(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0066972
(43) 공개일자 2020년06월11일

(51) 국제특허분류(Int. Cl.)
G06F 40/20 (2020.01) *G06F 40/10* (2020.01)
(52) CPC특허분류
G06F 40/205 (2020.01)
G06F 40/10 (2020.01)
(21) 출원번호 10-2018-0153759
(22) 출원일자 2018년12월03일
심사청구일자 없음

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
임희석
경기도 수원시 영통구 영통로 460, 304동 1704호 (영통동, 대우.동신아파트)
조재춘
서울특별시 성북구 오패산로16가길 40, 102동 1306호(하월곡동, 꿈의숲푸르지오아파트)
(뒷면에 계속)
(74) 대리인
송인호, 윤형근, 최영중, 최관락

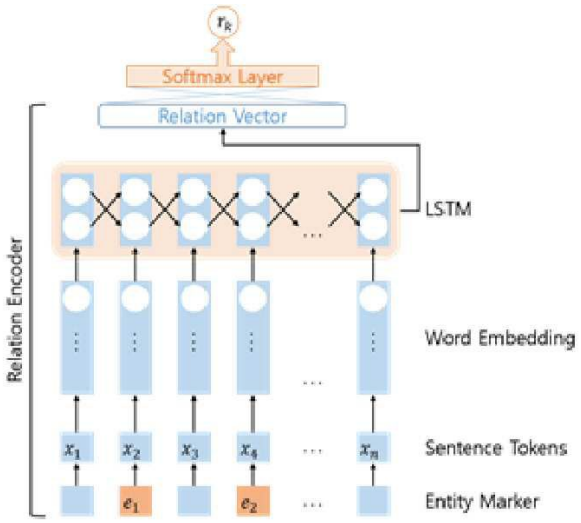
전체 청구항 수 : 총 1 항

(54) 발명의 명칭 **한글 문서 내 전역적 관계 추출기**

(57) 요 약

본 발명은 한글 문서 내 전역적 관계 추출기를 개시한다.

대 표 도 - 도1



(52) CPC특허분류
G06F 40/30 (2020.01)
(72) 발명자
김규경
서울특별시 양천구 목동남로2길 60-7, 101동 1302호(신정동, 양천중앙하이츠아파트)
김경민
서울특별시 종로구 평창3길 10, 1동 101호(평창동, 삼성그린빌리지)

명 세 서

청구범위

청구항 1

한글 문서 내 전역적 관계 추출기.

발명의 설명

기 술 분 야

[0001] 본 발명은 자연어처리에 관한 것이며, 보다 상세하게는 문서 요약에 도움을 주는 모델 장치에 관한 것이다.

배 경 기 술

[0002] 관계 추출(Relation Extraction): 관계 추출은 구조화되지 않은 데이터로부터 구조화된 정보를 추출하는데 핵심적인 역할을 수행한다. 이는 단순히 텍스트 데이터뿐만 아니라 개체와 개체간의 상호관계로 의미를 전달하는 모든 종류의 매체에 사용되어 질 수 있다. 이러한 관계 추출은 크게 2가지 종류로 나눌 수 있는데 각각 전역 수준의 관계 추출(Global Level Relation Extraction)과 언급 수준의 관계 추출(Mention Level Relation Extraction)이 존재한다. 현재 대부분의 높은 성능을 자랑하는 관계 추출 모델은 후자에 속하며, 문장 내의 관계 추출 성능은 우수하다. 그러나 이는 많은 양의 정보를 요약하고 그 주제를 파악하는데에는 한계가 있다. 이 논문에서는 전역 수준의 관계 추출을 한 번에 대량의 문서를 전부 분석하여 관계를 추출하기보다는 각 문장을 분석한 뒤, 문장 내에서 추출된 단편적인 관계들을 전체적으로 분석함으로써 전역 수준의 관계추출을 시도한다. 이러한 시도방법을 취함으로써 전역 수준의 관계 추출을 하되, 언급 수준의 관계 추출을 병행함으로써 정보의 누락을 최대한 방지한다.

