영한 기계번역에서 문장 다시 쓰기에 관한 연구

김 성 동
한성대학교 컴퓨터공학과

Study on Sentence Rewriting in English-Korean Machine Translation

Sung-Dong Kim
Dept. of Computer Engineering, Hansung University

요 약

규칙 기반의 영한 기계번역에서는 영어의 문법 규칙을 구축하고 이를 이용하여 영어의 구문 분석을 수행한다. 그러나 쉼표를 포함한 문장이나 특수한 형식의 문장들은 문법에 의해 분석하기 어렵다. 이를 문법에 의해 분석하기 위해서는 문법이 복잡해지고 문법의 수가 많아지게 되어 분석의 복잡도를 증가시키게 된다. 이러한 문제를 해결하기 위해 이미 존재하는 규칙에 의해 분석할 수 있는 형태로 문장을 바꾸는 문장 다시 쓰기를 제안한다. 문장 다시 쓰기를 위해 쉼표를 포함한 문장에 대해서 다시 쓰기가 필요한 패턴을 구축하였으며 이에 대해 문장 다시 쓰기를 실험하였다. 문장 다시 쓰기를 통해 입력 문장을 변형함으로써 규칙의 추가 없이 구문 분석이 가능하며 제안한 방법은 특수한 형식을 가진 문장 및 쉼표에 의해 연결되는 문장들에 대해 보다 정확한 분석과 번역을 위한 새로운 방법으로서 의의가 있다.

1. 서 론

현재의 영한 기계번역 시스템은 정형화된 문장들에 대해서는 어느 정도 만족할만한 번역 성능을 보이고 있다. 그러나 긴 문장이나 특수한 형식의 문장에 대해서는 자연스러운 번역을 생성하지 못하고 있는 상황이다. 규칙 기반의 번역 방식의 경우 영어 구문 구조 규칙을 문맥 자유 문법 형식으로 표현하는데 일반적인 문장들이 가지는 구문 구조는 쉽게 표현할 수 있는데 특수한 형식의 문장들의 구문 구조는 표현하기가 어렵기 때문에 이러한 문장을 규칙 기반 방식만으로 분석하는데 한계가 있다. 특히 쉼표를 포함하는 문장의 경우 쉼표의 역할[1]에 따라서 다양한 구문 구조가 가능하다. 이러한 상황을 모두 규칙으로 표현하기에는 어려움이 있으며 규칙으로 표현한다 하더라도 규칙의 크기가 커져 구문 분석의 복잡도가 증가하게 된다. 이는 구문 분석의 효율성 저하를 유발하여 실용적인 영한 기계번역 시스템으로서의 역할을 할 수 없게 한다. 복잡한 구문 구조나 특수한 구문 구조를 가지는 영어 문장의 분석을 위해서 숙어 번역(idiom based translation) 방식[2]을 이용하기도 한다. 숙어 번역 방식은 고정된 숙어(fixed format idiom) 또는 구 구조 숙어(phrasal idiom)인 경우에는 효과를 얻을 수 있다. [3]에서는 번역이 어려운 특수 형식의 문장을 숙어 방식으로 인식하고 번역하는 확장된 숙어(extended idiom) 방식을 제안하였다. 그러나 이 경우에는 숙어의 복잡도가 증가하여 숙어 인식이 어려우며 숙어 인식의 부작용(side effect)으로 인해 오히려 구문 분석을 방해하고 결과적으로 잘못된 번역을 생성하기도 한다.

본 논문에서는 특수한 형식의 문장을 올바르게 분석하고 보다 자연스러운 번역을 생성하기 위한 방법으로서 **문장 다시 쓰기(sentence rewriting)**를 제안한다. 입력 문장을 그대로 번역하기 보다는 번역 시스템이 보유하고 있는 규칙으로 분석이 용이한 형식으로 분석 이전에 문장을 변형하여 기존의 규칙으로 분석 및 번역이 가능하게 하려는 것이 문장 다시 쓰기의 목적이다. 예를 들어 “I am going to make an early start ***so that*** I don't get stuck in the traffic.”에서 [so that]은 “그래서” 또는 “~하기 위하여”라는 의미로 번역되어야 한다. 이러한 형식을 규칙으로 표현하기는 어려우며 숙어로 표현한다면 [so that SENT]라는 형식으로 표현된다. 그런데 이 숙어 가 인식되기 위해서는 SENT[1]를 인식해야 하며 이를 위해 부분 파싱(partial parsing)이 필요하다. 이는 숙어 인식을 위한 구문 분석이며 이러한 구문 분석이 전체 구문 분석 및 번역 과정의 복잡도를 높이고 성능을 저하시키는 한 요인이 될 수 있다. 따라서 위의 문장을 구문 분석 이전에 “I am going to make an early start, **so** I don’t get stuck in the traffic.”으로 변형한다면[2] 번역 시스템이 가지고 있는 규칙만으로 쉽게 분석이

[1] 문장(sentence)을 나타냄.

