Asia-pacific Journal of Convergent Research Interchange
Vol.9, No.5, May 31 (2023), pp.55-67
http://dx.doi.org/10.47116/apjcri.2023.05.05

A Methodology for Purifying Data for Machine Translation Using Vector Alignment

벡터 얼라인먼트를 활용한 기계 번역 학습 데이터 정제 방법론

Ahyoung Lee[1], Donghoon Lee[2], Namgyu Kim[3]

이아영[1], 이동훈[2], 김남규[3]

[1] Graduate Student, Graduate School of Business IT, Kookmin University, Korea, dkdud5068@kookmin.ac.kr

[2] Graduate Student, Graduate School of Business IT, Kookmin University, Korea, ldh1468@kookmin.ac.kr

[3] Professor, Graduate School of Business IT, Kookmin University, Korea, ngkim@kookmin.ac.kr

Corresponding author: Namgyu Kim

Abstract: With recent advancements in GPUs and novel AI algorithms, remarkable progress has been made in the field of deep learning-based machine translation. A vast amount of parallel corpus consisting of pairs of source and target languages is required to perform machine translation tasks successfully. However, such a parallel corpus is rare, and it is difficult to verify its quality. Although the quality of machine translation training data is a crucial factor directly affecting the performance of translation models, discussions on the quality of machine translation training data have not been addressed sufficiently so far. In this study, therefore, we propose a method to improve the quality of machine translation training data using deep learning-based vector alignment techniques. Specifically, we train a deep learning-based vector alignment model to convert English sentence vectors to Korean sentence vectors using high-quality Korean-English data, align the English sentence vectors of newly given Korean-English data to the Korean vector space using the trained model, and remove the ones with the low similarity between the aligned vectors and Korean sentence vectors as noise. Experimental results show that the proposed method achieves the best performance with an F1-score of 0.860 on average when refining 10% of the entire data.

Keywords: Deep Learning, Machine Translation, Data Purification, Vector Alignment

요약: 최근 GPU 및 AI 알고리즘의 발전에 따라, 딥 러닝 기반 기계 번역 분야에서도 괄목할 성과가 나타나고 있다. 기계 번역 태스크를 수행하기 위해서는 원문 언어와 대상 언어의 쌍으로 구성된 방대한 양의 병렬 코퍼스가 필요한데, 이러한 병렬 코퍼스가 드물 뿐 아니라 해당 데이터의 품질을 확인하기 어렵다는 어려움이 존재한다. 하지만 기계 번역 학습 데이터의 품질은 번역 모델의 성능에 직접적인 영향을 미치는 매우 중요한 요소임에도 불구하고, 기계 번역 학습 데이터의 품질에 대한 논의는 거의 다루어지지 않았다. 이에 본 연구에서는 벡터 정렬 기술을 활용하여 기계 번역 학습용 데이터의 품질을 높이는 방안을

Received: January 26, 2023; 1st Review Result: March 13, 2023; 2nd Review Result: April 09, 2023
Accepted: April 30, 2023

제시하였다. 구체적으로 제안 방법론은 고품질 한/영 데이터를 사용하여 영어 문장 벡터를 한글 문장 벡터로 변환하는 딥러닝 기반 벡터 정렬 모델을 학습하고, 학습된 모델을 사용하여 새로 주어진 한/영 데이터의 영어 문장 벡터를 한글 벡터 공간으로 정렬한 뒤, 정렬된 벡터와 한/영 데이터의 한글 문장 벡터와의 유사성이 낮은 경우 이를 노이즈로 간주하여 제거한다. 제안 방법론을 실제 데이터 분석에 적용한 실험 결과, 전체 데이터 중 10%를 정제했을 때의 성능이 F1-score 기준 평균 0.860으로 가장 우수하게 나타남을 확인하였다.

핵심어: 딥 러닝, 기계 번역, 데이터 정제, 벡터 정렬

1. 서론

최근 컴퓨팅 기술이 빠르게 발전함에 따라 대량의 데이터를 수집하는 것이 용이하게 되었으며, 딥 러닝(Deep Learning)을 활용하여 방대한 양의 데이터를 학습하고 이를 바탕으로 좋은 성능을 내는 연구가 활발히 수행되고 있다. 딥 러닝은 여러 층(Layers)의 깊은 구조로 이루어진 인공신경망(Artificial Neural Network)을 사용한 분석 기법으로, 자연어 처리 분야에서 높은 수준의 인식 정확도를 달성하며 지속적인 관심을 받고 있다.

텍스트 데이터를 활용하여 다양한 태스크를 수행하기 위해서는 단어, 문장, 그리고 문서와 같은 비정형 데이터를 그 의미를 정확히 포함한 정형 데이터로 변환하여 표현할 수 있는 작업이 반드시 수행되어야 한다. 데이터의 특징을 보존하면서 특정 공간으로 정형화하는 이 작업을 임베딩(Embedding)이라 칭하며, 통계 기반 기법, 신경망 기반 기법 등 다양한 임베딩 기법이 연구되고 있다. 임베딩을 통해 단어와 단어 간 혹은 문서와 문서 간 관계를 파악할 수 있으며, 전이 학습(Transfer Learning)을 통해 단어나 문서의 임베딩 값을 다른 모델의 입력 값으로도 사용하는 시도도 다수 이루어지고 있다.

임베딩을 활용한 태스크들은 대부분 하나의 언어로 기술된 텍스트 데이터를 하나의 임베딩 공간(Embedding Space)에 임베딩하여 데이터의 표현을 학습하는 데 초점을 맞췄다. 하지만 최근 들어서는 하나의 언어가 아니라 다국어로 구성된 데이터를 임베딩하는 다중 언어 임베딩(Multilingual Word Embedding)[1] 에 대한 관심이 높아지면서 이를 다양한 태스크에 적용하려는 시도도 활발히 이루어지고 있다. 특히 기계 번역 태스크에서 다중 언어 임베딩 벡터의 활용도가 점점 높아지고 있다.

