어 품사에 대한 확률 분포가 <0.6, 0.1, 0.1, 0.2>라고 가정하면, 이 값을 먼저 정렬된 단어들에 의해 구한 벡터 <1, 1, 1, 1>에 더한다. 이렇게 해서 최종적으로 구성된 영어 문장의 한국어 품사 분포 벡터는 <1.6, 1.1, 1.1, 1.2>가 된다. 이렇게 구성된 두 벡터의 유사도가 바로 문장간 유사도이다.

벡터간 유사도 계산은 평균에 대한 총 상대 엔트로피(total KL divergence to the mean)[11]를 이용하여 계산하였다. 수식 8)은 유사도 계산을 위한 수식이다.

$$sim_{pos}(B_k) = e^{-A(\vec{E},\vec{K})}$$

$$A(\vec{E},\vec{K}) = 2\ln 2 + \sum_{w_i \in both} \left\{ \vec{E}(w_i) \ln \frac{\vec{E}(w_i)}{\vec{E}(w_i)+\vec{K}(w_i)} + \vec{K}(w_i) \ln \frac{\vec{K}(w_i)}{\vec{E}(w_i)+\vec{K}(w_i)} \right\} \quad 8)$$

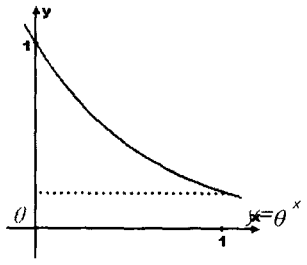
여기서 $\vec{E}$와 $\vec{K}$는 각각 영어 및 한국어 문장의 벡터 요소들의 합이 1이 되도록 정규화한 한국어 품사 분포 벡터이고, sim_{pos}는 정렬 비즈 B_k에서의 두 문장간 유사도이며, $A(\vec{E},\vec{K})$는 두 확률 분포 벡터의 평균에 대한 총 상대 엔트로피이다. 품사 분포 벡터는 단어 정렬에서 대역어가 없는 단어에 대해서도 추정치를 이용하여 유사도를 계산할 수 있다.

4.3. 길이에 의한 방법과의 결합

앞서 3장에서 설명한 길이에 의한 방법과 4.2절에서 설명한 단어 정렬에 의한 문장간 유사도를 함께 이용하는 방법은 여러 가지 형태가 가능하겠지만, 본 논문에서는 길이에 의해 얻은 문장간 거리에, 단어 정렬에 의한 문장간 유사도를 가중치로 하여 최종적으로 문장간 거리를 계산하는 방식을 택한다. 길이에 의한 방법은 대부분의 경우 잘 적용되지만, 앞서 3장에서 지적했던 것처럼 길이만을 이용해서 측정한 문장간 유사도에는 한계점이 있다. 따라서, 길이에 의한 방법의 견고함에 단어 정렬을 이용하여 세밀함을 더하고자 한다. 아래 수식 9)는 문장 길이와 단어 정렬을 이용한 정렬 비즈 할당 비용을 나타낸다.

$$cost(B_k) = cost_{length}(B_k) \times \theta^{sim_{(word|pos)}(B_k)} \quad 9)$$

여기서 $cost_{length}$는 길이에 의한 정렬 비즈 할당 비용이고, $sim_{(word|pos)}$는 앞서 4.2절에서 설명한 단어 정렬을 이용한 두 가지 유사도이며, θ는 0과 1사이의 값을 가지는 가중치 파라미터이다. [그림9]는 θ에 따른 가중치 값의 변화 곡선이다. $sim_{(word|pos)}$가 0부터 1사이의 값을 가지므로, 단어 정렬에 의한 유사도가 1이 되었을 때 최대 θ만큼의 가중치를 길이에 의한 정렬 비즈 할당 비용에 곱하게 된다. 만약, 단어 정렬 결과 대응되는 단어가 하나도 없어, 유사도 값이 0이 되더라도, 길이에 의해 측정한 정렬 비즈 할당 비용이 그대로 적용된다. θ를 낮은 값으로 설정할 수록 단어 정렬에 의한 유사도를 더욱 신뢰하는 것이 된다.



[그림9] θ에 따른 가중치 곡선

5. 실험 및 평가

5.1. 실험 환경 및 평가 척도

실험을 위해 사용한 말뭉치는 21세기 세종계획[4]을 통해 1999년과 2000년에 개발된 병렬말뭉치이다[12, 13]. 장르는 소설, 비소설, 기사, 보고서, 연설문, 교과서 등으로 비교적 일반적인 문서들이다. 총 40 개의 문서 중에서 오류가 많은 9 개의 문서를 제외하고, 수작업으로 오류를 정정한 31 개의 문서를 대상으로 하였다. 단락 구분이 너무 세밀한 부분을 하나로 합쳐서 모두 1 대 1로 대응되도록 하고, 이렇게 구성된 단락쌍은 모두 7,659 개이다. 포함된 문장 정렬 비즈는 모두 20,656 개이며, [표1]은 정렬 비즈의 범주별 분포이다.

