(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0078948
(43) 공개일자 2024년06월04일

(51) 국제특허분류(Int. Cl.)
G06F 40/58 (2020.01) *G06F 40/242* (2020.01)
G06F 40/284 (2020.01) *G06N 20/00* (2019.01)
(52) CPC특허분류
G06F 40/58 (2020.01)
G06F 40/242 (2020.01)
(21) 출원번호 10-2022-0161691
(22) 출원일자 2022년11월28일
심사청구일자 2022년11월28일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
임희석
서울특별시 성북구 안암로 145 자연계캠퍼스 애기능생활관 311호 NLP&AI연구실
이정섭
서울특별시 동대문구 무학로 184, 401호
(74) 대리인
김등용

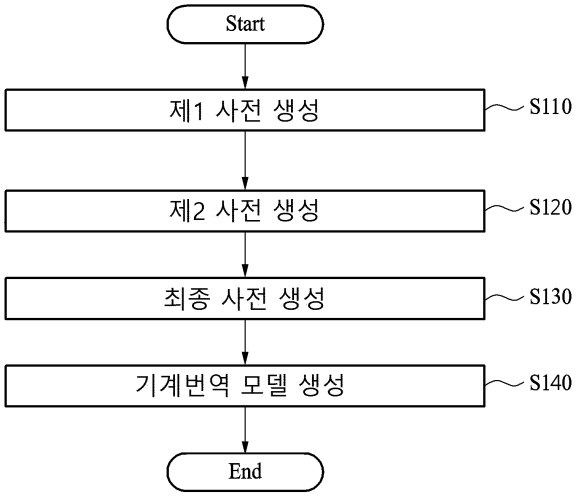
전체 청구항 수 : 총 6 항

(54) 발명의 명칭 **도메인 적응 토큰화 방법 및 장치**

(57) 요 약

도메인 적은 토큰화 방법이 개시된다. 상기 토큰화 방법은 적어도 프로세서(processor)를 포함하는 컴퓨팅 장치에 의해 수행되고, 제1 코퍼스로부터 제1 사전을 생성하는 단계, 상기 제1 코퍼스로부터 제2 사전을 생성하는 단계, 및 상기 제1 사전과 상기 제2 사전에 기초하여 최종 사전을 생성하는 단계를 포함하고, 상기 제1 사전과 상기 제2 사전은 BPE(Byte Per Encoding) 기법을 통해 생성되고, 상기 제1 사전은 미리 정해진 길이 이상의 길이를 갖는 토큰들만을 포함한다.

대 표 도 - 도2



(52) CPC특허분류
G06F 40/284 (2020.01)
G06N 20/00 (2021.08)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711159969
과제번호	2018-0-01405-005
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	Human-inspired 복합지능 원천기술 연구 및 인력양성
기 여 율	1/1
과제수행기관명	고려대학교산학협력단
연구기간	2022.01.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

적어도 프로세서(processor)를 포함하는 컴퓨팅 장치에 의해 수행되는 토큰화(tokenization) 방법에 있어서,

제1 코퍼스로부터 제1 사전을 생성하는 단계;

상기 제1 코퍼스로부터 제2 사전을 생성하는 단계; 및

상기 제1 사전과 상기 제2 사전에 기초하여 최종 사전을 생성하는 단계를 포함하고,

상기 제1 사전과 상기 제2 사전은 BPE(Byte Per Encoding) 기법을 통해 생성되고,

상기 제1 사전은 미리 정해진 길이 이상의 길이를 갖는 토큰들만을 포함하는,

토큰화 방법.

청구항 2

제1항에 있어서,

상기 제1 사전을 생성하는 단계는,

상기 제1 코퍼스를 임의의 비율로 분할하여 제1-1 코퍼스와 제1-2 코퍼스를 생성하는 단계; 및

상기 제1-1 코퍼스로부터 상기 제1 사전을 생성하는 단계를 포함하는,

토큰화 방법.