[2] 여기서는 [so that]이 [, so]로 다시 쓰여졌다.

가능하고 따라서 올바른 번역을 얻을 수 있다. 본 논문에서 제안하는 문장 다시 쓰기는 기존 기계번역 시스템에 대한 수정 없이 성능을 개선할 수 있다. 번역이 어려운 특수한 형식의 문장을 수집하는 과정은 현재 전문가 또는 기계번역 시스템의 테스트 과정을 통해 이루어지고 있다. 본 논문에서는 쉼표를 가진 문장에서 쉼표 다시 쓰기 방법에 대하여 논하고 다시 쓰기를 통해 번역 결과가 개선되었음을 보인다.

본 논문은 다음과 같이 구성된다. 2장에서는 복잡한 문장 또는 특수한 문장을 번역하기 위한 기존의 방법을 살펴본다. 3장에서는 쉼표 다시 쓰기 패턴을 제시하고 다시 쓰기 방법이 통합된 기계번역 시스템의 구조를 설명한다. 4장에서는 쉼표 다시 쓰기에 의해 번역 품질이 개선된 결과를 보이고 5장에서 논문을 마무리한다.

2. 관련 연구

규칙 기반의 영한 기계번역 시스템은 언어의 특징이 매우 다른 두 언어를 대상으로 하기 때문에 자연언어의 고유한 모호성 문제 이외에 정확한 번역을 얻기 위해서는 해결해야 할 많은 어려움이 있다. 그 중에서 본 논문은 긴 문장이나 특수한 형식의 문장을 보다 정확하게 번역하기 위한 방법에 대한 연구에 초점을 맞추고 있으며 이에 관한 기존의 연구들로는 다음과 같은 것들이 있다.

긴 문장의 처리를 위해서 일반적으로 문장 분할(sentence segmentation/partitioning) 방법이 적용되었다. [4]에서 부분 파싱에 관한 방식이 제시된 것이 문장 분할 방법이 적용된 시초라 할 수 있다. [5]에서는 문장 패턴을 정의하여 패턴 매칭에 의해 문장을 분할하고 부분 파싱을 적용하였다. [6]에서도 긴 문장의 번역을 위해 문장 패턴을 이용하였는데 여기서는 문장 분할 뿐만 아니라 패턴에 의한 번역도 수행하였다. [7]에서는 확률적인 방법으로 문장 분할을 시도하여 구문 분석의 복잡도를 줄이려는 시도를 하였으며 [8]에서는 기계학습 방법을 이용하여 문장 분할을 시도하였다. 이러한 시도들은 실용적인 번역을 제공할 수 있도록 긴 문장을 처리하기 위한 것이었다. 그러나 복잡하거나 특수한 형식의 문장의 정확한 분석 및 번역에 기여하지는 못하였다고 할 수 있다.

이러한 문제에 대한 해결 방식으로 확장된 숙어 번역 방식(extended idiom translation)이 적용되었다[3]. 기존의 숙어 번역 방식은 고정된 숙어 및 구 구조 숙어를 이용하여 영어와 한국어 간의 차이를 극복하려 하였다. 이에 비해 확장된 숙어 번역 방식은 특수한 문장 형식을 숙어로 간주하기 때문에 숙어 인식을 통해 양 언어 간의 차이를 극복하는데 기여하였다. 그러나 확장된 숙어는 숙어 인식의 어려움을 유발하였으며 결과적으로 숙어 인식의 정확성을 저하시켜 숙어가 아닌 문장에 대해서 오히려 잘못된 번역을 생성하기도 하였다.

