(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2019년07월11일
(11) 등록번호 10-1999152
(24) 등록일자 2019년07월05일

(51) 국제특허분류(Int. Cl.)
G06F 17/27 (2006.01) *G06N 3/08* (2006.01)
(52) CPC특허분류
G06F 17/2705 (2013.01)
G06F 17/2755 (2013.01)
(21) 출원번호 10-2017-0182571
(22) 출원일자 2017년12월28일
심사청구일자 2017년12월28일
(65) 공개번호 10-2019-0080234
(43) 공개일자 2019년07월08일
(56) 선행기술조사문헌
김현지 외, 컨볼루션 신경망을 이용한 구인 광고 데이터 정형화 시스템, 한국정보처리학회 2018 춘계학술발표대회논문집 v.25 no.1 (2018.05)
김도우, Doc2Vec을 활용한 CNN 기반 한국어 신문 기사 분류에 관한 연구, 서강대학교 석사학위 논문 (2017.01)
US20170139899 A1
KR101847847 B1

(73) 특허권자
포항공과대학교 산학협력단
경상북도 포항시 남구 청암로 77 (지곡동)
(72) 발명자
한욱신
경상북도 포항시 남구 청암로 77 창의IT융합공학과 (지곡동, 포항공과대학교)
김현지
울산광역시 동구 월봉12길 50, C동 308호 (화정동, 송정타워맨션3차)
(뒷면에 계속)
(74) 대리인
특허법인이룸리온

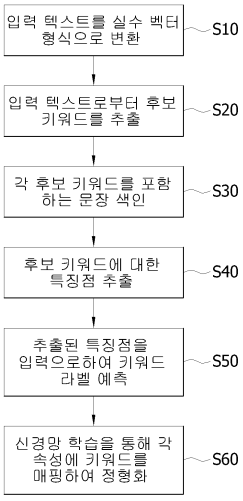
전체 청구항 수 : 총 13 항 심사관 : 경연정

(54) 발명의 명칭 **컨벌루션 신경망 기반 영문 텍스트 정형화 방법**

(57) 요 약

본 발명은 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 관한 것으로, 영문 텍스트로 이루어진 문서의 집합을 미리 정의된 데이터 스키마를 가진 정형 데이터로 변환하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 있어서, a) 입력 텍스트를 전처리하여 입력 텍스트를 실수 벡터 형식으로 변환하는 단계와, b) 입력 텍스트로부터 후보 키워드를 추출하는 단계와, c) 각 후보 키워드를 포함하는 문장을 색인하는 단계와, d) 상기 후보 키워드에 대한 특징점을 추출하는 단계와, e) 추출한 특징점을 입력 데이터로 하여 키워드의 라벨을 예측하는 단계와, f) 상기 산출된 키워드 속성을 네거티브 샘플링 기반 신경망 학습처리를 수행하여 각 속성에 키워드를 매핑하는 단계를 포함한다.

대 표 도 - 도2



(52) CPC특허분류
G06F 17/277 (2013.01)
G06N 3/084 (2013.01)
(72) 발명자
박용희
경상북도 포항시 남구 지곡로 102, 6동 504호(지곡동, 포스빌)
김경민
충청남도 천안시 동남구 풍세로 769-28, 114동 703호(용곡동, 용곡마을세광2차엔리치타워아파트)
소병훈
전라북도 전주시 덕진구 솔내8길 25, 1110호(송천동1가, 광진한솔아파트)

이 발명을 지원한 국가연구개발사업
과제고유번호 R03461510070001002
부처명 과학기술정보통신부
연구관리전문기관 정보통신기술진흥센터
연구사업명 IT명품인재양성사업
연구과제명 [후원금_산학수익]포스텍 미래 IT 융합연구원
기 여 율 1/1
주관기관 포항공과대학교 산학협력단
연구기간 2017.01.01 ~ 2017.12.31