[0003] 메모리 증강 신경망(Memory Augmented Neural Network): 본 논문에서는 전역 수준 관계 추출과 언급 수준 관계 추출을 병행함으로써 많은 양의 개체 관계들을 처리하게 된다. 그러나 이러한 관계들 중 주요 개체가 포함된 개체들의 관계는 전역 수준 관계 추출에 커다란 영향을 끼친다. 그렇기 때문에 이들이 가지고 있는 관계들은 따로 샘플로 외부메모리에 저장하여 전역 수준 관계 추출 때 해당 관계 분류를 할 때 사용된다. 그리고 이렇게 추출된 주요 개체 간의 관계들은 다시 한 번 언급 수준 관계들을 재조정하는데 사용된다. 이는 메타 학습(Meta Learning)에서 사용되는 One-shot 학습 방법을 적용한 것이다. 모델을 참조한 에서는 첫 번째로 나온 분류 예측 결과를 외부 메모리에 샘플로 저장하고, 이를 차후 비슷한 유형의 샘플을 예측할 때 가져와서 사용하고, 이 결과에 따라 첫 번째 분류 예측 프로세스에 역전파 신호(Back Propagated Signal)를 보내어 세부 사항을 다시 한 번 재조정한다.

발명의 내용

해결하려는 과제

[0004] 관계 추출에는 크게 2가지가 존재한다. 각각 전역 수준의 관계 추출(Global Level Relation Extraction)과 언급 수준의 관계 추출(Mention Level Relation Extraction)이다. 현재 대부분의 높은 성능을 자랑하는 관계 추출 모델은 후자에 속하며, 문장 내의 관계 추출 성능은 우수하나, 장문의 텍스트를 처리할 경우 그 성능과 정확도가 크게 떨어지는 경향이 보인다. 반면 전역 관계 추출에서는 한번에 대량의 텍스트를 처리하면서 정보 누락 현상이 자주 일어난다. 이는 실질적으로 관계추출기술이 요약 및 분석이 많이 요구되는 문서 내 관계 추출에서는 활용도가 떨어지는 결과를 초래한다. 본 발명은 전역관계추출을 시도하되 문장 내에서 추출된 단편적인 관계들을 전체적으로 분석함으로써 전역 수준의 관계추출을 시도한다. 이러한 시도방법을 취함으로써 전역 수준의 관계 추출을 하되, 언급 수준의 관계 추출을 병행함으로써 정보의 누락을 최대한 방지하는 형식으로 이를 해결하고자 한다.

과제의 해결 수단

[0005] 상기한 종래기술의 문제점을 해결하기 위해 본 발명의 바람직한 일 실시예에 따르면, 한글 문서 내 전역적 관계

추출기가 제공된다.

발명의 효과

[0006] 문서 단위의 관계 추출을 시도할 때 더욱 정확한 관계추출이 가능해지게 한다. 또한 일반적으로 자연어로 이루어진 장문의 텍스트에서 단 하나의 문장 내에서 성립되는 관계보다는 여러 문장에 걸쳐 표현되어지는 개체들 간의 관계가 더 해당 텍스트 내의 주제를 더 잘 반영하나, 외부 메모리를 이용한 본 발명의 모델은 그러한 문제로부터 좀 더 자유롭다. 그렇기 때문에 이러한 여러 문장에 걸쳐서 표현되어지는 관계를 추출하는 것이 해당 문서를 더욱 정확하게 요약하며, 단순히 단편적인 관계들을 사용하는 것보다 훨씬 정형화되고 사용하기 편한 지식베이스를 구축할 있도록 해준다. 이는 점점 데이터가 많아지고 그 분류가 사람의 손으로 거치기 어려워지는 현 태세의 상황을 보면 앞으로 정확한 전역 관계 추출의 가치는 점점 높아질 것으로 보인다.

도면의 간단한 설명

[0007] 도 1은 LSTM 기반 단편적 관계 추출기를 도시한 도면이다.

도 2는 외부 메모리 신경망을 통한 메타학습을 나타낸 도면이다.

발명을 실시하기 위한 구체적인 내용

[0008] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다.

[0009] 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.

[0011] 이 발명에서 쓰이는 모델은 크게 2 부분으로 이루어져 있으며 훈련은 3단계에 걸쳐서 진행된다. 모델은 단편적 관계 추출 모델과 외부 메모리 신경망으로 이루어져 있다. 그리고 훈련은 각각 단편적 관계 추출 모델의 훈련, 전역 관계를 위한 메모리 증강 신경망 훈련, 마지막으로 메모리 증강 신경망 훈련의 결과를 반영한 관계 추출 모델의 재훈련으로 총 3단계가 존재한다.

