



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0033614
(43) 공개일자 2022년03월17일

- | | |
|--|--|
| <p>(51) 국제특허분류(Int. Cl.)
G06F 40/58 (2020.01) G06F 40/279 (2020.01)
G06N 3/08 (2006.01)</p> <p>(52) CPC특허분류
G06F 40/58 (2020.01)
G06F 40/279 (2020.01)</p> <p>(21) 출원번호 10-2020-0114913
(22) 출원일자 2020년09월08일
심사청구일자 2020년09월08일</p> | <p>(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)</p> <p>(72) 발명자
임희석
서울특별시 성북구 안암로 145 자연계캠퍼스애기
능생활관 311호 NLP&AI연구실</p> <p>박찬준
서울특별시 송파구 중대로 24, 306동 1303호(문정동, 올림픽훼밀리타운)</p> <p>(74) 대리인
김홍석</p> |
|--|--|

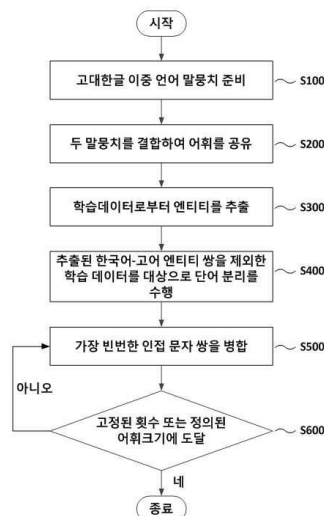
전체 청구항 수 : 총 4 항

(54) 발명의 명칭 기계 번역을 이용한 고대한글 번역 방법

(57) 요약

기계 번역을 이용한 고대한글 번역 방법은 학습 데이터로부터 한글-고대한글 엔티티 쌍(Korean-Ancient Korean entity pair)을 추출하는 단계 및 상기 추출된 한국어-고어 엔티티 쌍을 제외한 학습 데이터를 대상으로 단어 분리(subword tokenization)를 수행하는 단계를 포함한다.

대표도 - 도1



(52) CPC특허분류
G06N 3/08 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711116144
과제번호	2018-0-01405-003
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성(R&D)
연구과제명	Human-inspired 복합지능 원천기술 연구 및 인력양성
기 여 율	1/1
과제수행기관명	고려대학교산학협력단
연구기간	2020.01.01 ~ 2020.12.31

명세서

청구범위

청구항 1

학습 데이터로부터 한글-고대한글 엔티티 쌍(Korean-Ancient Korean entity pair)을 추출하는 단계; 및
상기 추출된 한국어-고어 엔티티 쌍을 제외한 학습 데이터를 대상으로 단어 분리(subword tokenization)를 수행하는 단계를 포함하는 기계 번역을 이용한 고대한글 번역 방법.

청구항 2

제1항에 있어서,

상기 한글-고대한글 엔티티 쌍에서 엔티티는 인명, 지명, 또는 기관 중 하나를 포함하는 기계 번역을 이용한 고대한글 번역 방법.

청구항 3

제1항에 있어서,

상기 한글-고대한글 엔티티 쌍은 "한글(고대한글)" 구조로 상기 학습 데이터 내에 구성되는 기계 번역을 이용한 고대한글 번역 방법.

청구항 4

제1항에 있어서,

상기 단어 분리는 바이트 쌍 인코딩(byte pair encoding, BPE)을 사용하는 기계 번역을 이용한 고대한글 번역 방법.

발명의 설명

기술 분야

[0001] 본 발명은 바이트 쌍 인코딩(byte pair encoding, BPE)을 사용하는 번역 방법에 관한 것으로, 보다 상세하게는 기계 번역을 이용한 고대한글 번역 방법에 관한 것이다.

배경 기술

[0002] 한국고전번역원은 우리 조상들의 정신문화를 담고 있는 한국고전의 수집, 정리, 번역을 통해 한국학 연구의 기반을 구축하고 나아가 전통문화를 계승, 발전시키기 위하여 2007년 11월 설립된 교육부 산하 기타공공기관으로 고전문헌을 수집·정리·번역함으로써 한국학 연구의 기반을 구축하고 전통문화를 계승·발전시키는데 노력을 다하고 있다. 현재 한국에서 해당 기관을 통해 국가적인 차원에서 고전번역에 대한 연구 및 복원 작업을 지속하고 있다. 대개 조선왕조실록, 승정원일기와 같은 역사서에 대한 번역 작업을 지속하고 있으며 대부분 사람이 번역을 진행하고 있다.