명 세 서

청구범위

청구항 1

컴퓨터를 이용하여 영문 텍스트로 이루어진 문서의 집합을 미리 정의된 데이터 스키마를 가진 정형 데이터로 변환하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 있어서,

a) 입력 텍스트를 전처리하여 입력 텍스트를 실수 벡터 형식으로 변환하는 단계;

b) 입력 텍스트로부터 후보 키워드를 추출하는 단계;

c) 각 후보 키워드를 포함하는 문장을 색인하는 단계;

d) 상기 후보 키워드에 대한 특징점을 추출하는 단계;

e) 추출한 특징점을 입력 데이터로 하여 키워드의 라벨을 예측하는 단계; 및

f) 상기 추출된 키워드 특징점을 네거티브 샘플링 기반 신경망 학습처리를 수행하여 각 속성에 키워드를 매핑하는 단계를 포함하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 2

제1항에 있어서,

상기 a) 단계는,

비정형 데이터인 비정형 데이터인 영문 텍스트를 문장 단위로 분리한 뒤 문장 내의 각 단어를 식별하는 토큰화 단계; 및

상기 식별된 각 단어를 벡터로 표현하고, 각 단어에 대한 품사 태깅 및 개체명 클래스를 통해 벡터화하는 임베딩 단계를 포함하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 3

제1항에 있어서,

상기 b) 단계는,

최소 1에서 최대 k_{max}(k는 양의 정수) 크기를 가지는 윈도우(window)를 이용하여 텍스트의 모든 문장에 포함된 키워드를 추출하되,

키워드 시퀀스가 텍스트 내에 여러 번 등장하는 경우 중복 시퀀스를 제거하여 키워드 집합을 추출하는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 4

제3항에 있어서,

상기 c) 단계는,

추출된 각 키워드를 포함하는 모든 문장의 집합을 상기 입력된 텍스트에서 추출하는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 5

제4항에 있어서,

상기 문장의 집합을 추출하는 과정은 상기 키워드 집합을 구하는 과정에서 각 키워드에 대해, 키워드를 포함하는 문장의 인덱스 정보를 함께 저장하고, 상기 문장의 인덱스 정보를 이용하여 문장의 집합을 추출하는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 6

제1항에 있어서,

상기 d) 단계는,

키워드를 단어 임베딩 및 태그 임베딩 벡터를 구한 후, 두 벡터에 어텐션 기법을 적용하여 임베딩 벡터를 산출하여, 키워드가 포함하는 단어의 개수에 관계없이 키워드의 임베딩 벡터의 크기가 일정하게 유지되도록 하는 단어 특징점 추출과정과,

상기 키워드의 단어 임베딩 벡터로 변환한 결과를 이용하여 각 문장 내의 키워드와의 상대위치를 구하여 문장 특징점을 구하는 문장 특징점 추출과정을 포함하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 7

제1항에 있어서,

상기 e) 단계는,

상기 특징점을 소프트맥스 회귀(Softmax regression)의 입력 값으로 하여, 키워가 속하는 속성과 점수를 계산하는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 8

제1항에 있어서,

상기 f) 단계는,

상기 추출된 키워드 중 정형 데이터에 저장된 값과 일치하지 않는 타겟 키워드의 경우 NA를 해당 키워드의 라벨로 설정하는 단계;

상기 라벨이 정해진 키워드들 중에서 네거티브 샘플링을 통해 데이터 스키마의 속성을 라벨로 가진 키워드 개수와 NA를 라벨로 가진 키워드의 개수의 비율을 조절하는 단계; 및

상기 설정된 타겟 키워드와 라벨을 입력받아 컨볼루션 신경망 기반 문장 특징점(sentence feature)/어텐션 신경망 기반 키워드 특징점(keyword feature)를 포함하는 신경망을 학습하는 단계를 포함하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 9

삭제

청구항 10

제8항에 있어서,

상기 신경망은,

컨벌루션 신경망을 이용하여 계산한 타겟 키워드를 포함하는 문장의 벡터, 타겟 키워드의 임베딩 벡터, 타겟 키워드의 형태소 분석값/개체명 인식값의 임베딩 벡터, 문장 내 타겟 키워드 직전 단어의 형태소 분석값/개체명 인식값의 임베딩 벡터, 및 문장 내 타겟 키워드 직후 단어의 형태소 분석값/개체명 인식값의 임베딩 벡터를 특징점으로 포함하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 11

