(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0098628
(43) 공개일자 2022년07월12일

(51) 국제특허분류(Int. Cl.)
G06F 40/211 (2020.01) *G06F 40/14* (2020.01)
G06F 40/268 (2020.01) *G06F 40/284* (2020.01)
G06N 20/00 (2019.01)
(52) CPC특허분류
G06F 40/211 (2020.01)
G06F 40/14 (2022.01)
(21) 출원번호 10-2021-0000530
(22) 출원일자 2021년01월04일
심사청구일자 2021년01월04일

(71) 출원인
충남대학교산학협력단
대전광역시 유성구 대학로 99 (궁동, 충남대학교)
(72) 발명자
정상근
서울특별시 동작구 장승배기로16길 134 쌍용예가 아파트 102동 801호
서혜인
대전광역시 유성구 상대남로 26 도안신도시9블록 트리풀시티아파트 919동 1003호
(74) 대리인
이은철, 김재문

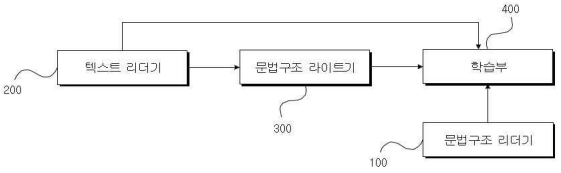
전체 청구항 수 : 총 12 항

(54) 발명의 명칭 **관계성 학습을 이용한 자연어 처리 시스템 및 방법**

(57) 요 약

본 기술은 관계성 학습을 이용한 자연어 처리 시스템 및 방법이 개시된다. 이러한 본 기술에 대한 구체적인 구현예는 수집된 다수의 말뭉치에 대한 각 문장 벡터를 토대로 구조 벡터를 생성하여 생성된 다수의 구조 벡터를 입력된 말뭉치와 매칭시켜 학습 모델을 구축하고, 입력된 말뭉치에 대해 구축된 학습 모델을 기반 학습 수행하여 입력된 말뭉치와 유사한 문장 구조를 가지는 다수의 유사 구조 벡터를 추정하고 추정된 유사 구조 벡터 및 문장 벡터의 거리 함수를 토대로 정답 문장구조 태그열을 생성함에 따라 입력된 문장의 문법구조를 반영하여 입력된 문장을 인코딩할 수 있고, 이에 자연스러운 대화를 진행할 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06F 40/268 (2020.01)
G06F 40/284 (2020.01)
G06N 20/00 (2021.08)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711108241
과제번호	2019R1F1A1060601
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	개인기초연구(과기정통부)(R&D)
연구과제명	설명 가능한 딥러닝 자연어 처리 기술 개발
기 여 율	1/1
과제수행기관명	충남대학교
연구기간	2020.03.01 ~ 2021.02.28

명 세 서

청구범위

청구항 1

수집된 말뭉치를 문법구조 모델링 및 전처리를 통해 문장 형태로 표현하고, 문장 형태의 의존 구문을 선학습 BERT를 통해 토큰으로 분리한 다음 각 토큰의 임베딩값으로 구조 벡터를 생성하여 학습 모델링하는 문법구조 리더기;

입력된 말뭉치에 대해 선학습 BERT(Bidirectional Encoder Representations from Transformers) 언어모델의 토크나이저를 통해 토큰 분리한 다음 각 토큰의 임베딩값으로 문장 벡터를 생성하는 텍스트 리더기;

텍스트 리더기로부터 생성된 문장 벡터에 대한 상기 학습 모델링을 통해 학습 수행하여 유사 구조 벡터를 생성하는 문법구조 라이터기;

상기 문법구조 라이터기의 다수의 유사 구조 벡터 중 각 유사 구조 벡터와 텍스트 리더기의 문장 벡터에 대한 거리함수를 기반으로 정답 구조 벡터를 추정하는 학습부를 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 2

제1항에 있어서, 상기 문법구조 리더기는

수집된 다수의 문장에 대해 다수의 의존 구문 분석(dependency parsing) 및 전처리를 통해 각 단어 간의 의존 관계를 문장 형태로 표현하고 의존 구문의 문장 형태를 BERT 토크나이저를 통해 분리한 다음 각 토큰에 대해 임베딩값을 연산하여 의존 구문 벡터로 출력하는 의존 구조 리더모듈;

수집된 다수의 문장에 대해 다수의 구 구조 구문 분석(phrase structure parsing)을 수행한 결과 생성된 구 구조 트리를 문장 형태로 표현한 다음 구 구조 문장 형태를 BERT 토크나이저를 통해 분리하고, 각 토큰에 대해 임베딩값을 연산하여 구 구조 벡터로 출력하는 구 구조 리더모듈; 및

다수의 문장에 대해 다수의 형태소 분석(part-of-speech tagging)을 수행하고, 각 형태소에 따른 품사 태그를 문장 형태로 표현한 다음 형태소 분석 문장 형태를 BERT 토크나이저를 통해 분리하고, 각 토큰에 대해 임베딩값을 연산하여 형태소 벡터로 출력하는 형태소 분석 리더모듈; 중 적어도 하나를 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 3

제2항에 있어서, 상기 문법구조 리더기의 전처리는

입력된 텍스트 리더기의 문장 벡터에 대해 각 의존 구문 분석을 통해 의존 구문 트리를 생성하고, 생성된 각 트리에서 각 단어 간의 관계인 아크 라벨 arc label을 문장 형태의 의존 구문으로 변환하며, 변환된 의존 구문의 문장에 대해 선학습 BERT 언어모델에 전달하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 4

제3항에 있어서, 상기 텍스트 리더기는

입력된 말뭉치를 순차적으로 분리하는 BERT 토크라이저; 및

각 토큰에 대해 임베딩값을 연산하고 연산된 임베딩값을 문장 벡터로 출력하는 문장벡터 출력부를 도출되는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 5