[0003] 고전번역이란 조선왕조실록, 승정원일기와 같은 고어를 번역하는 것을 의미한다. 고전번역은 고어를 번역하는 작업이다 보니 많은 시간과 비용이 투자된다. 사람이 고어를 번역하는 작업은 전문가의 손길을 필요로 하며 사람마다 번역하는 내용도 다르기에 정확한 번역문을 내놓는 것은 쉽지 않다. 이러한 문제점을 극복하기 위하여 사람과 비슷한 수준의 고품질의 기계번역기를 만들어 시간과 비용을 단축할 수 있다. 인공지능 기술을 고전문헌 번역에 적용함으로써 고전문헌 연구, 역사 연구에 획기적인 변화를 일으킬 수도 있다. 인공지능이 인간 번역자의 수고를 덜어줄 수 있다면 그만큼 번역에 드는 시간을 단축함과 동시에 번역의 질을 올릴 수 있다. 더 나아가 규장각 도서 등 아직 미번역 상태의 방대한 고전문헌들의 번역에도 인공지능이 크게 기여할 수 있을 것이다. 본 발명은 이러한 고전문헌번역의 필요성에 입각하여 고전번역에 특화된 인공지능경망 기계번역의 초기연구를 진행하였다.

[0004] 고전번역의 학술적 활용은 분명히 증가하고 있는 추세다. 특히, 번역된 자료들의 DB화가 진척되면서, 그 활용추세가 증가함이 눈에 띄게 나타나고 있다. 과거에는 주로 인문학분야에서 주로 고전을 활용했다면, 현재는 전통적인 인문학의 범위를 넘어서서 사회과학 분야와 자연학, 기술공학 및 예체능 분야의 학문에서도 활용이 되고 있다.

[0005] 이러한 필요성에 입각하여 세계적으로 고전을 번역하려는 움직임이 나타나고 있으나 해당 작업을 위해서는 전문가의 손길이 필요하며 이러한 전문가를 양성하는데도 어려움이 존재하며 무엇보다 번역을 진행하는데 시간이 많이 소요된다. 이러한 문제점을 바탕으로 최근 기계번역 기술을 활용하여 고전문자를 복구하려는 시도가 최근에 연구되고 있으나 현재 한국 고전번역 관련하여 연구 발표된 연구가 존재하고 있지 않다.

선행기술문헌

특허문헌

- [0006] (특허문헌 0001) 대한민국 공개특허 제2012-0046409호 (2012.05.10. 공개)
(특허문헌 0002) 대한민국 공개특허 제2002-0028199호 (2002.04.16. 공개)
(특허문헌 0003) 대한민국 공개특허 제2014-0059877호 (2014.05.19. 공개)

발명의 내용

해결하려는 과제

[0007] 본원 발명이 해결하고자 하는 과제는 기계 번역을 통해 고대한글을 효율적으로 번역하기 위한 방법에 관한 것이다.

과제의 해결 수단

[0008] 해결하고자 하는 과제를 달성하기 위하여 본 발명의 실시 예들에 기계 번역을 이용한 고대한글 번역 방법은, 학습 데이터로부터 한글-고대한글 엔티티 쌍(Korean-Ancient Korean entity pair)을 추출하는 단계, 및 상기 추출된 한국어-고어 엔티티 쌍을 제외한 학습 데이터를 대상으로 단어 분리(subword tokenization)를 수행하는 단계를 포함한다.

발명의 효과

[0009] 본 발명의 실시 예들에 따르면, 딥 러닝(deep learning)을 통한 신경망 기계 번역이 인간 번역자의 수고를 덜어 줄 수 있다면 그만큼 번역에 드는 시간을 단축함과 동시에 번역의 질을 올릴 수 있다. 더 나아가 미번역 상태의 방대한 고전문헌들의 번역에도 인공지능이 크게 기여할 수 있을 것이다.