삭제

청구항 12

제10항에 있어서,

상기 특징점에서 타겟 키워드의 임베딩 벡터는,

타겟 키워드가 2개 이상의 단어로 이루어진 경우 타겟 키워드 각 단어의 임베딩 벡터에 어텐션 신경망을 적용하

여 계산되는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 13

제10항에 있어서,

상기 특징점에서 컨벌루션 신경망을 이용하여 문장의 벡터를 계산하는 과정은,

문장 내의 각 단어를 임베딩 벡터로 변환한 값과 문장 내의 타겟 키워드에 대한 문장 내의 각 단어의 상대적인 위치를 임베딩 벡터로 변환한 값을 입력으로 받는 단일 레이어 컨볼루션 신경망을 이용하는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 14

제10항에 있어서,

상기 특징점에서 타겟 키워드 및 타겟 키워드 전후 단어의 형태소 분석값/개체명 인식값의 임베딩 벡터는 신경망 학습 과정에서 가변성을 갖는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

청구항 15

제13항에 있어서,

상기 컨벌루션 신경망의 입력값 중 단어의 상대적인 위치를 임베딩 벡터로 변환한 값은 신경망 학습과정에서 가변성을 갖는 것을 특징으로 하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 관한 것으로, 더 상세하게는 컨벌루션 신경망 모형을 활용하여 키워드가 입력으로 주어지지 않고, 세 개 이상의 속성을 가지는 임의의 데이터 스키마가 주어지는 경우에도 효율적으로 정형화할 수 있는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 관한 것이다.

배 경 기 술

[0002] 일반적으로, 텍스트 데이터 정형화는 텍스트와 데이터 스키마가 주어졌을 때 입력된 텍스트로부터 주어진 데이터 스키마에 대응되는 최적의 키워드를 찾아내는 것을 뜻한다.

[0003] 텍스트 데이터 정형화는, 자연어 처리 및 인공 지능 분야, 데이터베이스 등 다양한 분야에서 중요한 핵심적인 기술 중 하나로서 많은 응용 프로그램에서 사용되고 있다. 최근 정형화 문제를 해결하는 많은 신경망 기반 알고리즘들이 제안되었으나, 기존의 모든 알고리즘들이 키워드의 후보가 입력으로 주어진다고 가정하고 있으며, 대부분의 알고리즘은 두 개의 속성(attribute)을 가지는 이진 관계(binary relation)만 처리할 수 있다는 한계가 있다.

[0004] 선행문헌1(Mary Elaine Califf and Raymond J. Mooney: Relational Learning of Pattern-Match Rules for Information Extraction. AAAI/IAAI 1999: 328-334)에서는 영문 텍스트와 데이터 스키마가 주어졌을 때, 영문 텍스트를 관계형 데이터 베이스(Relational database)의 튜플(Tuple)로 변환하는 규칙 학습 기반 정형화 기술 RAPIER를 제안하였다. RAPIER는 규칙을 학습하기 위해 세 가지 패턴을 정의한다.

[0005] 첫 번째는 필러(filler) 바로 앞의 텍스트와 일치하는 사전-필러(Pre-filler) 패턴이다. 두 번째는 실제 텍스트와 일치하는 필러 패턴이다. 마지막 세 번째는 필러 바로 뒤에 나오는 텍스트와 일치하는 사후-필러(Post-filler) 패턴이다.

[0006] RAPIER는 이 세가지 패턴을 이용해 데이터 정형화를 위한 규칙을 학습한다. RAPIER는 패턴 규칙 학습을 기반으로 하여 텍스트 데이터에서 빈번하게 등장하는 패턴을 자동으로 인식하고 정형 데이터를 추출한다. 따라서 다른 알고리즘에 비해 높은 정확도를 나타낸다. 그러나 데이터에 등장한 적이 없는 패턴을 처리할 수 없기 때문에 재현율이 매우 낮다는 한계점을 가진다.

[0007] 최근 지식 베이스 시스템의 발전에 따라 지도 학습에 필요한 데이터의 양이 증가하였고, 이에 따라 신경망을 이

용한 정형화 기술이 활발히 연구되고 있다. 신경망 기반 정형화 연구들은 크게 이진 관계 분류(binary relation classification), 슬롯 채우기(slot filling), N항 관계 추출(N-ary relation extraction) 연구로 나눌 수 있다.