기계 번역은 인간의 개입 없이 컴퓨터 프로그래밍을 통해 원문 언어(Source Language)로 구성된 데이터를 대상 언어(Target Language)로 구성된 데이터로 변환하는 것을 의미하며, 자동 번역이라 불리기도 한다. 기계 번역 태스크를 수행하기 위해서는 원문 언어와 대상 언어의 쌍으로 구성된 병렬 코퍼스가 필요하며, 임베딩을 통해 원문 언어와 대상 언어의 벡터들을 도출하여 활용한다. 이 기계 번역 태스크를 수행하기 위해서는 방대한 양의 학습 데이터가 필요한데, 병렬 코퍼스로 이루어진 방대한 양의 데이터가 드물 뿐만 아니라 해당 데이터의 품질을 담보할 수 없는 한계가 존재한다. 품질 저하를 일으키는 병렬 코퍼스의 구체적인 예는 [표 1]을 통해 확인할 수 있다.

[표 1]은 영어 문장과 한글 문장으로 이루어진 병렬 코퍼스의 예시이다. 본 예에서 1번과 2번의 데이터는 비교적 번역이 충실하게 이루어진 반면, 3번과 4번의 한글 문장은 영어 원문의 의미를 충분히 정확하게 표현하고 있다고 할 수 없다. 이러한 번역 품질의 저하 현상은 단순한 번역 오류로 인해 나타날 수 있지만, 전문 어휘가 많이 포함되어 있거나 특수 기호, 혹은 한자 등이 포함된 경우에도 비일비재하게 나타날 수 있다. 이와

같이 원문 언어와 대상 언어의 쌍으로 구성된 병렬 코퍼스 간 의미적 유사성이 낮은 데이터들을 기계 번역의 학습 데이터로 사용할 경우, 모델이 잘못된 정보를 학습하게 되어 번역 태스크의 성능도 낮게 나타날 우려가 있다. 따라서 다국어 말뭉치로부터 두 언어 간 번역의 품질이 현저히 낮은 문장들을 추출하여 제거함으로써, 기계 번역의 학습용으로 구축된 다국어 데이터의 품질을 향상시킬 필요가 있다.

[표 1] 병렬 코퍼스 구성 예시

[Table 1] Examples of Parallel Corpus Configuration

번호	영어	한국어	의미 유사성
1	Characteristic peaks indicative of other impurities such as WO3 did not appear.	WO3와 같은 다른 불순물을 나타내는 특징적인 피크가 나타나지 않았다.	높음
2	The user can click-and-drag the widget to rotate, making other parts of the widget visible to the user.	사용자는 위젯을 클릭 앤 드래그하여 회전시키면서, 위젯의 다른 부분을 사용자에게 보이게 할 수 있다.	높음
3	The blocking area 222 may be disposed on both sides of the open area 221.	제1색필터는 청색광을 차단하는 차단 필터로 기능할 수 있다.	낮음
4	Moreover, resin may have a repeating unit which has a carbonate structure, as shown in Fig.14.	그림 14와 같다.	낮음

이에 본 연구에서는 딥 러닝 기반 벡터 정렬을 통해 기계 번역 학습 데이터의 품질을 향상시키는 방법론을 제안한다. 구체적으로 제안 모델은 원문 문장과 이에 대응되는 대상 언어의 문장을 각각의 공간에 임베딩하여 벡터를 도출하고, 인코더/디코더 기반의 비선형 정렬을 통해 원문 문장을 대상 언어의 공간으로 매핑한다. 다음으로 해당 원문 문장에 대응되는 대상 언어의 문장 벡터와 원문 문장의 비선형 정렬을 통해 매핑된 대상 언어의 문장 벡터 간 유사도를 측정한다. 본 연구에서는 유사도가 지나치게 낮은 경우 대상 언어의 문장은 원문 문장의 의미를 충분히 반영하지 못한 것으로 이해할 수 있으므로, 이러한 데이터를 노이즈(Noise)로 간주하여 데이터 셋에서 제거하여 학습 데이터의 품질을 향상시키고자 한다.

본 논문의 이후 구성은 다음과 같다. 다음 장인 2장에서는 기계 번역 및 벡터 정렬에 대한 기존 연구를 정리하고, 3장에서는 본 연구에서 제안하는 방법론을 소개한다. 4장에서는 제안 방법론을 실제 데이터에 적용한 실험의 과정 및 결과를 제시하고, 마지막 장인 5장에서는 본 연구의 기여와 한계를 정리한다.

2. 관련 연구

2.1 기계 번역

기계 번역은 인간의 개입 없이 컴퓨터 프로그래밍을 통해 원문 언어의 데이터를 대상 언어의 데이터로 변환하는 기술이다. 기계 번역은 실제 언어의 구조와 문법을 바탕으로 규칙을 만들어 번역하는 규칙 기반(Rule-Based) 방식으로부터 시작되었는데, 원문 언어와 대상 언어의 문법 구조가 상이한 경우 번역이 제대로 이루어지지 않는 한계가 존재하여 문장을 작은 단위로 구분하고 이를 각각 번역해서 조합하는 통계 기반 기계 번역

기술(Statistical Machine Translation)이 등장하였다. 하지만 통계 기반 기술 또한 문장 구조가 상이하여 어순이 다를 경우 올바른 배열 조합을 찾아야 한다는 어려움이 존재한다. 따라서 이러한 한계를 개선하고자 신경망 기반 기계 번역(Neural Machine Translation)이 등장하였으며, 특히 seq2seq[2] 기법이 고안되면서 기계 번역은 빠른 속도로 발전하게 되었다.

기계 번역 태스크를 수행하기 위해서는 원문 언어와 대상 언어의 쌍으로 구성된 학습 데이터가 필요하다. 기계 번역은 원문 언어 데이터와 번역한 대상 언어 데이터를 비교하면서 오류를 줄여 나가는 과정으로 학습이 이루어지는데, 일반적으로 번역에 많이 사용되는 알고리즘인 Transformer[3]는 방대한 양의 병렬 코퍼스를 구축하여 학습을 진행한다. 따라서, 기계 번역은 학습 데이터를 얼마나 많이 확보하는지에 따라 기계 번역 서비스의 품질을 높일 수 있다. 최근에는 기계 번역 학습에 사용될 학습 데이터 양을 늘리는 연구가 지속적으로 이루어지고 있으며, 가장 대표적으로 데이터 증강(Data Augmentation)을 통한 학습 데이터 확보 방안이 많이 연구되고 있다.