[0010] 번역 완료된 고전 문헌들의 품질 편차를 최소화하고 일관된 번역 품질을 만들어 낼 수 있으며, 플랫폼을 통한 번역결과물의 데이터 베이스화 및 지식증강형 무제한 학습(Infinite Training) 모델을 구축할 수 있다.

[0011] 또한, 번역된 데이터가 데이터베이스(DB)로 전환이 진행된다면, 사회 과학, 자연 과학, 및 공학 분야는 물론 예술 및 물리 분야에서도 활용될 수 있다.

도면의 간단한 설명

[0012] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.

도 1은 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법을 설명하기 위한 순서도이다.

도 2는 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법을 설명하기 위한 모식도이다.

발명을 실시하기 위한 구체적인 내용

[0013] 본 발명의 구성 및 효과를 충분히 이해하기 위하여, 첨부한 도면을 참조하여 본 발명의 바람직한 실시 예들을 설명한다. 그러나 본 발명은, 이하에서 개시되는 실시 예들에 한정되는 것이 아니라, 여러 가지 형태로 구현될

수 있고 다양한 변경을 가할 수 있다.

- [0014] 또한, 본 발명의 실시 예들에서 사용되는 용어들은 다르게 정의되지 않는 한, 해당 기술 분야에서 통상의 지식을 가진 자에게 통상적으로 알려진 의미로 해석될 수 있다.
- [0015] 이하에서는 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법을 첨부된 도면을 참조하여 설명하기로 한다.
- [0016] 본 발명에서의 기계 번역은 신경망 기계 번역(Neural Machine Translation, NMT)으로, 트랜스포머(Transformer) 기반 네트워크를 사용하여 수행될 수 있다.
- [0017] 이하, 트랜스포머(Transformer)에 대하여 간략하게 설명한다. 순환 신경망(recurrent neural network) 기반 NMT 모델은 X와 Y가 각각 소스 문장과 대상 문장으로 설정될 때 $P(Y | X)$ 를 최적화할 수 있다. 그러나 인코더의 마지막 은닉 상태(hidden state)는 컨텍스트 벡터(context vector)로 사용되며, 고정 벡터로 인해 긴 문장 처리가 어렵다. 이 문제를 극복하기 위해 어텐션(attention)의 개념에 제안된다. 어텐션은 마지막 은닉 상태를 고려할 뿐만 아니라 모든 입력 상태의 가중치 조합을 고려한다. 즉, 어텐션은 매 시간마다 컨텍스트 벡터를 동적으로 생성하여 시스템이 단어의 중요한 부분에 집중하도록 돕는 기술이다. 구글(Google)에서 어텐션만 사용하는 트랜스포머(Transformer)라는 모델을 개발했고, 이전 모델보다 성능이 우수하다. 트랜스포머(Transformer)는 합성곱(convolution)이나 순환(recurrent) 없이 어텐션만 사용하는 네트워크이다. 따라서 학습 시간이 매우 빠르다. 한편, 트랜스포머(Transformer)의 전체 아키텍처는 누적된 멀티헤드 어텐션(multihead attention) 및 포인트 별 피드 포워드 레이어(pointwise feedforward layers)를 사용하여 구성된다.