청구항 3

제2항에 있어서,

상기 제2 사전을 생성하는 단계는, 상기 제1-1 코퍼스로부터 상기 제2 사전을 생성하는,

토큰화 방법.

청구항 4

제3항에 있어서,

상기 최종 사전을 생성하는 단계는,

상기 제1 사전에 포함된 토큰들 각각에 대하여, 상기 제1-2 코퍼스에서의 출현 빈도를 산출하는 단계;

상기 제1 사전에 포함된 토큰들을 출현 빈도의 내림 차순으로 정렬하는 단계; 및

출현 빈도의 내림 차순으로, 상기 제2 코퍼스에 포함되는 토큰들만을 상기 최종 사전에 추가하는 단계를 포함하는,

토큰화 방법.

청구항 5

제4항에 있어서,

상기 최종 사전은 제2 사전에 포함된 토큰들을 더 포함하는,

토큰화 방법.

청구항 6

제5항에 있어서,

상기 토큰화 방법은 상기 제2 코퍼스를 이용하여 소정의 모델을 학습하여, 기계번역 모델을 생성하는 단계를 더 포함하고,

상기 기계번역 모델을 생성하는 단계는 상기 최종 사전을 이용하여 토큰화를 수행하는,

토큰화 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 특정 도메인의 기계번역 모델을 학습하기 위한 사전 처리에 관한 것이다.

배 경 기 술

[0002] 기계번역은 언어를 다른 언어로 번역하여 제공하는 시스템이다. Google 번역, 파파고 등의 무료 서비스로 제공되고 있는 기계번역 소프트웨어가 있지만, 이러한 무료 서비스들은 모두 도메인을 특정하지 않는 오픈 도메인(Open-domain) 서비스이다. 하지만, 의료, 법률 등의 특정 도메인에 관한 번역은 오픈 도메인 내에서 사용하는 의미를 그대로 사용할 경우 잘못된 번역을 제공할 수 있다. 특정 도메인의 데이터의 학습을 효과적으로 하기 위해, 본 발명은 언어적 의미 사상을 사전 처리를 통해 반영하고자 한다.

선행기술문헌

특허문헌

[0003] (특허문헌 0001) 대한민국 공개특허 제2007-0058950호 (2007.06.11. 공개)

(특허문헌 0002) 대한민국 공개특허 제2011-0062867호 (2011.06.10. 공개)

발명의 내용

해결하려는 과제

[0004] 기존의 기계번역 시스템에서 도메인 내 번역에 관한 연구는 용어집(Terminology) 데이터베이스 내에 엔트리를 활용하여 번역 대상의 각 단어의 특징 사상을 바탕으로 도메인 내 특징을 반영한 모델을 개발하는 연구가 주를 이루고 있다. 하지만, 도메인 내 데이터에 대한 단어를 숫자로 변환하는 변환기인 토크나이저(tokenizer)를 기존의 서브워드(subword) 기반의 토크나이저를 사용하여 도메인 내 데이터가 임베딩 레이어에 제대로 사상되지 않는다. 따라서, 본 발명은 이전 기술과는 다르게 사전 처리를 통해 도메인 내 용어에 해당하는 임베딩을 구축하는 것을 목적으로 한다.

과제의 해결 수단

[0005] 본 발명의 일 실시예에 따른 토큰화 방법은 적어도 프로세서(processor)를 포함하는 컴퓨팅 장치에 의해 수행되는 토큰화(tokenization) 방법으로서, 제1 코퍼스로부터 제1 사전을 생성하는 단계, 상기 제1 코퍼스로부터 제2 사전을 생성하는 단계, 및 상기 제1 사전과 상기 제2 사전에 기초하여 최종 사전을 생성하는 단계를 포함하고, 상기 제1 사전과 상기 제2 사전은 BPE(Byte Per Encoding) 기법을 통해 생성되고, 상기 제1 사전은 미리 정해진 길이 이상의 길이를 갖는 토큰들만을 포함한다.