제4항에 있어서, 상기 학습부는,

입력된 상기 텍스트 리더기의 문장 벡터와 문장구조 라이트기의 유사 구조 벡터 간의 거리 비용을 도출하는 거리비용 도출모듈; 및

도출된 거리 비용이 가장 작은 문장 벡터 및 유사 구조 벡터에 대해 문법구조 태그열을 생성하여 생성된 문법구조 태그열을 정답 문법구조 태그열로 출력하는 정답 문법구조 도출모듈; 을 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 6

제5항에 있어서, 상기 학습부는,

실측된 정답 문법구조 태그열과 생성된 정답 문법구조 태그열을 기반으로 학습 성능을 제어하는 학습 제어기를 더 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 7

제6항에 있어서, 상기 학습 제어기는,

실측된 정답 문법구조 태그열과 상기 생성된 정답 문법구조 태그열에 대한 상관 엔트로피로 생성 비용을 연산하는 생성 비용 연산모듈; 및

연산된 생성 비용과 거리 비용의 합으로 손실 비용을 연산하는 손실 비용 연산모듈; 및

상기 손실 비용으로 학습 성능을 제어하는 학습 성능 제어모듈을 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 시스템.

청구항 8

수집된 말뭉치를 문법구조 모델링을 통해 문장 형태로 표현하고, 문장 형태의 의존 구문을 전처리 후 선학습 BERT를 통해 토큰으로 분리한 다음 각 토큰의 임베딩값으로 구조 벡터를 생성하여 학습 모델링하는 문법구조 리딩 단계;

입력된 말뭉치에 대해 선학습 BERT(Bidirectional Encoder Representations from Transformers) 언어모델의 토크나이저를 통해 토큰 분리한 다음 각 토큰의 임베딩값으로 문장 벡터를 생성하는 텍스트 리더 단계;

생성된 문장 벡터에 대한 상기 학습 모델링을 통해 학습 수행하여 유사 구조 벡터를 생성하는 문법구조 라이팅 단계; 및

다수의 유사 구조 벡터 중 각 유사 구조 벡터와 텍스트 리더기의 문장 벡터에 대한 거리함수를 기반으로 정답 구조 벡터를 추정하는 학습단계를 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 방법.

청구항 9

제8항에 있어서, 상기 문법구조 리더기의 전처리는,

입력된 텍스트 리더기의 문장 벡터에 대해 각 의존 구문 분석을 통해 의존 구문 트리를 생성하고,

생성된 각 트리에서 각 단어 간의 관계인 아크 라벨 arc label을 문장 형태의 의존 구문으로 변환하며,

변환된 의존 구문의 문장에 대해 선학습 BERT 언어모델로 전달하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 방법.

청구항 10

제8항에 있어서, 상기 학습 단계는,

입력된 상기 텍스트 리더기의 문장 벡터와 문장구조 라이트기의 유사 구조 벡터 간의 거리 비용을 도출하는 단계; 및

도출된 거리 비용이 가장 작은 문장 벡터 및 유사 구조 벡터에 대해 문법구조 태그열을 생성하여 생성된 문법구조 태그열을 정답 문법구조 태그열로 출력하는 단계를 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 방법.

청구항 11

제10항에 있어서, 상기 학습 단계는,

실측된 정답 문법구조 태그열과 상기 생성된 정답 문법구조 태그열에 대한 상관 엔트로피로 생성 비용을 연산하는 단계;

연산된 생성 비용과 거리 비용의 합으로 손실 비용을 연산하는 단계; 및

상기 손실 비용으로 학습 성능을 제어하는 단계를 더 포함하는 것을 특징으로 하는 관계성 학습을 이용한 자연어 처리 방법.

청구항 12

제8항 내지 제11항 중 어느 한 항의 관계성 학습을 이용한 자연어 처리 방법을 실행하기 위한 프로그램이 기록되고 컴퓨터에서 판단 가능한 기록매체.

발명의 설명

기 술 분 야

[0001] 본 발명은 관계성 학습을 이용한 자연어 처리 시스템 및 방법에 관한 것으로서, 더욱 상세하게는 입력된 문장의 문장 벡터와 입력된 문장에 대한 유사 문법 구조를 가지는 구조 벡터를 추정하고 추정된 구조 벡터 및 문장 벡터를 벡터 공간 상에서 서로 가까워짐에 따라 정답 문법구조 태그열을 생성함에 따라, 유사 문법 구조를 가지는 문장 검색을 실시간으로 인코딩할 수 있도록 한 기술에 관한 것이다.

배 경 기 술

[0002] 자연어 이해(Natural Language Understanding)란 인간이 발화하는 언어 현상을 기계적으로 분석해 컴퓨터가 이해할 수 있는 형태로 구조화하는 것을 말한다. 자연어 이해는 정보 추출, 문서 요약 등에 활용되는 핵심 기술 중 하나이다.

[0003] 문장과 의미들 사이의 관계성인 상호 연관성을 이용한 벡터공간 학습 방법을 소개하였고, 이는 자연어이해의 여러 분야에 다양하게 활용된다.

[0004] 그러나, 자연어 이해 시스템은 문장의 의미를 결정하는 중요한 요소인 단어 순서나 문장구조에 대해 문장과 문법적 구조 사이의 벡터 공간을 이용한 학습 방법은 전무한 상태이다.

[0005] 이에 본 출원인은 수집된 다수의 말뭉치에 대해 문장과 문장의 문법 구조 사이의 연관성에 대해 모델링하고, 문장과 각 문장의 문법 구조 사이의 상호 연관성을 벡터 공간상에서의 거리로 정의하여 입력된 말뭉치에 대한 학습을 수행하는 방안을 제안하고자 한다

선행기술문헌

비특허문헌

[0006] (비특허문헌 0001) Sangkeun Jung. Semantic vector learning for natural language understanding. Computer Speech & Language, 56:130 -145, 2019.