- [0018] 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법에서는, 트랜스포머(Transformer) 모델의 구조를 변경하지 않고 기계 번역의 전처리 단계 중 하나인 단어 분리(subword tokenization)를 적용할 수 있다. 단어 분리는 어휘 외 문제와 알려지지 않은 문제를 해결하기 위한 것으로, 통상적으로 NMT의 전처리로 적용하고 있다. 단어 분리 방법들로는 바이트 쌍 인코딩(byte pair encoding, BPE) 및 SentencePiece가 있으며, 이들은 기본적으로 단어가 여러 하위 단어의 조합으로 구성되어 있다고 가정하는 알고리즘이다. BPE는 BPE 알고리즘을 기반으로 단어 분리를 수행하고 SentencePiece는 유니그램(unigram) 언어 모델을 기반으로 단어 분리를 수행할 수 있다.
- [0019] 단어 분리(subword tokenization)는 하나의 단어가 여러 개의 의미 있는 서브 워드의 조합으로 구성되는 경우가 많기 때문에 하나의 단어를 여러 개의 서브 워드로 나누어 단어를 표현하는 것을 목표로 하는 전처리이다. BPE는 전체 문장을 하위 단어 단위로 분할하여 어휘를 줄이는 방법이며, 이를 통해 알 수 없는 토큰(unknown token)을 효율적으로 처리할 수 있다. 이러한 단어 분리를 통해 새로 만든 단어 또는 인쇄상의 오류와 같은 UNK를 하위 단어와 문자로 나누어 알려진 토큰의 조합을 만들 수 있다. 이를 통해 UNK를 제거하고 NMT의 품질을 향상시킴으로써 UNK를 효율적으로 처리할 수 있다.
- [0020] 본 발명의 일 실시 예에 따르면, 단어 분리 방법으로 BPE를 선택한다. 즉, 트랜스포머(Transformer) 모델의 구조를 변경하지 않고 BPE가 적용된 기계 번역을 이용한 고대한글 번역 방법을 개시한다.
- [0021] 한편, 고대 번역에서 중요한 요소 중 하나는 엔티티(entity)가 얼마나 잘 번역되는지 이다. 고대 번역 자료는 사람들의 이름, 장소 또는 기관이 문장의 대부분을 차지한다. 당시에는 기록을 남기는 것이 중요했기 때문에 엔티티에 대한 정보가 매우 중요하다. 고대 번역의 이러한 특징을 설명하기 위해 단어 분리 단계에서 엔티티 정보가 제한될 수 있다. 본 발명의 일 실시 예에 따르면, BPE를 기반으로 한글과 고대 한글의 어휘를 공유하면서 엔티티 정보를 제한한다. 어휘 공유(sharing vocabulary)가 엔티티 정보를 복사할 때 문자를 삽입하거나 삭제할 필요를 감소시킬 수 있다. 따라서, 엔티티 정보가 중요한 고대 한국어 문장에 적합한 방법이다. 한편, 엔티티는 학습 데이터에서 자동으로 추출되며 엔티티가 입력된 학습 데이터의 구조 "()"에서 확인할 수 있다. 일 예로, 학습 데이터 내에서 괄호(()) 내에 기재된 단어는 엔티티를 의미할 수 있다.
- [0022] 도 1은 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법을 설명하기 위한 순서도이다. 도 1을 통해 설명되는 고대한글 번역 방법은 적어도 프로세서 및/또는 메모리를 포함하는 컴퓨팅 장치에 의해 수행되는 방법을 의미할 수 있다. 또한, 실시예에 따라 상기 고대한글 번역 방법은 고대한글 번역 모델 생성 방법으로 명명될 수도 있다.
- [0023] 도 1을 참조하면, 고대 한글 이중 언어 말뭉치(ancient Korean bilingual corpus)를 준비하고(S100), 두 말뭉치를 결합하여 어휘를 공유할 수 있다(S200). 이렇게 학습 데이터를 형성하고, 학습 데이터로부터 한글-고대 한글 엔티티 쌍(Korean-Ancient Korean entity pair)을 추출할 수 있다(S300). 전술한 바와 같이, 한글-고대 한글