발명의 효과

[0006] 본 발명은 단어를 숫자로 변환하는 사전 구축에 관한 연구로, 기존 도메인 내 기계번역 모델을 개선하는 발명으로, 이전의 연구와 병합되어 사용될 수 있다. 또한, 이전 연구와는 달리 여분의 용어집을 필요로 하지 않는다는 장점이 있다. 또한, 용어집이 완벽히 구축되지 않은 도메인 내에서도 기존 학습 데이터셋으로 도메인 용어를 사상할 수 있는 효과가 있다.

도면의 간단한 설명

[0007] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.

도 1은 본 발명의 일 실시예에 따른 사전 구축 방법에 대한 알고리즘을 도시한다.

도 2는 본 발명의 일 실시예에 따른 토큰화 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0008] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태들로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.

[0009] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.

[0010] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.

[0011] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0012] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0013] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0014] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시예들을 상세히 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0015] 본 발명은 도메인 내 번역에서 정보가 유지되어야 하는 단어를 분절하지 않도록 하는 사전을 구축하여 도메인 내 임베딩을 구축하는 것을 목적으로 한다. 도메인 내 데이터는 여러 단어 또는 형태소의 조합으로 도메인 내에서 사용되는 새로운 의미의 단어를 생성한다. 특히, 한국어에서는 형태소가 상호의존적으로 빈번히 등장하기 때문에, 빈도수 기반의 서브워드 방법으로 일반적인 크기의 사전을 구축하면 도메인 내 단어가 모두 분절되어 모델의 임베딩 레이어를 구축하게 된다.

[0016] 반면, 서브워드의 큰 크기의 사전은 빈도수 기반의 긴 단어들을 포함하므로, 도메인 내 단어를 분리하지 않고 사전에 포함시킬 수 있다. 하지만, 이러한 큰 사이즈의 사전을 사용하여 모델을 학습할 경우, 임베딩의 크기가 커지게 된다. 이는 무수히 많은 번역 데이터셋이 없는 한 임베딩 레이어 업데이트 부족으로 성능의 저하를 야기

하고, 많은 파라미터로 인하여 고자원을 필요로 한다는 단점이 존재한다. 이는 데이터셋이 부족한 도메인 내 번역에서는 더 크게 작용한다.

[0017] 이러한 이유로, 본 발명에서는 큰 크기의 사전을 구축하여 생성되는 도메인 내 단어들을 기존의 서브워드 사전에 포함하는 사전을 구축한다. 추가하는 단어는 큰 크기의 사전에서 학습 데이터셋 내에서 가장 많이 포함된 최소 길이 L 이상의 도메인 내 단어를 추출하고 이를 기존의 사전에 추가하여 다른 공간에 사상하도록 한다.

[0018] 기계번역(Machine Translation, MT)에서 워드 단위 토큰화(word-unit tokenization)는 워드의 의미적 정보(semantic information)를 잠재 임베딩 공간(latent embedding space)에 맵핑함으로써 주목할 만한 성능을 달성하였다. 이러한 효용성에도 불구하고, 최근의 연구는 워드 단위의 토큰화에 내재된 한계인 OOV(Out-Of-Vocabulary) 문제를 지적하였다. 워드를 서브워드로 분절하기 위한 몇몇의 시도들은 이러한 한계를 완화하였다. 서브워드 사전(subward vocabulary)을 구축하는 BPE(Byte Per Encoding)는 신경망 기계번역(Neural Machine Translation, NMT) 분야에서 가장 선호되는 토큰화 기법 중 하나이다.

[0019] 서브워드 토큰화의 명확한 효과는 워드 단위 토큰화의 OOV 문제를 효과적으로 완화시킬 수 있다는 것이다. 특히, 몇몇 형태소(morphemes)가 단일의 워드를 구성하는 교착어(agglutinative languages)에서의 효과가 크다.