[표1] 정렬 범주별 분포 (영 ; 한)

정렬 범주	1 ; 1	1 ; 0	0 ; 1	2 ; 1	1 ; 2	2 ; 2
개수	18625	26	37	524	1236	25
정렬 범주	**1 ; 3**	**3 ; 1**	**2 ; 3**	**3 ; 2**	**3 ; 3**	**그 외**
개수	79	38	18	5	5	38

평가를 위해 재현율(recall), 정확도(precision), F값을 측정한다. 수식 10)은 각각을 구하기 위한 식이다.

$$\text{재현율} = \frac{(\text{맞은 정렬 비즈의 개수})}{(\text{말뭉치의 전체 정렬 비즈의 개수})}$$

$$\text{정확도} = \frac{(\text{맞은 정렬 비즈의 개수})}{(\text{시스템이 출력한 전체 정렬 비즈의 개수})} \quad 10)$$

$$F = 2\frac{\text{재현율} \times \text{정확도}}{\text{재현율} + \text{정확도}}$$

학습 및 평가를 위해, 전체 말뭉치를 크기가 같은 5 개의 작은 말뭉치로 분할한다. 4 개의 말뭉치로부터 파라미터를 얻고, 나머지 하나의 말뭉치에 적용하여 각각 재현율, 정확도, F값을 구했다. 이러한 작업을 5 개의 작은 말뭉치에 모두 적용하여 평균을 구하였다.

5.2. 실험 결과

문장 길이를 이용한 네 가지 방법을 적용한 결과는 아래 [표2]와 같다.

[표2] 문장 길이를 이용한 방법의 결과

	재현율(%)	정확도(%)	F값	표준편차
L/B	95.85	95.41	**95.63**	0.21
L/W	95.70	95.33	95.51	0.21
L/M	95.58	95.64	95.61	0.42
L/C	95.15	94.71	94.93	0.35

여기서 'L/B'는 바이트 길이, 'L/W'는 단어 길이, 'L/M'은 영어 단어와 한국어 형태소를 이용한 길이, 그리고 'L/C'는 영어의 기호, 관사 및 6 가지 품사를 제외한 단어와, 한국어의 실질 형태소의 길이를 이용한 방법을 나타낸다. 대부분 비슷한 성능을 보이고 있으며, 'L/C'의 경우 상대적으로 낮은 결과를 보인다. '표준편차'는 5 개의 분할된 말뭉치에 대하여 F값의 평균을 이용하여 얻은 F값

4) http://www.sejong.or.kr/

의 표준편차이다. 높지 않은 표준편차로 보아 길이에 의한 방법은 학습에 대해 안정적인 성능을 보임을 알 수 있다.

단어 정렬에 의한 문장간 유사도만을 이용하여 실험한 결과는 [표3]과 같다.

[표3] 단어 정렬을 이용한 방법의 결과

	재현율(%)	정확도(%)	F값	표준편차
ISF	89.16	92.21	90.66	0.40
POS	92.71	92.26	**92.49**	0.48

여기서 'ISF'는 ISF를 이용한 방법이며, 'POS'는 한국어 품사 분포 벡터를 이용한 방법을 나타낸다. 이 결과는 수식 2)의 pnorm을 대신하여 수식 6)의 sim_{isf} 혹은 수식 8)의 sim_{pos}를 이용한 것이다. 앞서 설명한 길이에 의한 방법에 비해 약 3~5% 정도 낮은 결과를 보인다.

문장 길이와 단어 정렬을 통한 문장간 유사도를 결합한 결과는 [표4]와 같다. 여기서 θ의 값은 0.1부터 0.9까지 0.1씩 변화시키면서 실험하였으며, 표의 값은 그 중 최대의 성능을 보인 값이다. '증감'은 [표2]의 문장 길이를 이용한 결과를 기준으로 한 F값의 5 개의 분할된 말뭉치에서의 평균 변화량을 나타낸다. 'T'는 두 개의 표본, 즉 길이에 의한 방법과 길이 및 단어 정렬을 결합한 방법의 5 개의 분할된 말뭉치에서의 결과가 같은 평균값을 갖는 두 개의 같은 모집단에서 추출되었을 확률을 t-검정을 통하여 얻은 값이다.