발명의 내용

해결하려는 과제

[0007] 따라서, 본 발명은 수집된 다수의 말뭉치에 대해 수집된 다수의 말뭉치에 대해 문장과 문장의 문법 구조 사이의 연관성에 대해 학습 모델링하고, 문장과 각 문장의 문법 구조 사이의 상호 연관성을 벡터 공간 상에서의 거리로 정의하여 입력된 말뭉치에 대해 학습을 수행한 다음 입력된 말뭉치와 유사한 문장 구조를 가지는 다수의 구조

벡터를 추정하고 추정된 구조 벡터 및 문장 벡터의 거리 함수를 토대로 정답 문장구조 태그열을 생성함에 따라 입력된 말뭉치의 문법적 구조를 반영하여 입력된 문장에 대한 인코딩을 수행할 수 있는 관계성 학습을 이용한 자연어 처리 시스템 및 방법을 제공하고자 한다.

[0008] 본 발명의 목적은 이상에서 언급한 목적으로 제한되지 않으며, 언급되지 않은 본 발명의 다른 목적 및 장점들은 하기의 설명에 의해서 이해될 수 있으며, 본 발명의 실시예에 의해 보다 분명하게 알게 될 것이다. 또한, 본 발명의 목적 및 장점들은 특허청구 범위에 나타낸 수단 및 그 조합에 의해 실현될 수 있음을 쉽게 알 수 있을 것이다.

과제의 해결 수단

[0009] 본 발명의 일 실시예에 따른 관계성 학습을 이용한 자연어 처리 시스템은,

[0010] 수집된 말뭉치를 문법구조 모델링 및 전처리를 통해 문장 형태로 표현하고, 문장 형태의 의존 구문을 선학습 BERT를 통해 토큰으로 분리한 다음 각 토큰의 임베딩값으로 구조 벡터를 생성하여 학습 모델링하는 문법구조 리더기;

[0011] 입력된 말뭉치에 대해 선학습 BERT(Bidirectional Encoder Representations from Transformers) 언어모델의 토크나이저를 통해 토큰 분리한 다음 각 토큰의 임베딩값으로 문장 벡터를 생성하는 텍스트 리더기;

[0012] 텍스트 리더기로부터 생성된 문장 벡터에 대한 상기 학습 모델링을 통해 학습 수행하여 유사 구조 벡터를 생성하는 문법구조 라이터기;

[0013] 상기 문법구조 라이터기의 다수의 유사 구조 벡터 중 각 유사 구조 벡터와 텍스트 리더기의 문장 벡터에 대한 거리함수를 기반으로 정답 구조 벡터를 추정하는 학습부를 포함하는 것을 일 특징으로 한다.

[0014] 바람직하게 상기 문법구조 리더기는

[0015] 수집된 다수의 문장에 대해 다수의 의존 구문 분석(dependency parsing) 및 전처리를 통해 각 단어 간의 의존 관계를 문장 형태로 표현하고 의존 구문의 문장 형태를 BERT 토크나이저를 통해 분리한 다음 각 토큰에 대해 임베딩값을 연산하여 의존 구문 벡터로 출력하는 의존 구조 리더모듈;

[0016] 수집된 다수의 문장에 대해 다수의 구 구조 구문 분석(phrase structure parsing)을 수행한 결과 생성된 구 구조 트리를 문장 형태로 표현한 다음 구 구조 문장 형태를 BERT 토크나이저를 통해 분리하고, 각 토큰에 대해 임베딩값을 연산하여 구 구조 벡터로 출력하는 구 구조 리더모듈; 및

[0017] 다수의 문장에 대해 다수의 형태소 분석(part-of-speech tagging)을 수행하고, 각 형태소에 따른 품사 태그를 문장 형태로 표현한 다음 형태소 분석 문장 형태를 BERT 토크나이저를 통해 분리하고, 각 토큰에 대해 임베딩값을 연산하여 형태소 벡터로 출력하는 형태소 분석 리더모듈; 중 적어도 하나를 포함할 수 있다.

[0018] 바람직하게 상기 문법구조 리더기의 전처리는

[0019] 입력된 텍스트 리더기의 문장 벡터에 대해 각 의존 구문 분석을 통해 의존 구문 트리를 생성하고, 생성된 각 트리에서 각 단어 간의 관계인 아크 라벨 arc label을 문장 형태의 의존 구문으로 변환하며, 변환된 의존 구문의 문장에 대해 선학습 BERT 언어모델을 통해 도출될 수 있다.

[0020] 바람직하게 상기 텍스트 리더기는

[0021] 입력된 말뭉치를 순차적으로 분리하는 BERT 토크라이저; 및

[0022] 각 토큰에 대해 임베딩값을 연산하고 연산된 임베딩값을 문장 벡터로 출력하는 문장벡터 출력부를 포함할 수 있다.

[0023] 바람직하게 상기 학습부는,

[0024] 입력된 상기 텍스트 리더기의 문장 벡터와 문장구조 라이트기의 유사 구조 벡터 간의 거리 비용을 도출하는 거리비용 도출모듈; 및

[0025] 도출된 거리 비용이 가장 작은 문장 벡터 및 유사 구조 벡터에 대해 문법구조 태그열을 생성하여 생성된 문법구조 태그열을 정답 문법구조 태그열로 출력하는 정답 문법구조 도출모듈; 을 포함하도록 구비될 수 있다.

[0026] 바람직하게 상기 학습부는,

[0027] 실측된 정답 문법구조 태그열과 생성된 정답 문법구조 태그열을 기반으로 학습 성능을 제어하는 학습 제어기를 더 포함할 수 있다.