[0008] 이진 관계 분류에 대한 연구는 컨벌루션 신경망 네트워크(convolutional neural networks) 기반의 방법을 소개하는 선행문헌2(Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao: Relation Classification via Convolutional Deep Neural Network. COLING 2014: 2335-2344), 선행문헌3(Daojian Zeng, Kang Liu, Yubo Chen, Jun Zhao: Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. EMNLP 2015: 1753-1762), 선행문헌4(Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, Maosong Sun: Neural Relation Extraction with Selective Attention over Instances. ACL (1) 2016)가 있다.

[0009] 또한, 순환 신경망/장단기 기억 네트워크(recurrent neural networks/Long Short Term Memory networks)기반의 방법을 소개하는 선행문헌5(Yang Liu, Furu Wei, Sujian Li, Heng Ji, Ming Zhou, Houfeng Wang: A Dependency-Based Neural Network for Relation Classification. ACL (2) 2015: 285-290) 및 선행문헌6(Daniil Sorokin, Iryna Gurevych: Context-Aware Representations for Knowledge Base Relation Extraction. EMNLP 2017: 1785-1790)이 있다.

[0010] 상기 선행문헌2는 컨벌루션 신경망 네트워크 기반 이진 관계 분류 방법으로, 입력으로 주어진 두 개의 명사에 대해 특징점을 추출하고 컨벌루션 신경망 접근법을 사용해 문장 수준의 특징점을 학습하여 두 개의 명사 사이 관계를 예측한다.

[0011] 선행문헌3은 컨벌루션 신경망 접근법에 개별적 최대값 선정(piecewise max pooling) 기법을 적용하여 기존 특징점 추출 방식의 오류를 개선한 알고리즘을 제안하였다.

[0012] 선행문헌4는 선택적 어텐션(selective attention) 기법을 적용한 컨벌루션 신경망 접근법을 사용하여 학습 모형의 정확도를 향상시킴과 동시에, 잘못 분류 표시된 학습 데이터(labeled data)의 문제를 해결하였다.

[0013] 선행문헌5는 문장의 종속 관계 기반 파스 트리(dependency-based parse tree)에서 두 키워드 사이의 최단 거리를 구한 뒤, 종속 관계 기반 순환 신경망을 이용하여, 최단 거리 상의 단어가 가지는 의미를 분석함으로써 관계 분류 문제 성능을 향상시켰다.

[0014] 선행문헌6은 장단기 기억 신경망 네트워크 접근법을 사용하였는데, 데이터셋 내에 두 키워드 중 어느 하나의 키워드만을 포함하는 데이터를 컨텍스트로 정의하고, 이 컨텍스트를 신경망의 입력값으로 하여 기존보다 정밀한 관계 추출이 가능한 학습 모형을 제안하였다.

[0015] 이처럼 최근 이진 관계 분류 문제를 해결하기 위한 다양한 신경망 모형이 제안되고 있지만, 이러한 기술들은 모두 2개 키워드 사이의 관계를 추출하는 문제에 한정되어 있다.

[0016] 따라서 스키마의 속성이 2개라고 가정하기 때문에, 3개 이상의 속성을 가지는 정형 데이터를 처리할 수 없다는 문제점이 있었다.

[0017] 또한, 앞서 설명한 연구들에서는 2개의 키워드가 입력으로 주어지며, 2개의 키워드는 한 문장 내에 동시에 등장한다고 가정하고 있기 때문에 정형화 처리가 매우 한정적인 문제점이 있었다.

[0018] 다음으로, 슬롯 채우기는 텍스트 뭉치(corpus)와 키워드 그리고 하나의 속성이 주어졌을 때, 키워드와 해당 속성 관계를 갖는 또 다른 키워드를 텍스트 뭉치 내에서 찾아내는 문제이다.

[0019] 선행문헌7(Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, Christopher D. Manning: Position-aware Attention and Supervised Data Improve Slot Filling. EMNLP 2017: 35-45)에는 장단기 기억 네트워크를 이용한 방법으로, 키워드 위치 정보를 활용하여 키워드와 속성 관계를 갖는 다른 키워드 추출 성능을 향상시켰다. 슬롯 채우기 문제는 N항 관계를 추출할 수 있으나, 키워드가 입력으로 주어지는 것에 한정되어 있는 문제점이 있었다.