문장 패턴을 구축하고 이를 이용하여 문장을 분할, 분석하는 방법이 구문 분석의 복잡도를 줄이는데 기여한 방식에 근거하여 특수한 문장 형식 또는 규칙 및 숙어를 이용하여 번역이 어려운 문장 형식에 대해서 문장 다시 쓰기를 통해 같은 의미의 분석 및 번역이 용이한 문장으로 변형하는 방법을 고안하였다. 즉 전문가의 지식을 활용하여 번역이 곤란한 문장 형식들을 수집하고 이들에 대한 다시 쓰기 패턴을 구축함으로써 구문 분석 이전에 패턴을 이용한 문장 다시 쓰기를 수행하는 것이다. 이는 확장된 숙어 방식의 숙어 인식 부작용 문제를 회피할 수 있을 뿐만 아니라 규칙 기반 기계번역에서 규칙을 확장하지 않고도 복잡한 문장 형식을 가진 입력 문장을 분석할 수 있게 한다.

3. 쉼표 다시 쓰기

본 절에서는 문장 다시 쓰기 과정을 쉼표를 대상으로 설명한다. 그림 1은 문장 다시 쓰기가 추가된 과정을 보여준다[3].

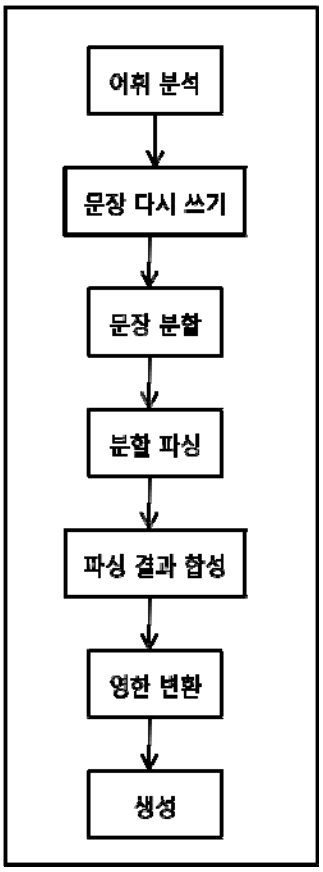

그림 1. 문장 다시 쓰기가 추가된 번역 과정.

[3] 영한 번역 시스템 SmarTran 5.0에 적용된 번역 과정이다.

쉼표를 포함하는 문장은 쉼표를 이용하여 문장 분할을 하게 되지만 경우에 따라서는 문장 분할을 하면 올바른 분석을 하지 못할 수 있다. 이러한 경우에 문장 분할 이전에 다시 쓰기를 통해 문제를 해결할 수 있다. 아래에 쉼표 다시 쓰기 패턴의 종류와 예문을 제시하였다[4].

(1) [ADV , ADV][5] → ADV
Some parts of the sea are *very, very* deep. →
Some parts of the sea are *very* deep.

(2) [ADJ , ADJ] → ADJ and ADJ
I know the fact of the *past, old* snow changes.→
I know the fact of the *past and old* snow changes.

(3) [ADJ , ADJ , and/or ADJ] → ADJ and ADJ and/or ADJ

(4) [ADJ , ADJ , and/or ADV ADJ] → ADJ and ADJ and/or ADV ADJ

(5) [ADJ(COMPR) , ADJ(COMPR)] → ADJ(COMPR) and ADJ(COMPR)

(6) [PRNOUN(HYPHEN) , PRNOUN(HYPHEN)] → PRNOUN(HYPHEN) and PRNOUN(HYPHEN)
The company now provides *high-speed, full-time* connections. → The company now provides *high-speed and full-time* connections.

(7) [ADJ , PRNOUN(HYPHEN)] → ADJ and PRNOUN(HYPHEN)
The program can describe the *sharp, 3-D* images. → The program can describe the *sharp and 3-D* images.

(8) [ADJ , PRNOUN] → ADJ and PRNOUN
It is for *large, Third World* development. →
It is for *large and Third World* development.

(9) [ADJ , PASTP] → ADJ and PASTP
He took a *small, worn* blue copy. → He took a *small and worn* blue copy.

(10) [ADJ , PRESP] → ADJ and PRESP

(11) [ADJ , ADV , ADJ] → ADJ and ADV ADJ

(12) [ADJ, ADJ, ADJ] → ADJ and ADJ and ADJ
The system becomes *reliable, secure, scalable*. → The system becomes *reliable and secure and scalable*.

(13) [PASTP , ADJ] → PASTP and ADJ
The system needs a more *centralized, manageable* environment. → The system needs a more *centralized and manageable* environment.