[0028] 바람직하게 상기 학습 제어기는,

[0029] 실측된 정답 문법구조 태그열과 상기 생성된 정답 문법구조 태그열에 대한 상관 엔트로피로 생성 비용을 연산하는 생성 비용 연산모듈; 및

[0030] 연산된 생성 비용과 거리 비용의 합으로 손실 비용을 연산하는 손실 비용 연산모듈; 및

[0031] 상기 손실 비용으로 학습 성능을 제어하는 학습 성능 제어모듈을 더 포함할 수 있다.

[0032] 본 발명의 다른 실시 양태에 의거한 관계성 학습을 이용한 자연어 처리 방법은

[0033] 수집된 말뭉치를 문법구조 모델링을 통해 문장 형태로 표현하고, 문장 형태의 의존 구문을 전처리 후 선학습 BERT를 통해 토큰으로 분리한 다음 각 토큰의 임베딩값으로 구조 벡터를 생성하여 학습 모델링하는 문법구조 리딩 단계;

[0034] 입력된 말뭉치에 대해 선학습 BERT(Bidirectional Encoder Representations from Transformers) 언어모델의 토크나이저를 통해 토큰 분리한 다음 각 토큰의 임베딩값으로 문장 벡터를 생성하는 텍스트 리더 단계;

[0035] 생성된 문장 벡터에 대한 상기 학습 모델링을 통해 학습 수행하여 유사 구조 벡터를 생성하는 문법구조 라이팅 단계; 및

[0036] 다수의 유사 구조 벡터 중 각 유사 구조 벡터와 텍스트 리더기의 문장 벡터에 대한 거리함수를 기반으로 정답 구조 벡터를 추정하는 학습단계를 포함하는 것을 일 특징으로 한다.

[0037] 바람직하게 상기 문법구조 리더기의 전처리는,

[0038] 입력된 텍스트 리더기의 문장 벡터에 대해 각 의존 구문 분석을 통해 의존 구문 트리를 생성하고, 생성된 각 트리에서 각 단어 간의 관계인 아크 라벨 arc label을 문장 형태의 의존 구문으로 변환하며, 변환된 의존 구문의 문장에 대해 선학습 BERT 언어모델을 통해 도출될 수 있다.

[0039] 바람직하게 상기 학습 단계는,

[0040] 입력된 상기 텍스트 리더기의 문장 벡터와 문장구조 라이트기의 유사 구조 벡터 간의 거리 비용을 도출하는 단계; 및 도출된 거리 비용이 가장 작은 문장 벡터 및 유사 구조 벡터에 대해 문법구조 태그열을 생성하여 생성된 문법구조 태그열을 정답 문법구조 태그열로 출력하는 단계를 포함할 수 있다.

[0041] 바람직하게 상기 학습 단계는,

[0042] 실측된 정답 문법구조 태그열과 상기 생성된 정답 문법구조 태그열에 대한 상관 엔트로피로 생성 비용을 연산하는 단계; 연산된 생성 비용과 거리 비용의 합으로 손실 비용을 연산하는 단계; 및 상기 손실 비용으로 학습 성능을 제어하는 단계를 더 포함할 수 있다.

발명의 효과

[0043] 일 실시 예에 따르면, 수집된 다수의 말뭉치에 대한 각 문장벡터와 구조 벡터 사이의 연관성 학습 방법을 통해 유사한 문장 구조를 갖는 문장들이 벡터 공간상의 비슷한 공간 안에 배치되도록 하는 벡터 형태의 문장 구조를 반영한 문장표현 방법에 대한 학습 모델을 구축하고, 입력된 말뭉치에 대해 구축된 학습 모델을 기반 학습 수행하여 입력된 말뭉치와 유사한 문장 구조를 가지는 다수의 유사 구조 벡터를 추정하고 추정된 유사 구조 벡터 및 문장 벡터의 거리 함수를 토대로 정답 문장구조 태그열을 생성함에 따라 입력된 문장의 문법구조를 반영하여 입력된 문장을 인코딩할 수 있고, 이에 자연스러운 대화를 진행할 수 있는 효과를 가진다.

도면의 간단한 설명

[0044] 본 명세서에서 첨부되는 다음의 도면들은 본 발명의 바람직한 실시 예를 예시하는 것이며, 후술하는 발명의 상세한 설명과 함께 본 발명의 기술사상을 더욱 이해시키는 역할을 하는 것이므로, 본 발명은 그러한 도면에 기재된 사항에만 한정되어 해석되어서는 아니된다.

도 1은 일 실시예의 관계성 학습을 이용한 자연어 처리 시스템의 구성도이다.

도 2는 일 실시예의 시스템의 문법구조 리더기의 세부 구성도이다.

도 3은 일 실시예의 시스템의 의존 구문 트리를 보인 예시도이다.

도 4는 일 실시예의 시스템의 구 구조 분석 결과를 보인 예시도이다.

도 5는 일 실시예의 시스템의 POS 분석 결과를 보인 예시도이다.

도 6은 일 실시예의 시스템의 텍스트 리더기의 세부 구성도이다.

도 7은 일 실시예의 시스템의 학습부의 세부 구성도이다.

발명을 실시하기 위한 구체적인 내용

[0045] 실시예들에 대한 특정한 구조적 또는 기능적 설명들은 단지 예시를 위한 목적으로 개시된 것으로서, 다양한 형태로 변경되어 실시될 수 있다. 따라서, 실시예들은 특정한 개시형태로 한정되는 것이 아니며, 본 명세서의 범위는 기술적 사상에 포함되는 변경, 균등물, 또는 대체물을 포함한다.

[0046] 제1 또는 제2 등의 용어를 다양한 구성요소들을 설명하는데 사용될 수 있지만, 이런 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 해석되어야 한다. 예를 들어, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.