[0020] 그러나, BPE와 같은 서브워드 토큰화 기법은 오버-토큰화(over-tokenization)의 부작용을 앓고 있다. 서브워드들의 합성 의미(synthetic meaning)는 오리지널 워드의 의미를 왜곡시킨다. 예컨대, 한국어에서, "큰코다치다(pay dearly)"의 의미는 서브워드들의 구성인 "큰(big)" + "코(nose)" + "다치다(hurt)"로 해석할 때와 완전히 달라진다.

[0021] 본 발명에서는 이러한 문제점들을 완화하기 위한 새로운 방법을 제안한다. 제안하는 기법을 구현하기 위해서, 길이기 긴 워드들을 포함하는 보조 사전(auxiliary vocabulary)를 채용한다. 작은 사전 크기를 갖는 BPE 토크나이저에 생성된 보조 사전을 추가함으로써, 새로운 토크나이저가 생성될 수 있다.

[0022] 이하에서는, 토크나이저를 위한 사전 구축 과정을 상세히 설명한다.

[0023] 새로운 워드를 생성하기 위해서, 상호의존적인 형태소들(interdependent morphemes)이 결합된다. 새로운 단어를 형성하는 과정에서, 형태소는 상호의존적으로 복수회 출현한다. 빈도수 기반 서브워드 방법(frequency-based subword method)으로 사용되는 적절한 크기(예컨대, 4k, 8k, 16k)의 사전을 구축할 수 있다면, 토큰의 평균 길이를 짧게 유지할 수 있다. 반면에, 서브워드 기반 방법에서 큰 크기의 사전(예컨대, 64k 이상)은 빈도수에 기반한 길이가 긴 토큰들을 포함한다(표 1 참조). 그러나, 큰 크기의 사전을 이용하여 모델을 학습할 때, 임베딩의 크기가 커지게 된다. 이는 무수히 많은 번역 데이터셋이 없는 한, 임베딩 레이어 업데이트 부족으로 성능의 저하를 야기하고, 많은 파라미터로 인하여 고자원을 필요로 한다는 단점이 존재한다. 따라서, 본 발명에서는 길이가 긴 토큰들을 포함하는 적적한 크기의 사전을 구축하는 방법(LeTo(Length-aware Tokenization)이라 명명될 수 있음)을 제안한다.

[0024] [표 1]

Vocab Size	# of tokens by length					% of length less than 4
	1	2	3	4	5+	
4k	1,890	1,678	341	82	5	0.978
8k	2,068	4,046	1,425	396	61	0.943
16k	2,259	7,736	4,259	1,449	293	0.891
32k	2,419	13,120	10,527	4,695	1,235	0.815
64k	2,516	20,893	23,082	12,817	4,668	0.727

[0025]

[0026] 표 1은, 스페셜 토큰들(special tokens)을 제외하고, 대용량의 단일언어 한국어 코퍼스에 의해 구축된 BPE 사전의 토큰 길이를 나타낸다. 표 1을 참조하면, 크기가 작은 사전일수록, 길이가 긴 토큰의 비율이 작아짐을 알 수 있다.

[0027] 본 명세서에서, 큰 사이즈의 서브워드 기반 사전을 V_l로, 작은 사이즈의 서브워드 기반 사전을 V_s로 나타낸다. 최종적으로, 사이즈가 s_t인 사전 V_{LeTo}를 이용하여 기계번역 모델을 학습한다. 본 발명의 목적은 빈번하게 나타나고 길이가 L 이상인 서브워드 토큰들 $w \subset V_l$을 최종 사전 V_{LeTo}에 추가함으로써, 길이가 긴 서브워드 토큰들의 비율을 증가시키는 것이다. 본 명세서에서, r은 추가된 토큰의 비율로, 0 보다 크거나 같고 1 보다 작거나

같은 값을 갖는다.