하지만 데이터 증강에 사용된 병렬 데이터 중 원문 언어의 데이터와 대상 언어의 데이터 간 연결이 부적절한 노이즈 데이터가 많을 경우, 데이터 증강 과정에서 노이즈도 함께 증가하여 분석에 더 부정적인 영향을 미치는 결과가 발생할 수 있다. 노이즈의 존재가 기계 번역의 품질에 미치는 영향은 기존 연구에서도 중요하게 다루어진 바 있다[4]. 따라서 기계 번역용 학습 데이터의 증강에 선행하여 초기 학습 데이터로부터 양질의 데이터를 선별하는 작업이 반드시 수행되어야 하며, 이 과정에 대한 중요성도 최근 연구에서 부각되고 있다[5].

학습 데이터의 정제를 위해 원문 언어 데이터와 대상 언어 데이터에 동일하게 등장하는 중복 단어를 제거하거나, 오류에 대한 수정 규칙을 적용하는 사후 교정 방안 등이 연구된 바 있다. 하지만 이들 연구는 형태적, 문법적인 검토를 다룰 뿐 의미적 차원에서의 품질을 평가하거나 개선하기 위한 연구는 아니라는 한계를 갖는다. 즉 원문과 번역문의 의미적 품질을 자동으로 평가하는 방안에 대한 연구는 상대적으로 매우 부족한 실정이다.

2.2 벡터 정렬

벡터 정렬(Vector Alignment)이란 서로 다른 공간의 데이터 간 매핑을 목적으로 고안된 기술이다. 벡터 정렬은 한 공간에 존재하는 벡터들의 상호 관계를 유지하면서, 해당 언어의 표현을 다른 언어의 표현으로 정렬하는 변환 행렬을 통해 이루어진다. 벡터 정렬은 교차 언어 전이 학습(Cross-lingual Transfer Learning)[6]에서 많이 활용되는 핵심 기술로, 대규모 양의 데이터 구축, 유사 문장 탐색, 기계 번역 등의 다양한 태스크에서 많이 활용되고 있다.

벡터 정렬은 우선 전체 데이터에 대한 임베딩을 통해 벡터를 도출한 후, 임베딩 벡터 간 변환 행렬을 도출하거나 매핑 함수를 학습하는 방식으로 진행된다. 이후 코사인 유사도 및 유클리드 거리와 같은 유사도 계산을 통해 기존 데이터와 매핑하고자 하는 데이터 간의 상호 관계를 파악할 수 있다. 일반적으로 벡터 정렬은 통계적 방식인 치환 행렬(Permutation Matrix)과 직교 행렬(Orthogonal Matrix)을 통해 선형적으로 이루어진다.

하지만 선형 정렬은 서로 다른 두 데이터의 공간을 기하학적으로 동일한 형태라고 가정한 후 정렬을 수행하기 때문에 동형 조건이 유지되지 않을 때는 정렬의 품질이

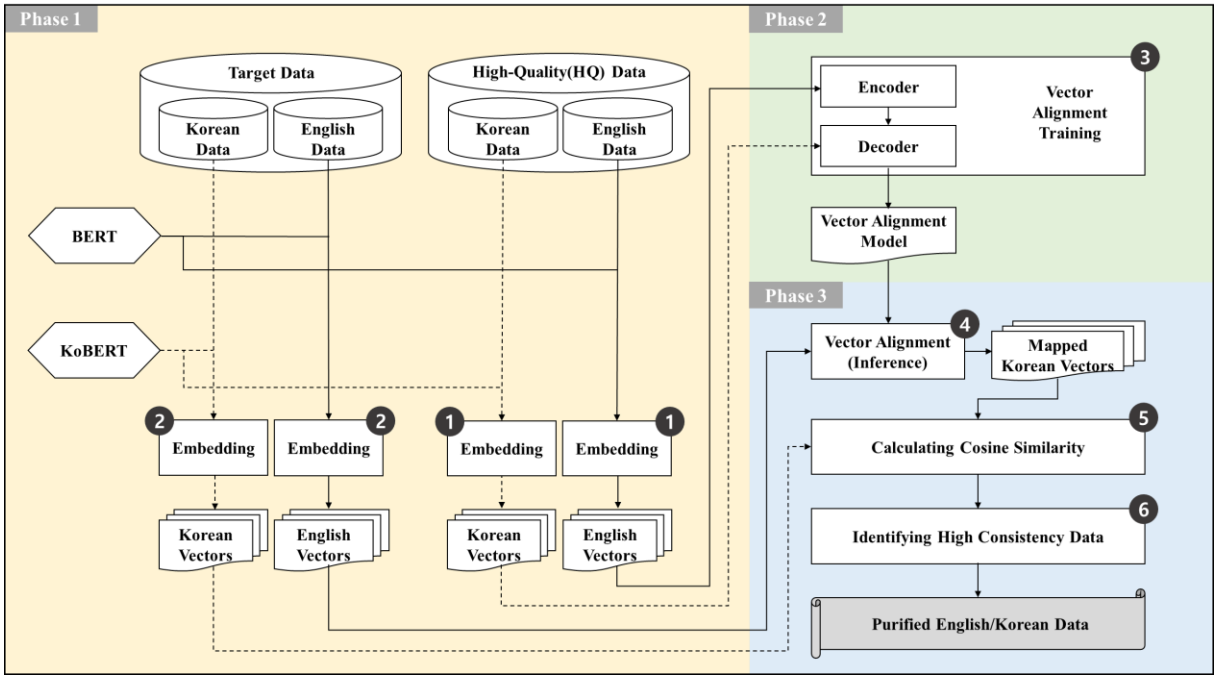
떨어진다는 한계를 갖는다. 특히, 매핑 대상이 되는 두 언어가 어원적으로 상이할 경우, 벡터 정렬 시 직교 변환이 제대로 이루어지지 않아 언어 간 매핑이 완전하게 이루어질 수 없다는 한계가 존재한다[7].

따라서 선형 정렬 방식의 한계를 극복하기 위해 머신 러닝 기반 모델을 통해 이종 데이터의 앵커 포인트를 한 공간에서 매칭시키는 비선형 정렬에 대한 연구가 등장하였다. 최근에는 딥 러닝을 활용한 비선형 정렬 연구가 활발하게 이루어지고 있으며, 대표적인 예로 오토인코더(Autoencoder)[8]를 통해 서로 다른 공간에서 임베딩된 전문어와 범용어를 매핑한 연구[9]가 수행된 바 있다. 해당 연구는 오토인코더를 사용한 비선형 벡터 정렬 방식이 기존 선형 정렬 모델보다 높은 성능을 보임을 확인하였다[9].