[0047] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다.

[0048] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 설명된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함으로 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0049] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 해당 기술분야에서 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.

[0050] 이하, 첨부된 도면을 참조하여 본 발명의 바람직한 실시 예를 설명함으로써, 본 발명을 상세히 설명한다.

[0051] 도 1은 일 실시예의 관계성 학습을 이용한 자연어 처리 시스템의 구성도이고, 도 2는 도 1의 문법구조 리더기의 세부 구성도이며, 도 3은 도 2의 의존 구문 트리를 보인 예시도이고, 도 4는 도 2의 구 구조 분석 결과를 보인 예시도이며, 도 5는 도 2의 POS 분석 결과를 보인 예시도이고, 도 6은 도 1의 텍스트 리더기의 세부 구성도이며, 도 7은 도 1의 학습부의 세부 구성도이다.

[0052] 도 1 내지 도 7을 참조하면, 일 실시 예에 따른 관계성 학습을 이용한 자연어 처리 시스템은 수집된 다수의 말뭉치에 대한 각 문장벡터와 구조 벡터 사이의 연관성 학습 방법을 통해 유사한 문장 구조를 갖는 문장들이 벡터 공간상의 비슷한 공간 안에 배치되도록 하는 벡터 형태의 문장 구조를 반영한 문장표현 방법에 대한 학습 모델을 구축하고, 입력된 말뭉치에 대한 문장 벡터에 대해 구축된 학습 모델을 기반 학습 수행하여 유사한 문장 구조를 가지는 다수의 유사 구조 벡터를 추정하고 추정된 다수의 유사 구조 벡터 각각과 문장 벡터의 거리 함수를 토대로 정답 문장구조 태그열을 생성하는 구성을 갖추며, 이에 시스템은 문법구조 리더기(100), 텍스트 리더기(200), 문법구조 라이트기(300), 및 학습부(400)을 포함할 수 있다.

[0053] 여기서, 문법구조 리더기(100)는 수집된 다수의 문장 벡터 각각에 대한 구조 벡터를 생성하여 학습 모델을 구축하도록 구비되고, 이에 문법구조 리더기(100)는 도 2에 도시된 바와 같이, 의존 구조(dependency structure) 리더모듈(110), 구 구조(phrases structure) 리더모듈(120), 및 형태소 구조(POS: Part-Of-Speech) 리더모듈(130) 중 적어도 하나를 포함할 수 있다.

[0054] 의존 구조 리더모듈 (110)는 수집된 다수의 문장에 대해 다수의 의존 구문 분석(dependency parsing) 및 전처리를 통해 각 단어 간의 의존 관계를 문장 형태로 표현하고 의존 구문의 문장 형태를 BERT 토크나이저를 통해 분리한 다음 각 토큰에 대해 임베딩값을 연산하여 의존 구문 벡터로 출력하고 출력된 의존 구문 벡터로 학습 모델

링한다.

[0055] 여기서, 전처리는 수집된 다수의 문장에 대해 각 의존 구문 분석을 통해 의존 구문 트리가 생성되고, 생성된 각 트리에서 각 단어 간의 관계인 아크 라벨 arc label을 문장 형태의 의존 구문으로 변환하며, 변환된 의존 구문의 문장에 대해 선학습 BERT 언어모델을 통해 의존 구조 벡터가 출력된다.

[0056] 일 례로, 수집된 말뭉치 “on april first I need a flight from phoenix to san diego”에 대한 의존 구문은 도 3을 참조하면, “prep-prbj-advmod-nsubj-ROOT-det_dobj_aci-prep-pobj-prep-compoung-pobj” 이다.

[0057] 여기서, 수집된 말뭉치에 한국어인 경우 AIOpen에서 공개한 에트리 의존 구문 분석 API를 사용하고 영어 문장의 경우 스파이시 오픈소스 라이브러리를 사용하며, 이러한 의존 구문 분석하는 일련의 과정은 본 명세서 상에서 구체적으로 기재하지 아니하였으나 당업자 수준에서 이해되어야 할 것이다.

[0058] 그리고, 구 구조 리더모듈 (120)는 수집된 다수의 문장에 대해 다수의 구 구조 구문 분석(phrase structure parsing)을 수행한 결과 생성된 구 구조 트리를 문장 형태로 표현한 다음 구 구조 문장 형태를 BERT 토크나이저를 통해 분리하고, 각 토큰에 대해 임베딩값을 연산하여 구 구조 벡터로 출력하여 학습 모델링을 수행한다.

[0059] 일 례로, 수집된 문장 “on april first I need a flight from phoenix to san diego” 에 대한 구 구조는 도 4에 도시된 바와 같이, “: “(S (PP (IN) (NP (NP (NNP)) (ADVP (RB)))) (NP (PRP)) (VP (VBP) (NP (NP (DT) (NN)) (VP (VBG) (PP (IN) (NP (NNP))) (PP (IN) (NP (NNP) (NNP))))))))” 이다.

[0060] 한편, 형태소 구조(POS tagging) 리더모듈(130)은 다수의 문장에 대해 다수의 형태소 분석(part-of-speech tagging)을 수행하고, 각 형태소에 따른 품사 태그를 문장 형태로 표현한 다음 형태소 분석 문장 형태를 BERT 토크나이저를 통해 분리하고, 각 토큰에 대해 임베딩값을 연산하여 형태소 벡터로 출력하고 이를 토대로 학습 모델링을 수행한다.

[0061] 일 례로 수집된 문장 “on april first I need a flight from phoenix to san diego”에 대한 구 구조는 도 5를 참조하면 “ADP-PROPN-ADV-PRON-VERB-DET-NOUN-VERB-ADP-PROPN-ADP-PROPN-PROPN” 이다.