[0028] 제안하는 기법에서는, 서브워드 기법을 이용한 종래의 사전 V로 기계번역 모델을 학습하지 않는다. 대신에, 기계번역 모델을 학습하기 위하여 큰 사이즈의 사전 V_l과 작은 사이즈의 사전 V_s가 결합된 사전 V_{LeTo}를 이용한다.

[0029] 보다 상세하게, 코퍼스(BPE 코퍼스) C_b가 BPE 사전을 구축하기 위하여 분할된다. 실시예에 따라, C_b는 임의의 비율로(예시적으로, 1/2) 분할되어 학습 코퍼스(training corpus) $\hat{C}$와 빈도수 조사 코퍼스(frequency examination corpus) $\bar{C}$로 나뉠 수 있다. 이때, BPE 코퍼스 C_b가 주어지지 않은 경우, 기계번역 모델을 위한 학습 데이터 C가 이용될 수 있다. 즉, C가 $\hat{C}$와 $\bar{C}$로 분할되거나, C를 $\hat{C}$와 $\bar{C}$로 이용할 수 있다. 그런 다음, $\hat{C}$를 이용하여 사이즈가 s_l인 V_l과 사이즈가 $s_t*(1-r)$인 V_s를 생성한다. 그리고, V_{LeTo} 내의 모든 토큰들을 더한다.

[0030] 다음으로, $\bar{C}$로부터의 빈도수 순서에 따라, V_l 내에서 s_t*r 만큼의 $\mathbf{w} \subset V_l$를 추출한다. 추출 과정에서, $\mathbf{w} \subset V_l$는 V_{LeTo}에 추가된다.

[0031] 마지막으로, 사이즈가 s_t인 사전 V_{LeTo}를 생성한다. 생성된 사전은 기계번역 모델의 학습에 사용될 수 있다.

[0032] 상술한 사전 구축 방법에 대한 알고리즘은 도 1에 도시되어 있다.

[0033] 도 2는 본 발명의 일 실시예에 따른 토큰화 방법을 설명하기 위한 흐름도이다. 토큰화 방법은 적어도 프로세서(processor) 및/또는 메모리(memory)를 포함하는 컴퓨팅 장치에 의해 수행될 수 있다. 따라서, 토큰화 방법을 구성하는 단계들 중 적어도 일부는 컴퓨팅 장치의 프로세서의 동작으로 이해될 수도 있다. 컴퓨팅 장치는 PC(Personal Computer), 서버(server) 등으로 구현될 수 있다.

[0034] 우선, 제1 사전이 생성될 수 있다(S110). 제1 사전은 길이가 긴 토큰들을 포함하는 사전을 의미할 수 있다. 예시적으로, 제1 사전에 포함되는 토큰들의 최소 길이는 L(L은 임의의 자연수로써 미리 정해진 값을 가진 수 있으며, 예시적인 값은 4 또는 5일 수 있음)일 수 있다. 제1 사전에 포함되는 토큰들은 BPE(Byte Per Encoding) 기법을 통해 생성될 수 있다.

[0035] 이때, 제1 코퍼스가 이용될 수 있다. 제1 코퍼스는 BPE 코퍼스를 의미할 수 있다. 또한, 제1 코퍼스는 임으의 비율(예컨대, 1/2)로, 즉 제1-1 코퍼스와 제1-2 코퍼스로 분할될 수 있다. 제1 사전에 포함되는 토큰들은 제1-1 코퍼스(또는 제1 코퍼스)로부터 추출될 수 있다. 실시예에 따라, 제1 코퍼스가 이용될 수 있다. 제2 코퍼스는 기계번역 모델의 학습이 이용되는 학습 데이터를 의미할 수 있다. 마찬가지로, 제2 코퍼스는 제2-1 코퍼스와 제2-2 코퍼스로 분할되거나 분할되지 않을 수 있다. 따라서, 제1 사전에 포함되는 토큰들은 제2-1 코퍼스나 제2 코퍼스로부터 추출될 수 있다.