(14) [PRESP , ADJ] → PRESP and ADJ

(15) [VERB , VERB and VERB] → VERB and VERB and VERB
I am here to *see, listen and learn*. →
I am here to *see and listen and learn*.

여기서 제시된 15가지 패턴 이외에도 추가의 패턴이 있지만 빈도가 적기 때문에 기술하지 않았다. 위에서 보듯이 주로 쉼표를 “and”로 다시 쓰거나 제거하는 간단한 작업을 수행하기 때문에 문장 다시 쓰기는 단순한 작업이라 할 수 있다. 패턴은 약간의 특성 정보(HYPHEN, CAPITAL)를 이용하고 어휘 분석 결과를 사용한다. 다시 쓰기 오류를 줄이기 위해서는 좀 더 정교한 패턴이 되어야 할 것이다.

위에서 살펴본 바와 같이 다시 쓰기는 동등한 의미의 다른 문장으로 원래 문장을 변환한다. 변환된 문장은 번역 시스템에 의해서 원래 문장보다 용이하게 분석될 수 있어 보다 정확한 번역을 얻을 수 있다.

4. 다시 쓰기에 의한 번역 품질 개선

본 절에서는 3절에서 제시한 예문들에 대해서 다시 쓰기를 적용하지 않은 번역문을 (a)에, 그리고 다시 쓰기를 적용한 번역문을 (b)에 제시한다.

(1) Some parts of the sea are very, very deep.
(a) 매우 바다의 부분이 있는 매우 깊은 몇몇
(b) 바다의 몇몇의 부분은 매우 깊습니다.

(2) I know the fact of the past, old snow changes.
(a) 나는 그 과거의 사실을 압니다. 그런데 그들은 오래된 눈 변화[잔돈]입니다.
(b) 나는 과거이고 오래된 그 눈 변화[잔돈]의 사실을 압니다.

(3) The company now provides high-speed, full-

[4] ADJ: 형용사, ADV: 부사, COMPR: 비교급, HYPHEN: 하이픈을 포함한 단어, PASTP: 과거분사형, PRESP: 현재분사형, PRNOUN: 고유명사, CAPITAL: 대문자로 시작하는 단어

[5] 같은 ADV가 반복되는 경우임.

time connections.
(a) 그 회사는 지금 high-speed를 제공합니다. 그런데 그들은 전시간의 연결입니다.
(b) 그 회사는 고속이고 전시간인 연결을 지금 제공합니다.

(4) The program can describe the sharp, 3-D images.
(a) 날카로운 그 프로그램은 설명할 수 있습니다. 그런데 그들은 3-D 이미지입니다.
(b) 그 프로그램은 날카롭고 그리고, 3-D 이미지를 설명할 수 있습니다.

(5) It is for large, Third World development.
(a) 큰 그것은 있습니다. 그런데 그것은 제3세계 개발입니다.
(b) 그것은 크고 제3세계 개발을 위한 것입니다.

(6) He took a small, worn blue copy.
(a) 작은 그는 잡았습니다. 그런데 그것은 헤어진 파란 복사였습니다.
(b) 그는 작고 헤어진 파란 복사를 잡았습니다.

(7) The system becomes reliable, secure, scalable.
(a) 그 시스템은 믿음직하게 됩니다. 확장할 수 있는 안전하게 합니다.
(b) 그 시스템은 믿음직하고 안전하게 됩니다. 그리고 확장할 수 있는

(8) The system needs a more centralized, manageable environment.
(a) 그 시스템은 다루기 쉬운 환경이 중앙에 집중된 더 많은 것을 필요로 합니다.
(b) 그 시스템은 집중했고 다루기 쉬운 더 많은 환경을 필요로 합니다.

(9) I am here to see, listen and learn.
(a) 나는 보기 위해 여기에 있습니다. 듣고 배웁니다.
(b) 나는 보고 듣고 배우기 위해 여기에 있습니다.

위의 결과에서 보듯이 대부분의 문장들은 다시 쓰기에 의해서 번역 결과가 개선되었음을 알 수 있다. 다시 쓰기에 의해서 번역이 달라졌지만 아직도 어색한 번역이 있을 수 있는데 (예를 들면 7, 8번 예문) 이는 번역 시스템이 가지고 있는 문법이 다시 쓰여진 문장의 구문에 대한 적절한 규칙을 가지고 있지 않기 때문이다. 따라서 다시 쓰기는 기존 문법의 규칙을 보완하는 데에도 유용하다고 할 수 있다.