[0020] 한편, 최근 그래프 기반 장단기 기억 네트워크(Graph LSTM)를 활용한 N항 관계 추출 연구가 제안되었다.

[0021] 선행문헌8(Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, Wen-tau Yih: Cross-Sentence N-ary Relation Extraction with Graph LSTMs. TACL 5: 101-115 (2017))에서는 N항 관계 추출 연구는 텍스트, N항 관계의 목록이 주어졌을 때, 텍스트에서 N개 키워드의 순서쌍이 관계 표현 목록 중 어떤 관계에 매핑되는지 찾아내는 문제이다. 그러나 N항 관계 추출 연구 또한 N개의 키워드 순서쌍이 입력 자료로 주어진다고 가정하고

있기 때문에 입력 자료가 주어지지 않는 경우에는 수행할 수 없는 문제점이 있었다.

[0022] 다크 데이터(Dark Data)는 정형화 되지 않은 형식으로 되어 있는 데이터의 방대한 집합인데, 공공기관 또는 회사의 공문서, 웹 페이지 등이 이에 속한다. 기업의 수많은 데이터 중 활용되는 데이터(Smart Data)는 12% 에 불과하다[16]. 나머지 이용되지 않는 데이터(다크 데이터) 88%는 대부분 비정형 데이터이기 때문에 상용 DBMS로 처리가 불가하여 거의 활용되지 않고 있다. 비정형 데이터의 정형화는 비정형 데이터가 주어졌을 때 이를 활용성이 높은 정형 데이터로 변환하는 문제로, 이 때 비정형 데이터는 텍스트, 사진, 음성 등을 포함할 수 있다.

[0023] 도 1은 비정형 데이터를 정형 데이터로 처리하는 예시도이다.

[0024] 도 1을 참조하면, (a)는 비정형 텍스트 데이터, (b)는 정형 데이터의 스키마이다. 이 예에서 비정형 텍스트는 영문 텍스트 데이터로서 문장을 이룬다. 이러한 문장에서 주요한 텍스트를 추출하고, (b)와 같이 사람과 국가를 표로 정리하는 정형 데이터의 스키마에 적용하기 위해서는 사람의 이름과 국가에 대한 텍스트 정보를 추출해야 한다.

[0025] (a)를 (b)에 대하여 정형화하면, (c)의 정형 데이터를 구할 수 있다. 이처럼 비정형 데이터를 정형 데이터로 변환할 수 있다면 활용도가 낮은 88%의 다크 데이터를 다양하게 활용할 수 있는 장점이 있다. 또한, 정형 데이터와 비정형 데이터의 분석 소프트웨어를 통합적으로 관리할 수 있는 장점과 정형화된 데이터를 가지고 상용 DBMS의 기능을 모두 활용할 수 있어 고도화된 분석이 가능하다는 장점이 있다.

[0026] 비정형 텍스트 데이터의 정형화 기법은 인신 매매 탐지, 유전학 기반 데이터 마이닝, 지질학과 고생물학 데이터 분석 등에서 폭넓게 활용되고 있다. 인신 매매 탐지에서는, 성 광고 텍스트에서 찾아낸 전화번호와 이메일을 광고가 등장한 위치 정보와 결합하여 해당 광고주에 의해 발생하는 성거래의 범위와 네트워크를 찾아내는 데 정형화 기술을 활용한다.

[0027] 유전학에서는 비정형 텍스트로 되어있는 문헌에서 유전자와 유전자 변이형 및 표현형을 정형 데이터로 저장하고, 이를 분석하여 유전자들의 관계를 추론한 결과를 통해 임상 유전 진단, 생식 카운셀링 등에 적용한다.

[0028] 또한, 지질학과 고생물학의 비정형 데이터를 심층적으로 분석하기 위해 정형화 기법을 활용한다.