엔티티 쌍에서 엔티티는 인명, 지명, 또는 기관 중 하나를 포함할 수 있다. 그리고, 한글-고대한글 엔티티 쌍은 "한글(고대한글)" 구조로 상기 학습 데이터 내에 구성될 수 있다.

[0024] 이어서, 하위 단어 어휘 크기(subword vocabulary size)를 설정한다. 그리고, 단어를 시퀀스 문자(sequence characters)로 분할한다(단어 분리: subword tokenization). 이때, 엔티티는 분할하지 않도록 제한한다. 즉, 추출된 한국어-고어 엔티티 쌍을 제외한 학습 데이터를 대상으로 단어 분리(subword tokenization)를 수행할 수 있다(S400).

[0025] 이어서, 가장 빈번한 인접 문자 쌍을 병합한다(S500). 고정된 횟수 또한 정의된 어휘 크기에 도달할 때까지 상기의 단계들을 반복한다(S600).

[0026] 도 2는 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법을 설명하기 위한 모식도이다.

[0027] 도 2를 참조하면, 학습 데이터를 구축한 후, 학습은 sequence-to-sequence 모델을 사용하여 수행될 수 있다. 본 실시 예에서 트랜스포머(Transformer)를 사용할 때, 인코더(encoder)는 6개의 층으로 구성되고 각 층은 self-attention 및 feedforward network로 구성된다. 디코더(decoder)는 또한 6개의 층으로 구성되며 각 층은 masked self-attention, encoder-decoder self-attention, 및 feedforward networks로 구성된다.

[0028] 고대 한글 문장이 모델에 입력되면 고대 한글 임베딩(Korean embedding)을 사용하여 벡터화된다. 인코더를 통해 모델은 고대 한국어 문장의 잠재 표현을 학습하고 디코더를 사용하여 현대 한국어로 번역한다. 어텐션의 기본 개념은 모든 시간 단계에서 디코더가 출력 단어를 예측하고 인코더의 전체 입력 문장을 다시 한 번 참조한다는 것이다. 그러나 모든 입력 문장이 동일한 가중치로 참조되는 것이 아니라, 해당 시점에 예측할 단어와 관련된 입력 텍스트 부분이 더 집중된다. 셀프-어텐션 메커니즘(self-attention mechanism)을 통해 네트워크는 쿼리(query), 키(key) 및 값 벡터(value vector)를 사용하여 현재 단어와 이전 단어 간의 상관 관계를 학습할 수 있다. 이 메커니즘을 "고대 어텐션(ancient attention)" (scaled dot-product attention와 동일)라고 하며 다음과 같이 총 6 단계로 구성된다.

[0029] 고대 쿼리(ancient query, An_Q), 고대 키(ancient key, An_K) 및 고대 값(ancient value, An_V) 등의 세 개의 벡터들을 형성하고, 고대 쿼리와 고대 키를 곱하여 점수를 결정한다. 스케일을 진행하되, 64의 고대 키 벡터 크기를 제곱근으로 나눈다. 현재 위치에서 단어를 인코딩할 때 표현되는 단어 수의 비율인 softmax를 계산한다. 이상의 단계까지의 결과에 고대 값(ancient value, An_V)을 곱하는데, softmax의 높은 값과 낮은 값은 무시된다. 결과 벡터를 결합하며, 이것이 하기 수학적 1의 셀프-어텐션(self-attention)의 결과이다.

[0030] [수학적 1]

$$AA(An_Q, An_K, An_V) = \text{softmax}\left(\frac{An_Q * An_K^T}{\sqrt{d_{An_K}}}\right) An_V$$

[0031]

[0032] 이때, AA는 고대 어텐션을 의미한다.

[0033] 한편, 멀티헤드 고대 어텐션(Multihead Ancient Attention)은 8개의 다른 가중치 행렬에 대해 고대 어텐션을 8번 수행하는 프로세스하는 것으로 수학적 2를 통해 고대 어텐션을 획득할 수 있다.

[0034] [수학적 2]

$$\text{MultiHeadAA}(An_Q, An_K, An_V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h) W^O$$

[0035]

[0036] 이때, $\text{head}_i = AA(An_Q W_i^{An_Q}, An_K W_i^{An_K}, An_V W_i^{An_V})$ 이다.

[0037] 이어서, 디코더를 통해 훈련된 인코더의 정보를 바탕으로 한글 번역을 수행할 수 있다.

[0038] 디코더의 셀프-어텐션(self-attention) 층은 출력 시퀀스에서 현재 위치의 이전 위치에만 집중한다. 이는 디코더에서는 인코딩과 달리 결과가 순차적으로 생성되어야 하기 때문이다. 또한, 쿼리 행렬(query matrices, K_{qq})은 기본 층(underlying layer)에서 가져오고 키 및 값 행렬(key and value matrices, An_K^T)은 인코더의 출력에서 가져올 수 있다. 즉, 고대 한글 문장과 한글 문장 사이의 어텐션이 처리된다.

[0039] 출력에 대한 고대 한글 어텐션 행렬($AK_{\text{Attention}}$)을 다음의 수학적 식 3과 같이 계산할 수 있다.