[0036] 또한, 제1 사전의 사이즈는 s_l로 미리 정해질 수 있다.

[0037] 다음으로, 제2 사전이 생성될 수 있다(S120). 제2 사전은 길이가 작은 토큰들을 포함하는 사전을 의미할 수 있다. 예시적으로, 제2 사전에 포함되는 토큰들은 L보다 작을 수 있다. 그러나, 실시예에 따라, 제2 사전에 포함되는 토큰들의 길이에 대한 조건은 정해지지 않을 수도 있다. 길이에 대한 조건이 없어도 추출되는 대부분의 토큰들의 길이는 L 보다 작을 수 있기 때문이다. 제2 사전에 포함되는 토큰들은 BPE 기법을 통해 생성될 수 있다.

[0038] 이때, 제2 사전에 포함되는 토큰들은 제1 코퍼스 또는 제1-1 코퍼스로부터 추출될 수 있다. 실시예에 따라, 제2 사전에 포함되는 토큰들은 제2 코퍼스나 제2-1 코퍼스로부터 추출될 수도 있다.

[0039] 또한, 제2 사전의 사이즈는 $s_t*(1-r)$로 미리 정해질 수 있다. 여기서, s_t는 최종적으로 생성되는 최종 사전의 사이즈를, r은 최종 사전에 더해지는 토큰들의 비율을 의미한다.

[0040] 다음으로, 생성된 제1 사전과 제2 사전을 이용하여 최종 사전이 생성될 수 있다(S130). 구체적으로, 제1 사전에 포함된 토큰들을 빈도수의 내림차순으로 정렬하고, 정렬 순서대로 제2 코퍼스에 존재하는 토큰들만이 최종 사전에 추가될 수 있다. 여기서, 빈도수는 추출된 토큰들이 제1-2 코퍼스에 출현하는 빈도를 의미할 수 있다. 따라서, 제1-2 코퍼스에서의 출현 빈도를 산출하는 과정이 추가적으로 수행될 수 있다. 또한, 최종 사전은 제2 사전의 토큰들을 포함하도록 구성될 수 있다. 마찬가지로, 제2 사전의 토큰들 중 제2 코퍼스에 존재하는 토큰들만이 최종 사전에 추가될 수 있으나, 실시예에 따라, 제2 사전에 포함된 모든 토큰들이 최종 사전에 추가될 수도 있

다.

[0041] 결국, 길이가 긴 토큰의 비율이 증가된 최종 사전이 구축될 수 있다.

[0042] 물론, 제1 코퍼스가 존재하지 않는 경우, 빈도수 조사 등의 동작은 생략될 수 있다.

[0043] 추가적으로, 생성되 최종 사전을 이용하여 소정의 모델을 학습함으로써 기계번역 모델을 생성할 수 있다(S140). 이때, 학습 데이터는 제2 코퍼스이고, 토큰화 동작에서는 최종 사전이 이용될 수 있다.

[0044] 따라서, 토큰화 방법은 기계번역 모델 생성 방법으로 명명될 수도 있고, 컴퓨팅 장치는 기계번역 모델 생성 장치로 명명될 수 있다.

[0045] 이상에서 설명된 장치는 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 집합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성 요소는, 예를 들어, 프로세서, 콘트롤러, ALU(Arithmetic Logic Unit), 디지털 신호 프로세서(Digital Signal Processor), 마이크로컴퓨터, 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 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(Parallel Processor)와 같은, 다른 처리 구성(Processing Configuration)도 가능하다.

[0046] 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.

[0047] 실시예에 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 실시예를 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 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

[0048] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 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.