3. 제안 방법론

3.1 제안 방법론 개요

본 장에서는 비선형 기반의 벡터 정렬 결과를 활용하여 병렬 데이터 내 의미 유사성(Semantic Consistency)이 낮은 저품질 데이터를 정제하는 방법론을 제안한다. 제안 방법론의 전체적인 과정은 [그림 1]과 같으며, 해당 그림에서는 영문 데이터를 한글 데이터 공간으로 매핑하는 예를 통해 제안 방법론을 소개한다.



[그림 1] 제안 방법론의 전체 개요

[Fig. 1] Overall Overview of the Proposed Methodology

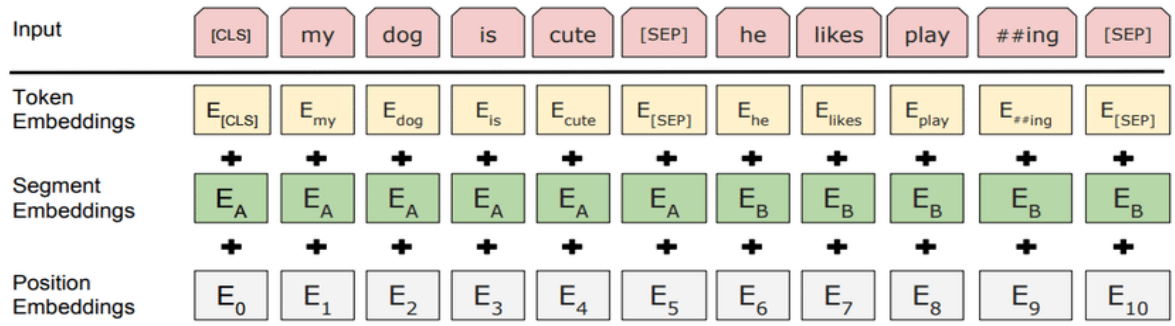
제안 방법론은 총 3단계로 구성된다. Phase 1은 각 데이터로부터 벡터를 도출하는 과정으로, 한/영 간 고품질 번역이 이미 검증된 이상적인 고품질(High-Quality, HQ) 데이터와 품질이 검증되지 않은 정제 대상(Target) 데이터에 대한 임베딩이 수행된다. 이후 Phase 2는 HQ 데이터의 한글과 영문 벡터에 대한 학습을 통해 벡터 정렬 모델을

구축한다. 마지막으로 Phase 3은 해당 모델을 사용하여 Target의 영문 데이터의 벡터를 입력으로 받아 한글 데이터 공간의 벡터로 추론한다. 또한 Phase 3에서는 이렇게 추론된 한글 벡터와 실제 정답인 한글 벡터와의 유사도 비교를 통해 유사도가 낮은 벡터 쌍을 식별하고, 이를 저품질 데이터로 인식하여 제거하는 작업을 수행한다. 제안 방법론을 구성하는 각 단계에 대한 자세한 설명은 본 장의 이후 절에서 다룬다.

3.2 임베딩 벡터 도출

본 절에서는 Phase 1의 학습 데이터와 추론 데이터에 대하여 임베딩을 통해 벡터를 도출하는 단계, 즉 서로 다른 두 언어로 구성된 데이터들을 각각 다른 사전 학습 언어 모델을 통해 임베딩하는 과정을 소개한다. 본 방법론은 다양한 이종 언어 쌍에 대해 적용될 수 있지만, 본 예에서는 이해를 돕기 위해 영문과 한글로 구성된 병렬 데이터에 한정하여 이후 과정을 소개한다. 우선 제안 방법론을 적용하기 위해서는 임베딩을 통해 텍스트 데이터를 컴퓨터가 이해할 수 있는 벡터로 변환해주어야 하며, 구체적으로 본 연구에서는 BERT[10] 기반의 사전 학습 언어 모델을 통해 각 문장에 대한 임베딩을 수행한다.

전통적인 BERT의 임베딩 층은 [그림 2]와 같이 Token Embeddings, Segment Embeddings 그리고 Position Embeddings 총 3가지 층으로 구성되어 있다. 시퀀스의 첫 토큰인 [CLS]는 새로운 시퀀스의 시작을 나타내며, 중간에 나타난 토큰인 [SEP]는 문장의 구분을 위해 사용된다. 이러한 3가지 유형의 임베딩에 대한 합이 BERT의 입력으로 사용되며, 이러한 입력 벡터들은 이후 BERT의 12개 레이어를 거치면서 문맥 정보를 학습한다.



[그림 2] BERT의 입력 표현 구조[10]

[Fig. 2] Input Representation Structure of BERT[10]

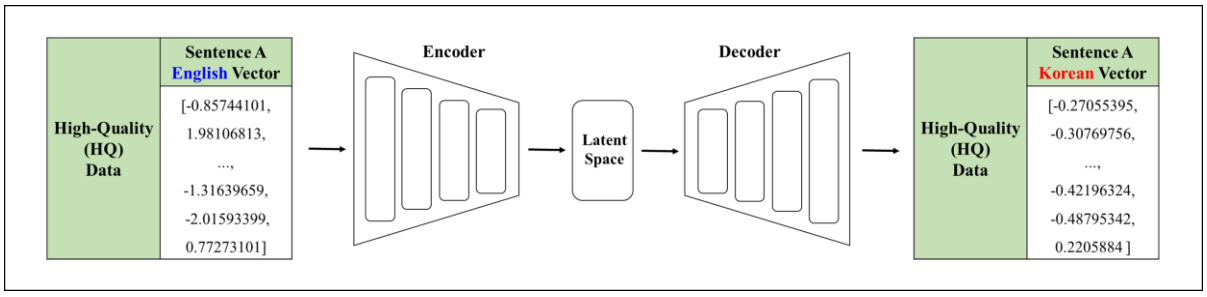
BERT의 12개 층에 대한 학습이 완료된 후, 이로부터 문장 벡터를 도출하는 과정은 매우 다양하게 연구되어 왔다. 예를 들면 [CLS] 토큰을 문장의 대표 벡터로 사용할 것인지 아니면 전체 토큰 벡터의 평균으로 문장 벡터들 도출할 것인지에 대한 논의가 있었으며, 마지막 층의 결과만 활용할 것인지 아니면 여러 층의 결과를 합산하여 사용할 것인지에 대한 논의도 진행되었다. 본 연구에서는 BERT의 다양한 최종 임베딩 방식 중 벡터 길이를 유지하면서도 성능이 좋은 것으로 알려진 방식[10], 즉 마지막 4개 층의 모든 토큰 벡터들의 평균을 합산하여 문장 벡터를 도출하는 방식을 채택하였다.