[0029] 최근 들어 다양한 분야에서 비정형 텍스트 데이터를 분석해 활용 가능한 데이터를 추출하는 신경망 기반 기술이 활발히 연구되고 있다. 신경망 기반 정형화 기술은 기존 규칙 기반 기술과는 달리 도메인 지식을 요구하지 않고 다양한 데이터 스키마에 폭넓게 적용 가능하다는 장점이 있다. 또한, 규칙 기반 방법과 대비하여 높은 수준의 재현율을 가진다는 장점이 있다.

[0030] 그러나 앞서 설명한 바와 같이 많은 선행 연구들이 2개의 키워드 사이의 이진 관계를 추출하는 문제에 한정되어 있다. 이 연구들은 데이터 스키마의 속성이 2개라고 가정하고 있기 때문에, 3개 이상의 속성을 가지는 다양한 실세계의 정형 데이터를 추출할 수 없다.

[0031] 또한, 이 연구들은 추론 과정에서 타겟으로 하는 키워드가 입력으로 주어진다고 가정 한다. 그러나 실제 어플리케이션에서는 텍스트 내의 수많은 키워드 중에 중요한 키워드가 무엇인지 알 수 없는 경우가 많다.

[0032] 마지막으로, 이 연구들은 2개의 키워드를 모두 포함하는 한 문장을 입력받고 이를 분석하여 2개의 키워드 사이의 관계를 추출하는데, 텍스트에2개의 키워드가 중복해서 등장하는 경우나, 한 문장에 2개의 키워드가 동시에 등장하지 않는 경우를 고려하지 않는다. 따라서, 텍스트로부터 주어진 데이터 스키마에 대한 엔티티를 추출해내기 위해서는 기존 방법들을 그대로 사용할 수 없다.

[0033] 한편, 세 개 이상의 속성으로 구성된 데이터 스키마를 처리할 수 있는 선행 연구로 앞서 설명한 바와 같이 슬롯 채우기(slot filling)가 있다. 슬롯 채우기는 문서 코퍼스와 키워드, 속성 목록이 입력으로 주어졌을 때, 키워드가 어떤 속성에 해당하는 엔티티인지 찾아내는 문제이다. 그러나 이 연구들은 앞서 설명한 케이스와 마찬가지로 키워드가 입력으로 주어진다고 가정하고 있다. 또한, 전체 문서 집합에서 키워드의 속성이 유일하게 결정되기 때문에 키워드가 입력으로 주어지지 않고, 텍스트에 따라 키워드의 속성이 달라지는 경우에 이 방법을 어떻게 확장하느냐는 어려운 이슈이다.

[0034] 최근 여러 개의 문장으로 구성된 텍스트 데이터에서 N항 관계를 추출하는 연구가 제안되었으나, 이 연구 또한 N개의 키워드가 입력으로 주어진다는 한계를 가진다.

발명의 내용

해결하려는 과제

[0035] 본 발명이 해결하고자 하는 기술적 과제는, 영문 텍스트와 테이블의 스키마가 주어졌을 때, 텍스트에 등장한 키워드 별로 특징점을 추출하고, 추출된 특징점을 입력으로 하는 신경망을 이용하여 데이터 스키마의 각 속성에 키워드를 매핑하는 효율적인 정형화 기법을 제공함에 있다.

[0036] 특히, 이진 관계 추출이 아닌 3개 이상의 속성(N항 관계)을 가지는 정형 데이터를 처리하는 경우, 서로 다른 문장에 속하는 후보 키워드 간의 관계를 고려하여 처리할 수 있는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법을 제공함에 있다.

[0037] 또한, 본 발명이 해결하고자 하는 다른 기술적 과제는, 텍스트가 여러 개의 문장으로 구성되고, 타겟 키워드가 입력으로 주어지지 않는 경우에도 데이터를 정형화할 수 있는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법을 제공함에 있다.

과제의 해결 수단

[0038] 상기와 같은 과제를 해결하기 위한 컨벌루션 신경망 기반 영문 텍스트 정형화 방법은, 영문 텍스트로 이루어진 문서의 집합을 미리 정의된 데이터 스키마를 가진 정형 데이터로 변환하는 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 있어서, a) 입력 텍스트를 전처리하여 입력 텍스트를 실수 벡터 형식으로 변환하는 단계와, b) 입력 텍스트로부터 후보 키워드를 추출하는 단계와, c) 각 후보 키워드를 포함하는 문장을 색인하는 단계와, d) 상기 후보 키워드에 대한 특징점을 추출하는 단계와, e) 추출한 특징점을 입력 데이터로 하여 키워드의 라벨을 예측하는 단계와, f) 추출된 상기 키워드 특징점을 네거티브 샘플링 기반 신경망 학습처리를 수행하여 각 속성에 키워드를 매핑하는 단계를 포함한다.