길이에 의한 방법 네 가지와 단어 정렬에 의한 유사도 두 가지를 결합할 수 있는 총 여덟 가지 방법 중에서 'L/W&ISF'가 F값에 있어서 가장 좋은 성능을 보인다. '증감'의 경우 모두 양의 값을 가지는 것으로 보아, 단어 정렬에 의한 유사도가 길이에 의한 방법의 오류를 보완하고 있음을 알 수 있다. 'T'의 값은 다섯 개의 분할된 말뭉치에서 [표2]의 F값과 [표4]의 F값이 서로 같은 평균을 갖는 모집단으로부터 추출되었을 확률인데, 'ISF'와 결합한 방법의 경우 대부분 낮은 값을 보이고 있어서 성능 향상에 대해 신뢰할 수 있다고 본다. 그러나, 'POS'와 결합한 방법의 경우 'L/W&POS'를 제외한 다른 결합의 경우 의미 있는 성능 향상을 보이지 못한다.

'ISF'와 결합한 방법이 'POS'와 결합한 방법에 비해 전체적으로 높은 성능을 보이는데, 이것은 [표3]에서 'ISF'와 'POS' 각각을 비교했을 때 'POS'가 나은 성능을 보이는 것과는 반대의 결과이다. 단어 정렬이 되지 않을 경우, 'ISF'가 0의 유사도를 가지는 반면, 'POS'는 단어 정렬이 되지 않더라도, 품사 대역 확률을 이용하기 때문에 결코 0의 유사도를 갖지 않는다. 따라서, 단락간 거리를 계산할 때 단어 정렬만을 이용할 경우에는 'ISF'는 0의 유사도로 인해 최적의 정렬해 탐색에 실패할 경우가 상대적으로 많아 'POS'에 비해 낮은 성능을 보인다. 하지만 결합에 의한 방법은 'ISF'만을 이용한 유사도의 단점과 길이만을 이용한 방법의 단점을 서로 보완하고 있다.

5.3. 오류 분석

'L/B' 방법에 의해 바르게 정렬되었으나, 'L/B&ISF' 방법에서는 오류를 나타내는 총 42 개의 단락을 추출하여, 수작업으로 분석하고 분류한 결과, 몇 가지 대표적인 오류의 유형을 발견하였다.

- **대역어 사전 오류** : 대역어 후보가 있음에도 사전에 등록되어 있지 않아서 단어 정렬이 불가능한 경우이다. 이러한 오류는 사전 정보만 추가함으로써 보완이 가능하다.
- **의역으로 인한 오류** : 이것은 번역자의 특성에 따른 것인데, 번역시 문장의 변형이 심하게 일어나서, 단어 정렬이 제대로 이뤄지지 않는 경우이다. 이러한 오류는 의미 분석, 혹은 그 이상의 복잡한 언어 처리를 요구하는 것으로, 본 연구에서는 다루기 힘든 유형에 속한다.
- **구 단위 대응 오류** : 의역 문제에 있어서 대응 관계를 찾지 못하는 것과는 달리, 영어와 한국어의 여러 단어(구)간 대응 관계가 명백하지만, 1 대 1 대응 제약으로 인해 단어 정렬을 수행할 수 없는 경우이다.
- **전처리 상의 오류** : 품사 부착, 사전 탐색을 위한 단어 정규화 등 정렬 시스템의 하위 구성 요소인 전처리기의 오류가 전파되어 생기는 오류이다. 이러한 오류는 전처리 시스템의 성능 향상에 따라 해결될 문제로서, 본 연구에서는 다루지 않는다.
- **대명사에 의한 오류** : 번역시 문맥에 의해 단어가 대명사로 대치되는 경우, 혹은 그 반대의 경우에 단어 정렬을 할 수 없어서 생기는 오류이다. 이러한 오류도 복잡한 언어 처리를 요구하는 유형이라 할 수 있다.
- **고유명사의 음차 복원 오류** : 사전에 등록되어 있지 않은 고유명사의 경우에 단어 정렬을 수행할 수 없게 된다. 고유명사는 정렬에 있어서 매우 중요한 단서가 될 수 있음에도, 이를 처리하지 못해 오류가 발생하게 된다.
- **병렬 말뭉치의 정렬 오류** : 본 논문의 실험에서 사

[표4] 결합 방법의 결과

	θ	재현율(%)	정확도(%)	F값	표준편차	증감	T
L/B&ISF	0.6	96.11	95.71	95.91	0.40	+0.28	0.1181
L/W&ISF	0.6	96.10	95.80	**95.95**	0.29	+0.43	0.0069
L/M&ISF	0.5	95.77	95.89	95.83	0.53	+0.22	0.0120
L/C&ISF	0.5	95.67	95.28	95.47	0.58	+0.54	0.0845
L/B&POS	0.9	95.87	95.43	95.65	0.28	+0.02	0.7426
L/W&POS	0.6	95.89	95.55	95.72	0.17	+0.21	0.0128
L/M&POS	0.7	95.58	95.67	95.63	0.33	+0.02	0.7624
L/C&POS	0.3	95.18	94.79	94.98	0.51	+0.06	0.6555

[표5] 오류의 유형 별 비율

오류 유형	사전	의역	구 대응	전처리	음차 복원	대명사	말뭉치
비율(%)	32.03	17.97	14.84	14.84	12.50	3.91	3.91

용한 병렬 말뭉치의 경우 문장 단위 정렬 결과를 포함하고 있는데, 이 정렬이 잘못 되거나, 사람에 의해서도 바른 정렬을 결정하기 모호한 경우이다.