또한, BERT 자체가 영어로 구성된 데이터로 학습된 모델이기 때문에, 영어와는 상이한 언어적 특성을 갖는 한국어의 경우 기본 BERT만으로는 정확한 특징 추출이 어렵다는 한계가 지적되어 왔다. 따라서 본 연구에서는 영문 데이터는 BERT를 통해 임베딩하고 한글 데이터는 한국어 기반 BERT모델 중 7만여개의 한국어 말뭉치 데이터로 학습한 모델인 KoBERT를 통해 임베딩한다.

이상 본 연구의 Phase 1을 요약하면 다음과 같다. 우선 학습에 사용할 HQ 데이터에 대하여 영문 데이터들은 BERT로, 한글 데이터들은 KoBERT로 임베딩을 진행하여 768차원으로 구성된 벡터를 도출한다(①). 이후 정제하고자 하는 Target 데이터에 대해서도 동일한 방식으로 임베딩을 진행하여 벡터를 도출한다(②). 결과적으로 Phase 1을 통해 학습과 추론에 사용할 네 가지 유형의 벡터 집합을 도출하게 된다.

3.3 벡터 정렬 모델 구축

본 절에서는 Phase 2의 벡터 정렬 모델 학습 과정에 대하여 서술한다. 본 단계는 Phase 1에서 도출한 학습 데이터인 HQ데이터의 영문 벡터와 한글 벡터들을 사용하여 벡터 정렬 모델을 학습한다(③). 제안 모델은 [그림 3]과 같이 딥 러닝 기반 인코더/디코더 구조를 사용하여 구현한다.



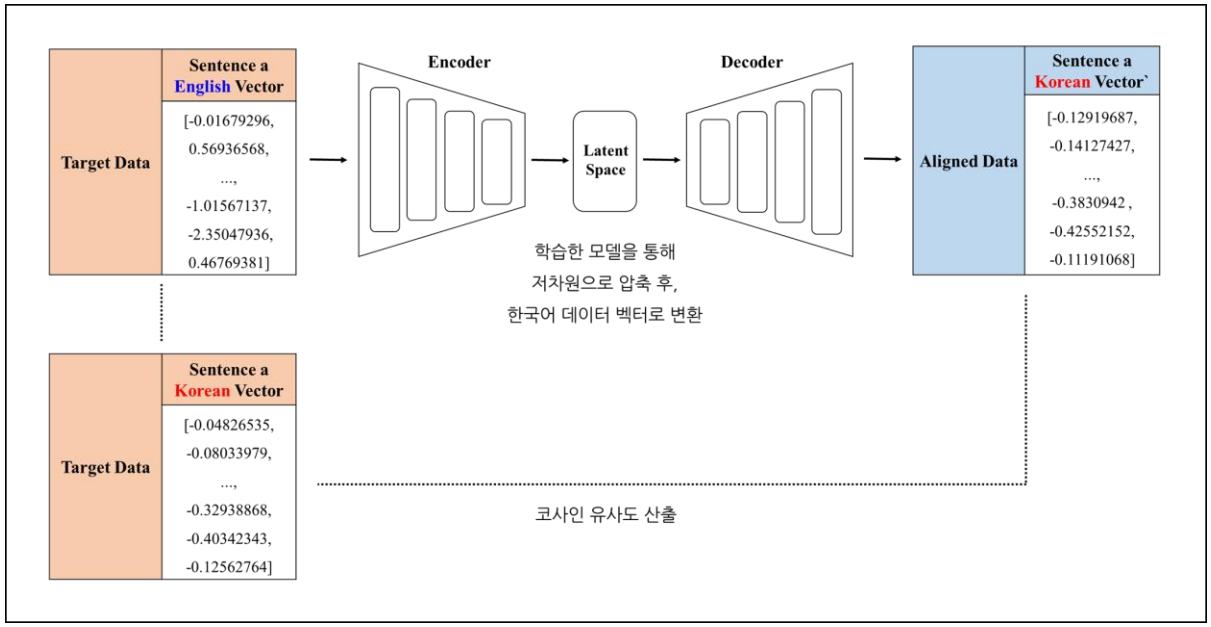
[그림 3] 인코더/디코더 구조의 비선형 벡터 정렬 모델 예시

[Fig. 3] Example of the Non-linear Vector Alignment Model with Encoder/Decoder Structure

본 단계에서는 원문 언어와 대상 언어 간 의미 유사성이 높은 고품질의 병렬 데이터, 즉 HQ 데이터의 영문 및 한글 벡터를 학습에 사용한다. 구체적으로는 [그림 3]의 인코더/디코더 구조에 HQ 영문 벡터를 입력으로, HQ 한글 벡터는 출력으로 제공하여 학습을 진행한다. 본 모델의 구조는 오토인코더(Autoencoder)[8]와 동일하게 고차원의 벡터를 잠재 공간으로 압축한 후 이를 다시 고차원으로 복원하는 과정으로 이루어져 있다. 하지만 제안 모델은 오토인코더와 달리, 입력과 출력에 제공되는 벡터가 서로 다른 딥 러닝 기반 인코더/디코더 구조의 비선형 정렬 모델이다. 본 모델을 통해 입력으로 주어진 영문 벡터를 잠재 공간으로 압축하고 이를 다시 한글 벡터로 복원하는 학습을 수행한다. 이때 학습에 사용된 데이터는 고품질 데이터, 즉 영문 데이터와 한글 데이터의 의미가 동일하거나 매우 유사함이 보장된 데이터이므로, 이러한 데이터를 통해 학습을 마친 정렬 모델은 새로 주어지는 영문 벡터에 대응하는 이상적인 한글 벡터를 추론할 수 있다.