[0040] [수학적 식 3]

$$AK_{\text{Attention}}(K_{oQ}, A_{nK}, A_{nV}) = \text{softmax}\left(\frac{K_{oQ} * A_{nK}^T}{\sqrt{d_{A_{nK}}}}\right) A_{nV}$$

[0041]

[0042] 로짓(logit)은 디코더를 통해 구축되며, Softmax는 다음 토큰 확률을 예측하기 위해 최적의 출력을 도출한다. 요약하면, 하기의 수학적 식 4에서와 같이, 인코더와 디코더는 조건부 로그 우도(conditional log-likelihood)를 최대화하기 위해 공동으로 훈련될 수 있다.

[0043] [수학적 식 4]

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^N \log p_{\theta}(K_n | A_n)$$

[0044]

[0045] 모델 매개 변수의 집합으로써, 각 (A_n, K_n)은 학습 데이터의 (고대 한글 문장, 한글 문장) 쌍이다.

[0047] 이하, 본 발명의 일 실시 예에 따른 기계 번역을 이용한 고대한글 번역 방법을 이용한 경우와, 기존의 방법을 이용한 경우를 실험을 통해 BLEU(bilingual evaluation understudy) 점수로 성능 평가하는 것을 설명한다.

[0048] 데이터 설정

[0049] 고대 한국어 번역을 위한 학습 데이터를 얻는 것은 쉽지 않다. 유일한 선택은 한국 고전 번역원에서 제공하는 고대 DB를 이용하는 것이다. 따라서 데이터를 크롤링(crawling)하고, 간단한 데이터 필터링을 수행했다. 길이가 80자 미만인 텍스트는 학습에서 제외되었고 총 52,778 개의 고대 한국 평행 말뭉치가 작성되었다. 고대 한국 평행 말뭉치는 한국어 고어(고대 한글)와 이에 대한 번역(한글)을 포함하는 개념으로 이해될 수 있다.

[0050] 학습 데이터에서, 장소나 사람의 이름과 같은 엔티티 정보가 문장의 대부분을 차지한다. 즉, 고대 한국어 번역에서는 엔티티 정보를 있는 그대로 번역하는 것이 매우 중요하다.

[0051] 테스트 세트를 만들기 위해 전체 학습 세트에서 3,000 개의 문장을 무작위로 추출하고, 5,000 개의 문장을 무작위로 추출하여 유효성 검사 세트를 생성했다.

[0052] 모델 및 하이퍼파라미터

[0053] LSTM(Long short-term memory) 어텐션 모델(기존의 방법) 과 트랜스포머(Transformer) 기반 모델(본 발명의 일 실시 예에 따른 방법)을 사용하여 고대 한국어 번역에 NMT를 적용했다. 또한, 다양한 단어 분리에 따른 성능 변화를 확인했다.

[0054] LSTM 어텐션 모델의 경우 2층 LSTM이 사용되었고 Luong 어텐션이 어텐션에 사용되었다. 또한 은닉 유닛은 500, 드롭 아웃 비율은 0.2, 미니 배치 크기는 128이었다.

[0055] 트랜스포머(Transformer)에는 인코더와 디코더에 6개의 어텐션 블록이 있었고 피드 포워드 차원은 2,048, 임베딩 크기는 512였다. 8개의 어텐션 헤드를 사용했으며 최적화 프로그램에는 Adam과 Noam decay가 사용되었다. 배치 크기는 4,096이었다.

[0056] 각 모델은 32,000의 어휘 크기, 교차 엔트로피 손실 함수 및 Adam 최적화 프로그램을 사용했다. 두 개의 GTX 1080을 사용하여 GPU 훈련을 수행했다. 번역을 위한 빔 크기(beam size)는 5로 고정되었다. 모든 번역 결과는 Moses Multibleu. perl script에서 계산한 BLEU를 사용하여 평가되었다.

[0057] 결과

[0058] 1) 단어 분리 평가

[0059] 단어 분리가 어떻게 처리되는지 평가하기 위해 다양한 실험이 수행되었다. 문자 단위 BPE, SentencePiece, 및 본 발명의 일 실시 예에 따른 방법(이하, SVER BPE: Share Vocab and Entity Restrict Byte Pair Encoding이라 한다)을 이용하여 단어 분리 처리 후 실험 결과를 비교하였다. 또한, 엔티티만 제한했을 때 성능이 어떻게 변화

는지 관찰했다. 실험 결과는 표 1를 참조한다.