[0062] 한편, 입력된 말뭉치는 텍스트 리더기(200)로 전달되고, 텍스트 리더기(200)는 선학습 BERT(Bidirectional Encoder Representations from Transformers) 언어모델 및 전처리를 토대로 문장 벡터를 생성하도록 구비되고 이에 텍스트 리더기(200)는 도 6을 참조하여 BERT 토크라이저(210) 및 문장벡터 출력부(220)을 포함한다.

[0063] 여기서 전처리는 즉, 문법구조 라이터기(300)는 입력된 말뭉치에 대해 각 의존 구문 분석을 통해 의존 구문 트리를 생성하고, 생성된 각 트리에서 각 단어 간의 관계인 아크 라벨 arc label을 문장 형태의 의존 구문으로 변환하며, 변환된 의존 구문의 문장에 대해 선학습 BERT 언어모델로 전달된다.

[0064] 이에 선학습 BERT 언어모델의 BERT 토크라이저(210)는 입력된 문장을 순차적으로 분리한 다음 분리된 토큰은 문장벡터 출력부(220)로 전달된다. 이에 문장벡터 출력부(220)는 각 토큰에 대한 임베딩값을 연산하고 연산된 임

v_t

베딩값을 문장벡터 로 출력한다.

v_t

[0065] 그리고, 출력된 문장벡터 는 문법구조 라이터기(300)로 전달된다. 문법구조 라이터기(300)는 텍스트 리더기

v_t v_t

(100)의 문장벡터 에 대해 구축된 학습 모델 기반으로 학습 수행하여 입력된 문장벡터 에 대한 다수의 유사 구조 벡터 Vy를 출력한다.

[0066] 일 례로, “오늘 미세먼지 정보를 알려줄래?”의 입력 문장에 대해, 의존 구분 분석을 통해 위존 구문 트리가 출력되고, 의존 구문 트리구조는 다음과 같다.

[0067] 

[0068] 그리고, 생성된 의존 구문 트리에서 각 단어 간의 관계인 아크 레벨 arc level을 의존 구문의 문장 형태는 “NP_AJT-NP-NP_OBJ-VP” 이다.

[0069] 그리고 문장 형태의 의존 구문에 대해 문법구조 문장 형태로 가공한 다음 선학습 BERT 언어모델을 수행하여 최 v_y v_y 종 다수의 유사 구조벡터 를 도출하고 도출된 각 다수의 유사 구조벡터 는 학습부(400)로 전달된다.

[0070] 힉습부(400)는 문법구조 라이터기의 다수의 유사 구조 벡터 중 각 구조 벡터와 텍스트 리더기(100)의 문장 벡터 v_t 에 대한 거리함수를 기반으로 정답 구조벡터를 추정하는 구성을 갖추며, 이에 학습부(400)는 도 7을 참조하면, 거리비용 도출모듈(410) 및 정답 문법구조 도출모듈(420)을 포함할 수 있다.

[0071] 거리비용 도출모듈(410)은 입력된 텍스트 리더기(200)의 문장 벡터 v_t 와 문장구조 라이트기(300)의 각 유사 구 v_y L_{dist} 조 벡터 에 대해, 기 정해진 코사인 거리 함수로 거리 비용 를 도출한다.

[0072] 즉, 벡터 공간 상에서 거리가 가장 근접한 문장 벡터 v_t 와 구조 벡터 v_y 에 대한 문법구조 태그열이 생성된다.

$$L_{dist} = dist(v_t, v_y)$$

즉, 이고 코사인 함수로 도출된다. 이러한 코사인 함수를 이용하여 벡터 공간 상에서 거리를 도출하는 일련의 과정은 선행문헌에 개시된 과정과 유사하므로 이에 대한 구체적인 설명은 생략한다.

[0073] 그리고, 정답 문법구조 도출모듈(420)은 도출된 거리 비용이 가장 작은 문장 벡터 및 구조 벡터에 대해 문법구조 태그열을 생성하여 생성된 문법구조 태그열로 정답 문법구조 태그열을 출력한다. 이에 정답 문법구조 태그열로 입력된 질의 문장에 대한 자연어 이해를 수행한 다음 이에 대한 응답 문장이 생성된다.

[0074] 한편, 관계성 학습을 이용한 자연어 처리 시스템은 실측된 정답 문법구조 태그열과 생성된 문법구조 태그열을 기반으로 학습 성능을 제어하는 학습 제어기를 더 포함하되, 학습 제어기는 도 7에 도시된 바와 같이 생성 비용 연산모듈(430), 소비 비용 연산모듈(440), 및 학습 성능 제어모듈(450)를 포함한다.

[0075] 즉, 생성 비용 연산모듈(430)은 실측된 정답 문법구조 태그열과 상기 생성된 문법구조 태그열에 대한 상관 엔트로피로 생성 비용을 연산하여 생성 비용을 도출하고, 도출된 생성 비용은 손실 비용 연산모듈(440)로 전달된다.

[0076] 그리고 손실 비용 연산모듈(440)은 연산된 생성 비용과 거리 비용의 합으로 손실 비용을 연산하고 연산된 손실 비용은 학습 성능 제어모듈(450)로 전달된다.

[0077] 학습 성능 제어모듈(450)은 최소 손실 비용을 가지도록 학습 변수를 조절하여 학습 성능 최대화를 수행한다.

[0078] 여기서, 학습 변수라 함은 일반적인 RNN(Recurrent Neural Network) 언어모델의 학습 파라미터들로서, 순환 신경망(RNN: Recurrent Neural Network) 모델을 기반으로 학습된 인코딩 장치는, 입력 계층(input layer)과 은닉 계층(hidden layer)을 포함하는 인코더(encoder) 및 은닉 계층과 출력 계층(output layer)을 포함하는 디코더(decoder)로 구성된다.