3.4 저품질 데이터 정제

본 절에서는 앞에서 소개한 비선형 벡터 정렬 모델을 사용하여, 정제 대상인 Target 데이터로부터 영문과 한글의 의미 유사성이 낮은 데이터 쌍을 제거하여 Target 데이터의 품질을 향상시키는 Phase 3의 과정을 서술한다(그림 4). [그림 4]의 좌측 상단에는 정제하고자 하는 Target 데이터의 영문 벡터가 제시되어 있으며, 해당 데이터의 한글 벡터는 좌측 하단에 제시되어 있다. Phase 3에서는 우선 앞에서 구축한 벡터 정렬 모델에 정제하고자 하는 Target 데이터의 영문 벡터를 입력으로 제공하여 [그림 4]의 우측과 같은 한글 벡터를 도출한다(④).



[그림 4] 벡터 정렬 후 코사인 유사도 기반 데이터 정제 과정 예시

[Fig. 4] Example of Cosine Similarity-based Data Refinement Process after Vector Alignment

전술한 바와 같이 앞에서 구축한 정렬 모델은 고품질 데이터를 학습한 모델이므로, 본 모델은 입력 영문 벡터에 대응하는 이상적인 한글 벡터를 반환한다. 따라서 입력 영문 문장에 대해 쌍으로 주어진 한글 문장이 의미상 적절하다면, 해당 한글 문장의 벡터(즉, 그림 좌측 하단의 벡터)는 이상적인 한글 벡터(즉, 그림 우측의 벡터)와 동일하거나 매우 유사하게 나타나야 한다. 본 방법론에서는 이들 벡터 간 유사도 측정을 위해 코사인 유사도를 사용한다(⑤). 즉 입력 영문 문장에 대해 추론으로 도출한 이상적인 한글 벡터와 입력 영문 문장에 대해 쌍으로 주어진 한글 벡터의 코사인 유사도가 충분히 높게 나타난 경우, 해당 입력의 영어 문장과 한글 문장은 의미적 유사성이 높은 양질의 병렬 데이터로 간주할 수 있다. 이와 반대로 코사인 유사도가 지나치게 낮게 나타난 경우, 해당 입력의 영어 문장과 한글 문장은 의미적 유사성이 높지 않은 것으로 간주할 수 있다. 이러한 원리에 근거하여 코사인 유사도가 낮은, 즉 원문 언어와 대상 언어 데이터 간 의미 유사성이 낮은 번역 쌍을 제거함으로써 저품질 데이터를 정제한다(⑥). 이러한 과정을 통해 제안 방법론은 병렬 데이터의 품질을 향상시킬 수 있을 것으로 기대하며,

제안 방법론의 성능은 다음 4장의 실험을 통해 검증한다.

4. 실험

4.1 실험 설계

본 절에서는 제안 방법론의 성능을 검증하기 위한 실험의 개요를 소개한다. 3장에서는 번역이 완벽하게 이루어진 것으로 가정된 HQ 데이터를 사용하여 모델을 학습하고, 해당 모델을 번역의 품질이 검증되지 않은 Target 데이터에 적용하여 Target 데이터를 정제하였다. 하지만 이상적인 HQ를 확보하는 것은 현실적인 어려움이 있으므로, 본 연구에서는 일반적으로 사용되는 번역 학습용 데이터를 HQ 데이터로, 인위적 노이즈를 삽입한 저품질 데이터를 Target데이터로 간주하여 성능을 평가하였다. 또한 제안 모델의 성능은 Target 데이터에 포함된 노이즈의 비율에 따라 상이하게 나타날 수 있으므로, 노이즈의 비율을 변화시켜가며 성능의 변화를 측정하였다.

구체적인 실험 과정은 다음과 같다. 실험을 위해 AI허브에 구축되어 있는 기술/과학 분야의 한/영 기계 번역 병렬 말뭉치를 수집하였으며, 한글 문장 및 이를 번역하여 전문가가 검증한 영문 문장의 쌍을 대상으로 실험을 수행하였다. 구체적으로 실험 데이터는 총 10만 개의 쌍을 선정하였고, 이를 학습용 6만 개와 검증용 4만 개로 분할하였다. 해당 데이터는 전문가의 검증을 거친 데이터이므로, 본 실험에서는 이를 HQ 데이터로 간주하였다. 즉 학습용 6만 개의 데이터는 그대로 HQ를 유지한 상태로 학습을 진행하여 벡터 정렬 모델을 생성하였고, 검증용 4만 개의 데이터는 강제 변환을 통해 저품질 데이터로 구축하였다. 저품질 데이터의 구축 과정 예는 [그림 5]와 같다.

[원본 병렬 데이터]

번호	영어	한국어	평가
1	Characteristic peaks indicative of other impurities such as WO3 did not appear.	WO3와 같은 다른 불순물을 나타내는 특징적인 피크가 나타나지 않았다.	의미 유사성이 높음
2	The user can click-and-drag the widget to rotate, making other parts of the widget visible to the user.	사용자는 위젯을 클릭 앤 드래그하여 회전시키면서 위젯의 다른 부분을 사용자에게 보이게 할 수 있다.	의미 유사성이 높음
3	The blocking area 222 may be disposed on both sides of the open area 221.	차단 영역(222)은 개방 영역(221)의 양측에 배치될 수 있다.	의미 유사성이 높음

↓ *Shuffling*

[노이즈 데이터]

번호	영어	한국어	평가
1	Characteristic peaks indicative of other impurities such as WO3 did not appear.	차단 영역(222)은 개방 영역(221)의 양측에 배치될 수 있다.	의미 유사성이 낮음
2	The user can click-and-drag the widget to rotate, making other parts of the widget visible to the user.	WO3와 같은 다른 불순물을 나타내는 특징적인 피크가 나타나지 않았다.	의미 유사성이 낮음
3	The blocking area 222 may be disposed on both sides of the open area 221.	사용자는 위젯을 클릭 앤 드래그하여 회전시키면서 위젯의 다른 부분을 사용자에게 보이게 할 수 있다.	의미 유사성이 낮음

[그림 5] 의미 유사성이 낮은 노이즈 번역 쌍 생성 과정 예시

[Fig. 5] Example of Generating a Noise Translation Pair with Low Semantic Similarity

[그림 5]의 상단은 원본 데이터, 즉 HQ 상태의 데이터이다. 이때 해당 데이터의 원문 언어는 그대로 두고 대상 언어 데이터를 강제로 섞어서, [그림 5]의 하단 표와 같이 원문 언어와 대상 언어 간 엉뚱한 내용이 서로 대응되도록 구성하였다. 즉 주어진 Target 데이터의 일부 영어 문장에 대해, 대응되는 한글 문장이 아닌 엉뚱한 한글 문장을 대응시킴으로써 노이즈 데이터를 생성하였으며, Target 데이터 4만 개 중 노이즈 데이터의 비율을 5%에서 25%까지 변화시켜가며 제안 방법론의 성능을 평가하였다.