[표5]는 'L/B'의 방법에 의해서는 정확히 정렬이 되지만, 'L/B&ISF'의 방법에서는 오류를 내는 총 42 개의 단락을 추출하여 오류의 유형 별 비율을 구한 것이다. 42 개 단락으로부터 나온 오류는 총 128 개다. 대역어 사전에 의한 오류가 가장 높은 비율을 차지한다. 이것은 대역어 사전의 정련 작업이 필요함을 의미하며, 어휘 정보를 이용한 문장 정렬의 성능 향상을 위해서는, 가장 먼저 양질의 사전 구축 작업이 필수적임을 나타낸다.

6. 결론 및 향후 연구

문장 정렬은 병렬 말뭉치 활용 및 구축의 기본 작업이라 할 수 있다. 본 논문에서는, 문장 정렬에서 기존의 길이 정보를 이용한 통계적 접근 방법이 한-영 문장 정렬에 얼마나 효과적인 지 알아보았다. 그리고, 길이 정보만을 이용했을 때의 한계점과, 이를 해결하기 위해 어휘 정보의 활용이 필수적임을 지적하였다. 어휘 정보 활용의 방안으로 간단한 단어 정렬을 이용한 방법을 제시하였으며, 길이를 이용한 방법과의 결합 형태도 제시하였다. 실험 결과를 통해 제안한 방법이 정렬의 성능 향상에 기여할 수 있음을 알았고, 오류 분석을 통하여 보다 정확한 어휘 정보 활용을 위해 해결해야할 문제점에는 어떤 것들이 있는지 알아보았다. 발견된 대부분의 문제점들은 본 연구 외의 자연 언어 처리 분야에서도 공통적으로 안고 있는 것들이며, 제반 연구 성과들의 향상에 따라 점차 해결될 것으로 생각된다.

본 논문에서는 문장 정렬 해 탐색을 위한 알고리즘으로, Gale과 Church가 제시한 다이내믹 프로그래밍 알고리즘만을 적용하였다. 이 알고리즘은 좋은 성능을 보이지만, 정렬 비즈 범주에 제약을 받고 있으며, 범주 확장에 따라 탐색 공간이 급속히 증가하는 단점을 지니고 있다. 향후에는 앞서 제안한 단어 정렬 방법이 잘 활용될 수 있는 알고리즘을 연구하고, 다이내믹 프로그래밍 알고리즘과 비교 연구를 진행할 계획이다.

7. 참고 문헌

[1] M. Simard et al., "Using Cognates to Align Sentences in Bilingual Corpora," Proceedings of TMI, 1992.

[2] P. Brown et al., "Aligning Sentences in Parallel Corpora," Proceedings of 29th Annual Meeting for ACL, 1991.

[3] W. A. Gale and K. W. Church, "A Program for Aligning Sentences in Bilingual Corpora," Proceedings of 29th Annual Meeting for ACL, 1991.

[4] S. F. Chen, "Aligning Sentences in Bilingual Corpora Using Lexical Information," Proceedings of 31st Annual Meeting for ACL, 1993.

[5] M. Kay and M. Roscheisen, "Text-Translation Alignment," Computational Linguistics, 19(1), 1993.

[6] H. Papageorgiou et al., "Automatic Alignment in Parallel Corpora," Proceedings of 32nd Annual Meeting for ACL, 1994.

[7] 양주일 외, "웹 문서로부터 한-영 병렬 말뭉치 자동 구축과 문장 단위 정렬", 제 11회 한글 및 한국어 정보처리 학술대회, 1999.

[8] C. D. Manning and H. Schutze, "Foundations of Statistical Natural Language Processing," The MIT Press, pp. 463-494, 1999.

[9] E. Brill, "Some Advances in Transformation-Based Part of Speech Tagging," Proceedings of 12th NCAI, 1994.

[10] 김진동 외, "Twoply HMM : 한국어의 특성을 고려한 형태소 단위의 품사 태깅 모델," 한국정보과학회 논문지, 24(12), 1997.

[11] L. Lee, "Similarity-Based Approaches to Natural Language Processing," PhD thesis, Harvard University, 1997.

[12] 정태구, "한-영 병렬 말뭉치 개발," 학술용역 과제 보고서, 문화관광부, 1999.

[13] 이상섭, "한-영 병렬 말뭉치 개발," 연구보고서, 문화관광부, 2000.