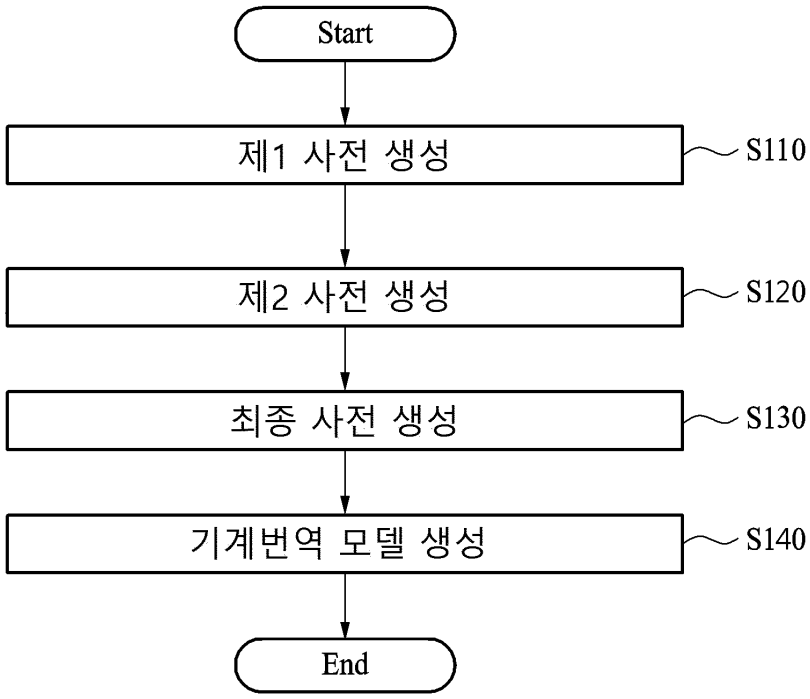
도면

도면1

Algorithm 1: Length-aware Subword Tokenization

Required: target vocab size s_t, large vocab size s_l, min length L, extraction ratio r $(0 \leq r \leq 1)$, translation corpus C
Required or not: bpe corpus C_b
// Initialization, v_e is a set of extracted words.
$i = 1$, $j = 1$, $m = 1$, $v_e = []$
if C_b is not **None then**
  $\hat{C}$, $\bar{C} \leftarrow$ RANDOM-HALF-SPLIT(C_b)
else
  $\hat{C}, \bar{C} \leftarrow C$
end if
// Train large vocabulary
$V_l \leftarrow$ TRAIN-BPE$(s_l, \hat{C})$
// Train small vocabulary
$V_s \leftarrow$ TRAIN-BPE$(s_t * (1 - r), \hat{C})$
// Extract words $\mathbf{w} \subset V_l$ *with length* $\geq L$ *and sort w in descending order by frequency from* $\bar{C}$
$V_e \leftarrow$ SORT-FREQ-DESC$(V_l, L, \bar{C})$
// $V_e = \{w_1, \ldots, w_N | L \leq |w|\}$, N *is # of extracted tokens*
if C_b is not **None then** ▷ *General domain scenario*
  for $w_k \in V_e$ **do**
    if LOOKUP-CORPUS(w_k, C) **then** ▷ *True if* w_k *exists in* C
      if $w_k \notin V_s$ **then**
        $v_e[j] \leftarrow w_k$
        $j \leftarrow j + 1$
      end if
      if $j = s_t * r$ **then break**
      end if
    end if
  end for
else ▷ *In-domain scenario*
  for $w_k \in V_e$ **do**
    if $w_k \notin V_s$ **then**
      $v_e[m] \leftarrow w_k$
      $m \leftarrow m + 1$
    end if
    if $m = s_t * r$ **then break**
    end if
  end for
end if
$V_{LeTo}[1, 2, \ldots, s_t * (1 - r)] \leftarrow V_s[1, 2, \ldots, s_t * (1 - r)]$
while $length(V_{LeTo}) \neq s_t$ **do**
  $V_{LeTo}[s_t * (1 - r) + i] \leftarrow v_e[i]$
  $i \leftarrow i + 1$
end while
return V_{LeTo}

도면2


Start
제1 사전 생성
S110
제2 사전 생성
S120
최종 사전 생성
S130
기계번역 모델 생성
S140
End