[0079] 인코더와 디코더에 학습된 모델은 학습 과정에서 계속 데이터를 생성해 내고, 학습이 끝난 이후에는 인코더의 출력 형식의 수학적 함수 분포(distribution)를 디코더에 넣어 주면 일정한 출력물을 생성할 수 있다. 즉, RNN은 신경망 학습 과정에서 시계열적으로 이전 또는/및 이후 데이터를 활용하여 학습을 수행한다. RNN은 시간 스텝 t에서의 출력값이 이전 시간 스텝 및/또는 이후의 시간 스텝에서 들어오는 입력값에도 영향을 받을 수 있다는 아이디어에 기반한다. 예를 들어, 한글 문제에서 빈칸에 가장 일치하는 단어를 채우기 위해서는 빈칸보다 앞쪽 문장들을 기반으로 빈칸 이후의 단어들의 문맥을 파악할 수 있다. 이러한 순환 신경망 구조는 양방향

(birectional) RNN의 경우 두 개의 RNN을 포함하고, 출력값은 두 개의 RNN의 은닉 계층에 의존하여 결정된다.

[0080] 이하 한국어 기반의 weather 및 nevi 말뭉치와 영어 기반의 Atis 및 snips 말뭉치 각각에 대한 학습 결과에 의거 도출된 정답 문장구조 태그열에 대해, 기 정해진 평가 기준(BLEU 및 ROUGE)으로 평가하는 시뮬레이션에 대해 설명한다.

[0081] 여기서 평가 기준 BLEU는 레퍼런스에 대한 점수를 내는 방식으로 n_gram을 통해 순서쌍들이 얼마나 겹치는 지를 측정하는 방식이고, ROUGE는 단어 수에 따라 두 문장 사이 가장 겹치는 문장 길이를 이용하여 점수를 도출하는 방식이다.

[0082] 일 례로 “오늘 미세먼지 정보 알려줄래”의 입력된 말뭉치에 대한 문자 벡터와 구문 벡터에 대해, 학습 결과로 도출된 유사 의존 구조 벡터는 하기 표 1에 도시된 바와 같다.

[0083] [표 1]

Query (text)	Top	Sentence / dependency parsing tags
오늘 미세먼지 정보 알려줄래? NP_AJT-NP-NP_OBJ-VP	1	오늘 미세먼지 수치 알려줘. NP_AJT-NP-NP_OBJ-VP
	2	내일 독도 날씨 알려줘. NP_AJT-NP-NP_OBJ-VP
	3	오후에 태백시 강수량 알려줄래? NP_AJT-NP-NP_OBJ-VP
	4	내일모레 강원도 날씨 알려줄래? NP_AJT-NP-NP_OBJ-VP
	5	오늘 수성구 최저기온 알려줄래? NP_AJT-NP-NP_OBJ-VP

(a) 유사 문장 예측 결과

Query (Syntax)	Top	Sentence / dependency parsing tags
NP_AJT-NP-NP_OBJ-VP 오늘 미세먼지 정보 알려줄래?	1	내일모레 경북 최고온도 알려주라. NP_AJT-NP-NP_OBJ-VP
	2	오늘 광진구 최저기온 말해줘. NP_AJT-NP-NP_OBJ-VP
	3	저녁 대구 눈 오려나? NP_AJT-NP-NP_OBJ-VP
	4	7시포항 기상 정보 말해줄래? NP_AJT-NP-NP_OBJ-VP
	5	8시 창녕군 체감온도 말해줘. NP_AJT-NP-NP_OBJ-VP

(b) 유사 구조 예측 결과

[0084]

[0085] 표 1을 참조하면, “오늘 미세먼지 정보 알려줄래?”에 대한 말뭉치 및 “NP_AJT_NP_NP_OBJ_VP”의 문장 벡터에 대해 “오늘 미세먼지 수치 알려줘”, “내일 독도 날씨 알려줘”, 등 의존 구문 및 “NP_AJT_NP_NP_OBJ_VP”의 의존 구문 벡터가 학습부(400)로 전달되면, 학습 결과에 따라 “내일 모래 경북 최고온도 알려주라”의 유사 의존 구문과 “NP_AJT_NP_NP_OBJ_VP”의 의존 구문 벡터가 도출된다.

[0086] 여기서, 문장 벡터는 입력된 한국어 기반 말뭉치에 대해, BERT-base, Multilingual Cased, 영어 기반 말뭉치에 대해 BERT _base, Cased 선학습 모델로 도출된다.

[0087] 또한, 구조 벡터는 한국어 기반 말뭉치에 대해, BERT-base, Multilingual Cased, 영어 기반 말뭉치에 대해 BERT _base, Cased 선학습 모델로 도출된다.

[0088] 이러한 학습 결과를 기 정해진 평가 기준에 의거한 평가 결과는 하기 표 2에 도시된 바와 같다.

[0089] [표 2]

		Weather		Navi		Atis		Snips	
	n-gram	random (baseline)	our	random (baseline)	our	random (baseline)	our	random (baseline)	our
BLEU	n=1	0.705	0.925	0.598	0.946	0.585	0.738	0.467	0.799
	n=2	0.531	0.870	0.407	0.907	0.461	0.631	0.326	0.660
	n=3	0.397	0.782	0.270	0.798	0.372	0.537	0.238	0.555
	n=4	0.299	0.624	0.164	0.475	0.294	0.455	0.178	0.477
ROUGE	-	0.766	0.963	0.843	0.970	0.741	0.816	0.782	0.842

표 2. 말뭉치 별 유사도 기반 구문파악 평가, 구조유사도를 검증하고자 random(baseline)에서는 테스트 문장과 임의로 추출한 문장과의 유사도를 비교하였다.