[표 1]

Model	BLEU	Tokens per Second
SentencePiece-LSTM Attention	24.39	2,758
SentencePiece-Transformer	22.69	982
BPE-LSTM Attention	25.18	2,029
BPE-Transformer	24.43	1,122
Char- LSTM Attention	23.66	8,785
Char- Transformer	16.24	1,466
Entity Restrict- LSTM Attention	14.74	3,013
Entity Restrict- Transformer	15.12	1,174
(Ours) SVER BPE - LSTM Attention	29.40	5,004
(Ours) SVER BPE - Transformer	29.68	1,379

표 1에 따르면, 단어 분리가 수행되는 방식에 따라 성능이 크게 달라지는 것을 볼 수 있다.

하위 단어 분리가 엔티티 제한으로만 구성되었을 때 성능이 가장 낮았다. 즉, 엔티티를 제한하면 성능에 부정적인 영향을 미친다는 것을 알 수 있다. 문자 단위가 사용되었을 때, Transformer 모델은 LSTM 모델보다 훨씬 더 나은 성능을 보였으며 문자 단위의 임베딩이 Transformer에 부정적인 영향을 미친다는 것을 알 수 있다. SVER BPE는 최상의 성능을 보였다. SVER BPE에서, 단어 분리를 적용한 LSTM 어텐션 모델도 단순 BPE를 사용한 Transformer 모델보다 더 나은 성능을 보였다. Transformer 모델을 비교해도 SVER BPE은 BPE보다 5.25점 더 높은 BLEU 점수를 얻었다.

이러한 결과는 모델이 동일하더라도 단어 분리가 수행되는 방식에 따라 성능이 크게 달라질 수 있음을 의미한다. 이것은 고대 한국어 번역뿐 아니라 기계 번역의 일반적인 분야에서도 마찬가지이다. 따라서 이 실험 결과는 고대 언어 번역 분야에만 국한되지 않는다고 할 수 있다. 또한 단어 분리가 수행되는 방식에 따라 토큰 처리 속도가 분명하게 변하는 것으로 볼 수 있다. SVER BPE은 기존 BPE 및 SentencePiece 방법보다 빠르게 디코딩을 수행하였다. 요약하면, SVER BPE은 고대 한국어 번역의 성능 향상뿐만 아니라 번역 속도 향상에도 도움이 된다는 것을 확인했다.

2) 빔 사이즈(beam size) 평가

빔 검색(beam search)은 디코딩 중에 빔 크기의 가장 큰 확률을 계속해서 찾는 알고리즘이다. 빔의 크기를 변경하면서 성능 비교를 수행했다. 이 실험의 모든 모델은 Transformer를 기반으로 수행하였다. 단어 분리를 위해 본 발명의 일 실시 예에 따른 SVER BPE이 적용되었다. 실험 결과는 표 2와 같으며, 빔 크기가 3일 때 최상의 성능을 얻으며, 이는 최대 0.58 BLEU 포인트의 차이를 보였다.

[표 2]

Beam Size	BLEU
1	28.79
2	28.79
3	29.88
4	28.81
5	29.68
6	29.70
7	29.59
8	29.48
9	29.44
10	29.30

3) N-gram 반복 차단(repeat blocking) 평가

N-gram 차단은 디코딩이 n번 반복될 때 특정 단어를 차단하는 방법이다. N- 그램 반복 차단은 빔 검색이 반복되는 n-gram hyston2011 효율적이다. 실험은 빔 크기 3으로 수행되었다. 표 3에서 볼 수 있듯이, n-gram 차단을 적용하여 성능이 약간 향상되었다. 성능은 8 그램 차단을 적용하여 0.04 BLEU 포인트를 향상시켰다.

[0071] [표 3]

N-gram block	BLEU
Uni-gram	26.21
Bi-gram	29.49
Tri-gram	29.78
4-gram	29.85
5-gram	29.89
6-gram	29.90
7-gram	29.91
8-gram	29.92
9-gram	29.89
10-gram	29.89

[0072]

[0073] 4) 앙상블(ensemble) 평가

[0074] 앙상블의 주요 목적은 여러 모델의 집단 지능을 활용하여 더 나은 결과를 얻는 것이다. SVER BPE로 구축된 LSTM 어텐션 모델과 Transformer 모델을 사용하여 앙상블 모델을 만들었다. 빔 크기와 n-gram 차단은 각각 3 및 8-gram 차단으로 설정되었다. 실험 결과를 표 4에 나타내었다. 앙상블 모델을 사용했을 때 BLEU 점수가 2.65점 증가했다. 이 결과로 다양한 디코딩 전략 중 가장 높은 성능 향상을 얻는다는 것을 알 수 있다.