4.2 실험 결과

본 절에서는 실험 수행 결과를 요약한다. 전술한 바와 같이 Target 데이터 중 노이즈의 비율은 5%에서 25%까지 변화시켜가며 구성하였다. 또한 정제율, 즉 주어진 데이터 중 제안 방법론을 통해 제거할 데이터의 비율이 높을수록 더욱 많은 노이즈가 제거될 것이며, 이와 동시에 불필요하게 제거되는 정상 데이터의 양도 증가할 것이다. 따라서 정제율을 5%에서 50%까지 변화시켜가며 성능을 측정하였다. 성능 지표로는 정상 데이터는 얼마나 많이 보존되었는지, 그리고 노이즈 데이터는 얼마나 많이 삭제되었는지를 동시에 측정하기 위해 F1-Score를 사용하였다. 실험 결과는 [표 2]와 같다. [표 2]의 각 행은 정제율, 즉 삭제될 데이터 개수의 변화에 따른 성능을 나타내며, 각 열은 Target 데이터 중 노이즈 비율의 변화에 따른 성능을 나타낸다. 또한 맨 우측 두 열은 다양한 노이즈 비율의 변화에 따른 성능의 평균 및 편차를 나타낸다.

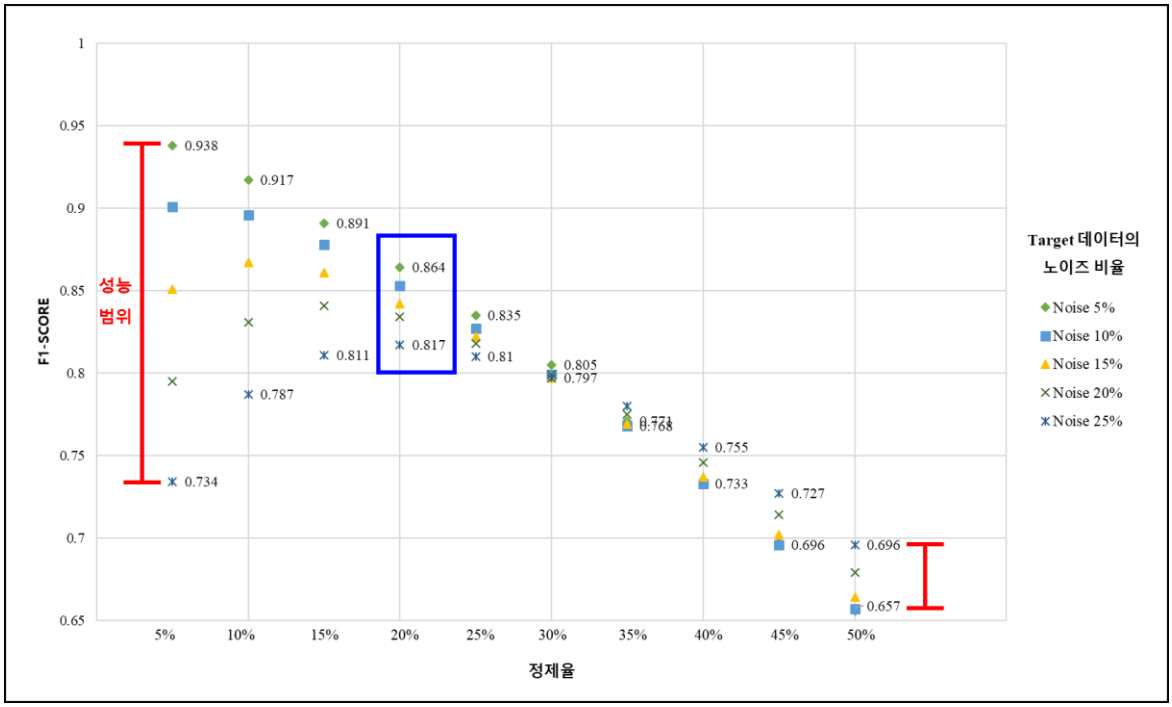
[표 2] 실험 결과 요약

[Table 2] Summary of Experiment Results

제거할 데이터 개수 (40,000 중 α%)	Noise 5% (2,000개) F1-score	Noise 10% (4,000개) F1-score	Noise 15% (6,000개) F1-score	Noise 20% (8,000개) F1-score	Noise 25% (10,000개) F1-score	Average F1-score	Std F1-score
2,000 (5%)	0.938	0.901	0.851	0.795	0.734	0.844	0.082
4,000 (10%)	0.917	0.896	0.867	0.831	0.787	0.860	0.052
6,000 (15%)	0.891	0.878	0.861	0.841	0.811	0.856	0.032
8,000 (20%)	0.864	0.853	0.842	0.834	0.817	0.842	0.018
10,000 (25%)	0.835	0.827	0.822	0.818	0.81	0.822	0.009
12,000 (30%)	0.805	0.799	0.797	0.799	0.797	0.799	0.003
14,000 (35%)	0.771	0.768	0.769	0.775	0.780	0.773	0.005
16,000 (40%)	0.735	0.733	0.737	0.746	0.755	0.741	0.009
18,000 (45%)	0.697	0.696	0.702	0.714	0.727	0.707	0.013
20,000 (50%)	0.656	0.657	0.664	0.679	0.696	0.670	0.017

실험 결과 정제율이 10%일 때 평균 F1-score가 가장 높게 나타났으며, 이때 F1-score의 최대값과 최소값은 각각 0.917와 0.787로 매우 양호하게 나타났다. 이는 정제율 10%의 기준으로 제안 방법론을 적용하는 경우, 노이즈 제거에 대한 F1-score가 최소 0.787

이상으로 유지됨을 의미한다. 한편 정제율이 30%일 때 편차가 가장 작게 나타났으며, 이는 해당 정제율이 가장 안정적인 성능을 나타내는 것으로 이해할 수 있다. 이러한 결과는 [표 2]를 도식화한 [그림 6]을 통해 시각적으로 확인할 수 있다. 제안 방법론의 실제 적용 과정에서는 Target 데이터의 노이즈 비율이 알려져 있지 않으므로 [그림 6]의 성능 범위를 참고하여 정제율을 결정해야 하며, 예를 들어 파란색 사각형으로 표시된 20%의 정제율을 채택할 경우 Target 데이터의 노이즈 포함 비율에 무관하게 F1-score 기준 최소 0.817 ~ 최대 0.864의 성능으로 노이즈를 제거할 수 있다.