[0090]

[0091] 표 2를 참조하면, weather, nevi, Atis 및 snips 말뭉치에 대해, 기준 보다 높은 성능을 보이고 있고, BLEU의 경우 1_gram에서 가장 성능이 향상됨을 확인할 수 있다.

[0092] 본 발명의 다른 실시예에 의한 관계성 학습을 이용한 자연어 처리 방법은,

[0093] 수집된 말뭉치를 문법구조 모델링을 통해 문장 형태로 표현하고, 문장 형태의 의존 구문을 전처리 후 선학습 BERT를 통해 토큰으로 분리한 다음 각 토큰의 임베딩값으로 구조 벡터를 생성하여 학습 모델링하는 문법구조 리딩 단계; 입력된 말뭉치에 대해 선학습 BERT(Bidirectional Encoder Representations from Transformers) 언어 모델의 토크나이저를 통해 토큰 분리한 다음 각 토큰의 임베딩값으로 문장 벡터를 생성하는 텍스트 리더 단계; 생성된 문장 벡터에 대한 상기 학습 모델링을 통해 학습 수행하여 유사 구조 벡터를 생성하는 문법구조 라이팅 단계; 및 다수의 유사 구조 벡터 중 각 유사 구조 벡터와 텍스트 리더기의 문장 벡터에 대한 거리함수를 기반으로 정답 구조 벡터를 추정하는 학습단계를 포함하고, 상기의 관계성 학습을 이용한 자연어 처리 하는 각 단계는, 전술한 문법구조 리더기(100), 텍스트 리더기(200), 문법구조 라이터기(300), 및 학습부(400)에서 수행되는 기능으로 자세한 원용은 생략한다.

[0094] 이에 일 실시예는, 수집된 수집된 다수의 말뭉치에 대한 각 문장벡터와 구조 벡터 사이의 연관성 학습 방법을 통해 유사한 문장 구조를 갖는 문장들이 벡터 공간상의 비슷한 공간 안에 배치되도록 하는 벡터 형태의 문장 구조를 반영한 문장표현 방법에 대한 학습 모델을 구축하고 입력된 말뭉치에 대해 구축된 학습 모델을 기반 학습 수행하여 입력된 말뭉치와 유사한 문장 구조를 가지는 다수의 유사 구조 벡터를 추정하고 추정된 유사 구조 벡터 및 문장 벡터의 거리 함수를 토대로 정답 문장구조 태그열을 생성함에 따라 입력된 문장의 문법구조를 반영하여 입력된 문장을 인코딩할 수 있고, 이에 자연스러운 대화를 진행할 수 있다.

[0095] 이상에서 설명된 실시예들은 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치, 방법 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate 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(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 기술분야에서 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.

[0096] 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, 구성요소(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.

[0097] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 컴퓨터 판독 가능 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 광매체(magneto-optical media), 및 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.

[0098] 이상과 같이 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면

상기를 기초로 다양한 기술적 수정 및 변형을 적용할 수 있다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

부호의 설명

[0099] 100 : 문법구조 리더기

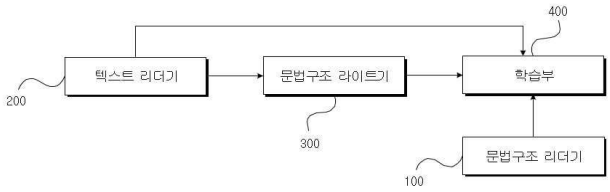
200 : 텍스트 리더기

300 : 문법구조 라이트기

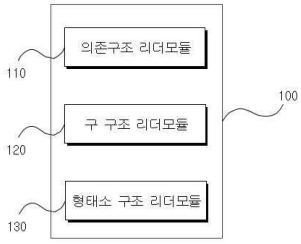
4300 : 학습부

도면

도면1

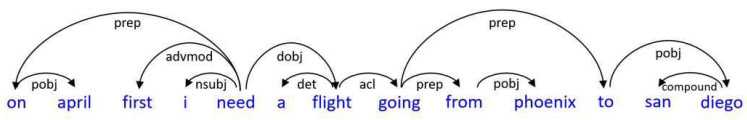


도면2



도면3

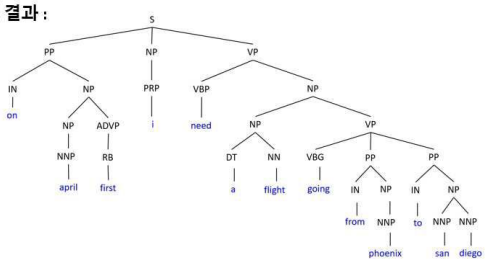
- **문장 : “on april first I need a flight going from phoenix to san diego”**
- **의존구조분석 결과 :**



- **의존구문 문장형태 : “prep-pobj-advmod-nsubj-ROOT-det-dobj-acl-prep-pobj-prep-compound-pobj”**

도면4

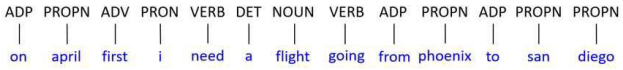
- **문장 : "on april first I need a flight going from phoenix to san diego"**
- **구구조 구문분석 결과 :**



- **구구조 문장형태 : "(S (PP (IN) (NP (NP (NNP)) (ADVP (RB)))) (NP (PRP)) (VP (VBP) (NP (NP (DT) (NN)) (VP (VBG) (PP (IN) (NP (NNP))) (PP (IN) (NP (NNP) (NNP)))))))"**

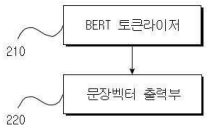
도면5

- **문장 : "on april first I need a flight going from phoenix to san diego"**
- **형태소 분석 결과 :**



- **형태소 분석 문장형태 : "ADP-PROPN-ADV-PRON-VERB-DET-NOUN-VERB-ADP-PROPN-ADP-PROPN-PROPN"**

도면6



도면7