[0075] [표 4]

Ensemble Model	BLEU
SVER BPE (LSTM Attention + Transformer)	32.57

[0076]

[0077] 이상에서 설명된 장치는 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 집합으로 구현될 수 있다. 예를 들어, 실시 예들에서 설명된 장치 및 구성 요소는, 예를 들어, 프로세서, 컨트롤러, ALU(Arithmetic Logic Unit), 디지털 신호 프로세서(Digital Signal Processor), 마이크로 컴퓨터, FPA(Field Programmable array), PLU(Programmable Logic Unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(Operation System, OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술 분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(Processing Element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 컨트롤러를 포함할 수 있다. 또한, 병렬 프로세서(Parallel Processor)와 같은, 다른 처리 구성(Processing Configuration)도 가능하다.

[0078] 소프트웨어는 컴퓨터 프로그램(Computer Program), 코드(Code), 명령(Instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(Collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성 요소(Component), 물리적 장치, 가상 장치(Virtual Equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(Signal Wave)에 영구적으로, 또는 일시적으로 구체화(Embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

[0079] 실시 예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시 예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(Magnetic Media), CD-ROM, DVD와 같은 광기록 매체(Optical Media), 플롭티컬 디스크(Floptical Disk)와 같은 자기-광 매체(Magneto-optical Media), 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록

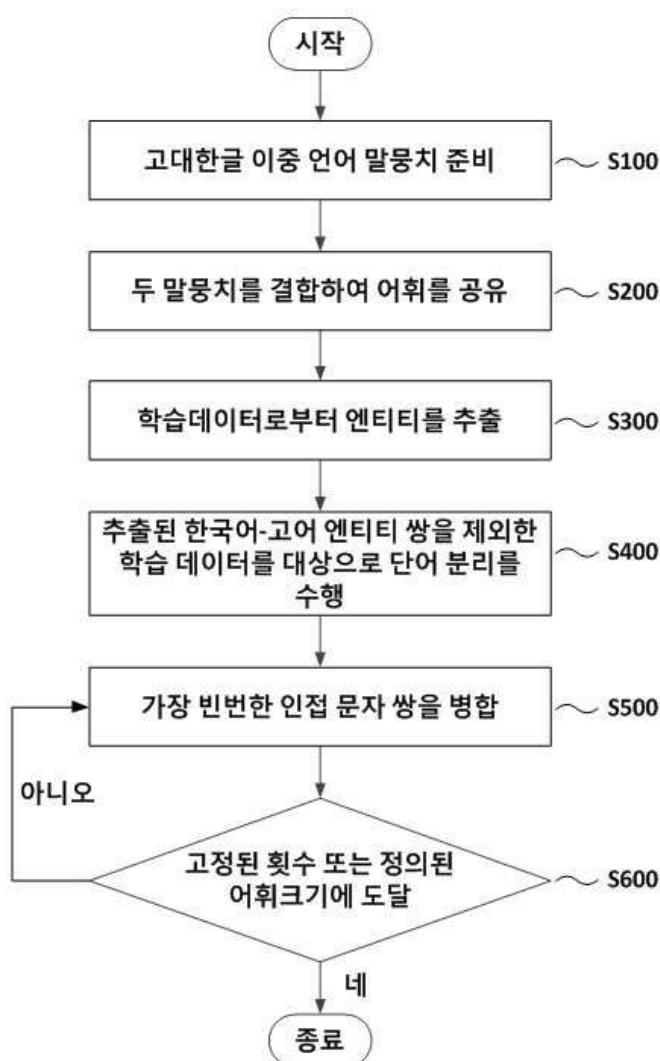
특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시 예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

[0080]

본 발명은 도면에 도시된 실시 예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시 예가 가능하다는 점을 이해할 것이다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성 요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

도면

도면1



도면2