[0012] 단편적인 관계 추출을 시도하기에 앞서 먼저 텍스트의 전처리가 요구된다. 텍스트의 전처리는 문장 분리 및 문장 내 개체명 인식(Named Entity Recognition), 그리고 마지막으로 문장 내의 관계를 연결(Entity Linking) 및 분류(Relation Classification)할 2 개체를 마킹하는 것이다. 해당 작업은 기존에 이미 존재하는 모델을 이용함으로써 해결하였으며 각 개체 간 마킹은 문장 내의 모든 개체들 간 관계를 추출 할 때까지 각각 마킹하여 관계를 파악한다.

[0013] 단편적 관계 추출 모델:

[0014] 전처리를 마친 텍스트는 아래 도면 1과 같은 LSTM 기반의 단편적 관계 추출 모델에 입력된다. 이 단계에서는 텍스트를 각 문장 단위로 분석하기 시작한다. 이때 문장 x에 존재하는 각 토큰 $x=\{x_1,x_2,\cdots,x_n\}$들을 k 차원의 벡터에 임베딩하기 위하여 행렬 $W\in R^{|V|\times k}$를 사용한다. 이때 $|V|$는 vocabulary의 크기를 나타낸다. 입력된 문장에 마킹된 개체 1과 개체 2를 기준으로 타 문장요소들인 토큰의 위치가 개체1 또는 개체 2에 더 가까운 지, 아니면 그 어느 쪽 개체와도 거리가 먼지에 따라 각각의 위치를 마킹한다. 이는 구문분석(Part of Speech Tagging)을 통해 이루어지며 각 토큰을 문법적으로 더 가까운 개체 쪽으로 분류하도록 하는 것이다. 그리고 일정 임계값이 넘는 거리가 나오면 그 어느 쪽과도 관계가 없는 것으로 표기하는 것이다. 이렇게 만들어진 토큰 임베딩들은 LSTM신경망(Long Short Term Memory Network)을 통하여 n개의 벡터들을 고정된 크기의 출력 벡터로 변환한다. 이 출력 벡터는 문장 내 관계 인코딩을 시도하는 2 개체의 관계를 나타낸다. 학습된 모델은 해당 출력 벡터 를 소프트맥스 레이어에(Softmax Layer) 입력하여 해당 개체 쌍의 관계여부의 판별 및 관계 분류를 통해 언급 수준의 관계 추출을 실행한다. 여기서 관계 분류기를 $P(r)$, 소프트맥스 레이어를 f_i, 첫 번째 개체와 두 번째 개체를 e_1, e_2라고 정의하였을 때 관계 분류기는 다음과 같다.

수학식 1

[0015] $$P(r|\langle e_1, e_2\rangle, x; \theta) = \frac{\exp(f_r)}{\sum_{i=1}^{n_r}\exp(f_i)} \qquad f_i = w_i \cdot o_s + b_i$$

[0016] 이때 w_i는 무게 벡터이며 b_i는 바이어스이다.

[0017] 외부 메모리:

[0018] 단편적 관계 추출 모델에서 얻어낸 결과들은 그러나 어디까지나 언급 수준의 관계들로 여러 문장에 걸쳐져있는 관계를 가진 개체들을 분석하기 위해서는 추가적인 조치가 요구된다. 그렇기 때문에 텍스트 내의 관계들에 대한 종합적인 분석이 요구된다. 그렇기 때문에 1차적으로 외부 메모리에 텍스트에서 얻어낸 언급 수준의 관계들의 출력벡터를 차례대로 저장하며 문장을 건너뛰는 관계 후보들을 찾는다. 이후 각 개체 사이에 공유하고 있는 중간 단계 개체들이 존재한다면 이러한 관계들을 모두 통합하여 새로운 예측 벡터를 생성한다. 이후 생성된 예측 벡터는 훈련된 메모리 증강 신경망을 통하여 타 문장에 속하는 개체들 간의 관계들을 정의한다. 이 때 i번째 순서에 입력된 언급 수준의 관계를 r_i라고 정의하고, r_i가 속해있는 전역 관계를 R이라고 한다. 이때 R의 양 끝에 해당하는 전역 관계를 대표하는 개체를 E_i로 표기하며 그에 속한 개체들은 e_i로 표기한다. 그에 해당 이러한 전역관계를 나타내는 예측 벡터 V_R은 다음과 같다.