5. 결 론

본 논문에서는 영한 기계번역에서 영어의 특수한 형식의 문장을 보다 쉽게 분석하고 번역하기 위한 방법으로 문장 다시 쓰기를 제안하였다. 영어와 한국어 사이에 존재하는 언어간의 차이 때문에 규칙 기반 기계번역에서 규칙을 이용하여 분석하기도 어렵고 숙어 기반 방식에서 확장된 숙어를 이용하여 분석 및 번역하기도 어려운 문장들에 대해서 다시 쓰기를 적용하였다. 이를 통해 기존의 규칙만으로 분석을 용이하게 할 수 있음을 쉼표 다시 쓰기를 통해 보였다.

문장 다시 쓰기는 번역 시스템의 구문 분석 이전에 구문 분석이 용이한 형태로 변형하는 것으로 기존의 번역 시스템에 독립적이다. 따라서 기존 시스템을 변경할 필요 없이 번역 시스템에 통합될 수 있으며 문장 다시 쓰기가 필요한 특수한 형식 패턴을 쉽게 추가할 수 있는 확장성이 있다.

본 논문에서 제안한 문장 다시 쓰기 역시 부작용이 있을 수 있다. 즉 다시 쓰기 패턴에 맞지만 다시 쓰기를 해서는 안 되는 경우도 존재한다. 현재의 제한된 문맥 정보만을 이용한 다시 쓰기 규칙은 부작용이 더 많이 발생할 수도 있다. 이러한 오류를 줄이기 위해서는 패턴에 추가의 특성 정보를 검사하는 장치가 필요하며 이를 이용하여 보다 정교하게 문장 다시 쓰기 패턴을 기술할 필요가 있다. 즉 확장된 문맥 정보를 활용하여 다시 쓰기 규칙을 보강하여야 한다. 다른 문제로서 문장 다시 쓰기의 효용성을 높이기 위해서 앞으로 번역 시스템이 보유하고 있는 확장형 숙어들을 대상으로 다시 쓰기 패턴을 구축하는 것을 들 수 있다. 또한 긴 문장의 경우도 보다 짧은 여러 개의 문장으로 분할하는 것 대신에 다시 쓰기를 하는 것이 분할의 잘못으로 인한 분석 및 번역 오류를 줄이는데 기여할 것으로 기대된다.

참고문헌

[1] 김성동, 박성훈. “영한 기계번역에서 긴 문장의 구문 분석 정확성 향상을 위한 쉼표 용도의 분류”, 한국정보과학회 추계학술대회, 2006년 10월, 세종대학교.

[2] 윤성희, “영어-한국어 기계번역을 위한 숙어 기반의 효율적 문장 분석”, 박사학위 논문, 서울대학교 컴퓨터공학과, 1993.

[3] Yu-Seop Kim and Yung Taek Kim, "Semantic Implementation based on Extended Idiom for English to Korean Machine Translation," *Journal of the Asia-Pacific Association for Machine Translation, 20*, pp23-39, 1998.

[4] Abney, Steven. “Parsing By Chunks,” Principle-Based Parsing. Kluwer Academic Publishers. pp. 257-

279, 1991.
[5] Sung Dong Kim and Yung Taek Kim. “Sentence Analysis using Pattern Matching in English-Korean Machine Translation,” In *Proceedings of the 1995 International Conference on Computer Processing on Oriental Language*, pp. 199-206, 1995.
[6] Yoon-Hyun Rho, Monpyo Hong, Sung-Kwon Choi, Ki-Yong Lee, Sang-Kyu Park. “For the Proper Treatment of Long Sentences in a Sentence Pattern-based English-Korean MT System,” In Proceedings of MT Summit IX, 2003.
[7] Sung Dong Kim,Byoung-Tak Zhang, and Yung Taek Kim. “Reducing Parsing Complexity by Intra-Sentence Segmentation based on Maximum Entropy Model,” In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, pp. 164-71, 2000.
[8] Sung-Dong Kim, Byoung-Tak Zhang, and Yung Taek Kim. “Learning-based Intrasentence Segmentation for Efficient Translation of Long Sentences,” *Machine Translation*, Vol. 16, No. 3, pp. 151-174, 2001.