[그림 6] 노이즈 비율 및 정제율에 따른 성능(F1-score) 분포

[Fig. 6] Performance(F1-score) Distribution According to Noise Ratio and Refinement Rate

5. 결론

본 연구는 기계 번역 학습용 데이터의 품질을 높이기 위해, 벡터 정렬 기술을 활용하여 원문 언어와 대상 언어 간 의미 유사성이 낮은 데이터를 제거하는 방안을 제시하였다. 구체적으로 제안 방법론은 고품질 한/영 데이터를 사용하여 영어 문장 벡터를 한글 문장 벡터로 변환하는 딥러닝 기반 벡터 정렬 모델을 학습하고, 학습된 모델을 사용하여 새로 주어진 한/영 데이터의 영어 문장 벡터를 한글 벡터 공간으로 정렬한 뒤, 정렬된 벡터와 한/영 데이터의 한글 문장 벡터와의 유사성이 낮은 경우 이를 노이즈로 간주하여 제거한다. 제안 방법론의 성능은 노이즈 데이터를 얼마나 정확하게 식별하는지의 정도로 측정할 수 있다. 실제 데이터에 대해 제안 방법론을 적용한 실험 결과 제안 방법론을 통해 노이즈 데이터를 매우 정확하게 식별해 낼 수 있음을 확인하였으며, 특히 데이터 중 10%를 정제했을 때의 성능이 F1-score 기준 평균 0.860으로 가장 우수하게 나타났다.

기계 번역 학습용 데이터 품질의 중요성은 널리 알려져 있지만, 학습용 데이터의 품질 향상을 위한 구체적인 방안은 충분히 다루어지지 않았다. 본 연구는 이러한 연구 문제를 강조하고 이를 해결하기 위한 방안을 제시했다는 점에서 학술적 기여를 인정받을 수 있으며, 향후 제안 방법론의 체계화 및 고도화를 위한 더욱 많은 후속 연구가 이루어질 것으로 기대한다. 또한 제안 방법론은 다국어 번역을 다루는 여러 응용이 공통적으로 갖는 문제인 번역 데이터의 품질 문제를 해결함으로써 이와 관련된 다양한 제품 및 서비스의 품질 향상에도 기여할 것으로 기대한다. 제안 방법론은 기계 번역 외에도 문서 요약과 같이 두 문서의 의미적 유사성이 유지되어야 하는 다양한 응용의 학습 데이터 품질 개선에 폭넓게 사용될 수 있으며, 이는 본 연구의 실무적 측면의 기여로 인정받을 수 있다. 본 연구의 실험에서는 영문과 한글로 구축된 병렬 데이터를 사용하여 실험을 수행하였으나, 향후 다양한 언어 조합에 대한 실험을 통해 더욱 엄밀한 성능 평가가 이루어져야 한다. 또한 영문 문장을 입력으로 받아서 한글을 추론하는 경우와 한글 문장을 입력으로 받아서 영문을 추론하는 경우, 제안 방법론을 통한 성능 향상에 어떠한 변화가 발생하는지도 확인할 필요가 있다. 본 연구에서는 노이즈 데이터를 얼마나 정확하게 식별하는지를 측정하는 방식으로 제안 방법론의 성능을 평가하였다. 향후 제안 방법론을 적용한 데이터와 그렇지 않은 데이터를 사용하여 각각 기계 번역 학습을 진행하고, 각 모델을 통해 수행한 기계 번역의 품질을 BLEU Score를 통해 평가하는 등의 추가 실험이 후속 연구에서 다루어질 필요가 있다.

References

[1] W. Ammar, G. Mulcaire, Y. Tsvetkov, G. Lample, C. Dyer, N. A. Smith, Massively Multilingual Word Embeddings, ArXiv, (2016)
DOI: https://doi.org/10.48550/arXiv.1602.01925

[2] I. Suskever, O. Vinyals, Q. V. Le, Sequence to Sequence Learning with Neural Networks, ArXiv, (2014)
DOI: https://doi.org/10.48550/arXiv.1409.3215

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is All You Need, ArXiv, (2017)
DOI: https://doi.org/10.48550/1706.03762

[4] H. Khayrallah, P. Koehn, On the Impact of Various Types of Noise on Neural Machine Translation, Proceedings of the 2nd Workshop on Neural Machine Translation and Generation, (2018), pp.74-83.
DOI: https://doi.org/10.18653/v1/W18-2709

[5] H. Moon, C. Park, S. Eo, J. Park, H. Lim, Filter-mBART based Neural Machine Translation using Parallel Corpus Filtering, Journal of the Korea Convergence Society, (2021), Vol.12, No.5, pp.1-7.
DOI: https://doi.org/10.15207/JKCS.2021.12.5.001

[6] R. Xu, Y. Yang, N. Otani, Y. Wu, Unsupervised Cross-lingual Transfer of Word Embedding Spaces, ArXiv, (2018)
DOI: https://doi.org/10.48550/arXiv.1809.03633

[7] M. Biesialska, M. R. Costa-jussà, Refinement of Unsupervised Cross-lingual Word Embeddings, ArXiv, (2020)
DOI: https://doi.org/10.48550/arXiv.2002.09213

[8] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, B. Frey, Adversarial Autoencoders, ArXiv, (2016)
DOI: https://doi.org/10.48550/arXiv.1511.05644

[9] J. W. Kim, B. H. Yoon, N. G. Kim, Nonlinear Vector Alignment Methodology for Mapping Domain-Specific Terminology into General Space, Journal of Intelligence and Information Systems, (2022), Vol.28, No.2, pp.127-146.

DOI: https://doi.org/10.13088/jiis.2022.28.2.127

[10] J. Devlin, M. W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, ArXiv, (2018)
DOI: https://doi.org/10.48550/arXiv.1810.04805