수학식 2

[0019] $$V_R = softmax(\sum_{i}^{t} r_i(t) \times \frac{d_t}{D})$$

[0020] 이후 V_R을 분류하는 데에는 코사인 유사도를 사용하여 해당 전역관계가 어떠한 분류에 속하는지 파악한다.

[0021] 훈련 과정:

[0022] 소개되는 2 가지 모델을 훈련시키기 위해서는 총 3단계의 훈련이 필요하다. 첫 번째 훈련은 단편적 관계 추출 모델에 대한 학습으로, 이는 두 개체가 마킹이 된 구문 부석이 이루어진 문장에서 정답 개체 간 관계를 통해 End-to-End 방식의 훈련을 통해 학습이 이루어졌다. 이는 외부 메모리에서 전역적 관계 추출을 학습할 때에도 마찬가지다. 다만 이 경우에는 텍스트를 통한 학습이 아닌 여러 관계 tuple을 제공하며 전역적 관계로 이어지는 여러 tuple들의 합성곱을 통한 예측 벡터와 주어진 정답을 통한 End-to-End 방식의 훈련으로 1차 훈련을 마친다.

[0023] 1차적인 훈련을 마친 뒤에는 메모리 증강 신경망을 통한 2차 훈련을 시작한다. 이는 전역적 관계를 언급 수준의 관계에 반영하는 것이다. 이때는 각 전역관계 R에서 순서 i에 해당되는 개체들과 R의 주요개체 간 관계추출을 재 정의할 때 해당되는 문장을 x_i라고 정의한다. 또한 해당 문장 내의 존재하는 전역관계 R에 속하는 개체를 k_i라고 정의하며 k_i가 속한 언급 수준의 관계를 외부 메모리에 있는 R의 타 관계들과 비교하게 된다. 이때 k_i가 속해있는 R 중 그 예측 벡터 V_R이 해당 문장 내의 k_i가 포함된 언급 수준의 관계와 유사한지 확인한다. 이를 $M(k_i)$라고 하여 코사인 유사도를 이용하여 유사도를 계산한다.

수학식 3

[0024] $$K(k_t, M(k_t)) = \frac{k_t \cdot M(k_t)}{\| k_t \| \cdot \| M(k_t) \|}$$

[0025] 이 계산을 통하여 무게 벡터 w_t를 생성할 수 있는데, 무게 벡터 w_t을 이용하여 문장 x_t 내의 개체 k_t가 속한 관계를 전역관계 R을 반영하여 재정의해야 또는 변화를 주어야하는지, 그리고 준다면 어떻게 변화를 주는지 결정하게 된다. w_t을 구하는 방법은 다음과 같다.

수학식 4

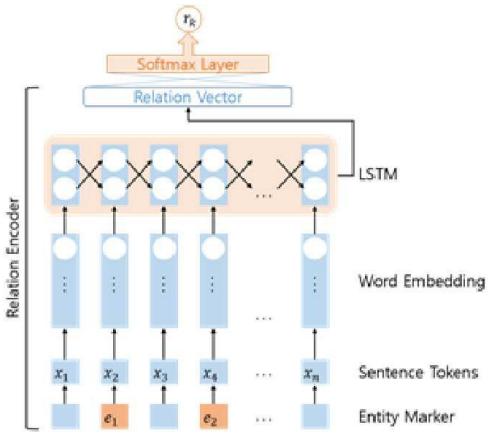
[0026] $$w_t \leftarrow \frac{\exp(K(k_t, M(k_t)))}{\Sigma_j \exp(K(k_t, M(j)))}$$

[0027] 구한 w_t가 임계값보다 클 경우 언급 수준의 관계 예측 벡터인 o_t에 합산곱을 통하여 새로운 o_t을 생성하도록 하며 단편적 관계 추출 모델을 재훈련시킨다. 1차 훈련을 반복한 뒤 다시 이를 통한 2차 훈련을 반복하여 성능이 최대한 높아질 때까지 이 과정을 반복한다.

[0028] 상기한 본 발명의 실시예는 예시의 목적을 위해 개시된 것이고, 본 발명에 대한 통상의 지식을 가지는 당업자라면 본 발명의 사상과 범위 안에서 다양한 수정, 변경, 부가가 가능할 것이며, 이러한 수정, 변경 및 부가는 하기의 특허청구범위에 속하는 것으로 보아야 할 것이다.

도면

도면1



도면2