[0039] 본 발명의 일실시예에 따르면, 상기 a) 단계는, 비정형 데이터인 비정형 데이터인 영문 텍스트를 문장 단위로 분리한 뒤 문장 내의 각 단어를 식별하는 토큰화 단계와, 상기 식별된 각 단어를 벡터로 표현하고, 각 단어에 대한 품사 태깅 및 개체명 클래스를 통해 벡터화하는 임베딩 단계를 포함할 수 있다.

[0040] 본 발명의 일실시예에 따르면, 상기 b) 단계는, 최소 1에서 최대 k_{max}(k는 양의 정수) 크기를 가지는 윈도우(window)를 이용하여 텍스트의 모든 문장에 포함된 키워드를 추출하되, 키워드 시퀀스가 텍스트 내에 여러 번 등장하는 경우 중복 시퀀스를 제거하여 키워드 집합을 추출할 수 있다.

[0041] 본 발명의 일실시예에 따르면, 상기 c) 단계는, 추출된 각 키워드를 포함하는 모든 문장의 집합을 상기 입력된 텍스트에서 추출할 수 있다.

[0042] 본 발명의 일실시예에 따르면, 상기 문장의 집합을 추출하는 과정은 상기 키워드 집합을 구하는 과정에서 각 키워드에 대해, 키워드를 포함하는 문장의 인덱스 정보를 함께 저장하고, 상기 문장의 인덱스 정보를 이용하여 문장의 집합을 추출할 수 있다.

[0043] 본 발명의 일실시예에 따르면, 상기 d) 단계는, 키워드를 단어 임베딩 및 태그 임베딩 벡터를 구한 후, 두 벡터에 어텐션 기법을 적용하여 임베딩 벡터를 산출하여, 키워드가 포함하는 단어의 개수에 관계없이 키워드의 임베딩 벡터의 크기가 일정하게 유지되도록 하는 단어 특징점 추출과정과, 상기 키워드의 단어 임베딩 벡터로 변환한 결과를 이용하여 각 문장 내의 키워드와의 상대위치를 구하여 문장 특징점을 구하는 문장 특징점 추출과정을 포함할 수 있다.

[0044] 본 발명의 일실시예에 따르면, 상기 e) 단계는, 상기 특징점을 소프트맥스 회귀(Softmax regression)의 입력 값으로 하여, 키워가 속하는 속성과 점수를 계산할 수 있다.

[0045] 본 발명의 일실시예에 따르면, 상기 f) 단계는, 상기 설정된 키워드 중 정형 데이터에 저장된 값과 일치하지 않는 타겟 키워드의 경우 NA를 해당 키워드의 라벨로 설정하는 단계와, 상기 라벨이 정해진 키워드들 중에서 네거티브 샘플링을 통해 데이터 스키마의 속성을 라벨로 가진 키워드 개수와 NA를 라벨로 가진 키워드의 개수의 비율을 조절하는 단계와, 상기 설정된 타겟 키워드와 라벨을 입력받아 컨볼루션 신경망 기반 문장 특징점(sentence feature)/어텐션 신경망 기반 키워드 특징점(keyword feature)를 포함하는 신경망을 학습하는 단계를 포함할 수 있다.

[0046] 본 발명의 일실시예에 따르면, 상기 학습한 신경망을 토대로 새로운 텍스트 문서를 정형 데이터로 추론하는 단계와, 상기 추론 단계에서 문서 내의 키워드를 상기 b) 단계의 방법으로 설정하는 단계와, 상기 설정된 키워드에 대해 데이터 스키마의 각 속성 및 NA를 대상으로 관련도 점수를 계산하는 단계와, 상기 계산한 관련도 점수를 랭킹하여 타겟 키워드를 하나의 데이터 스키마 속성 혹은 NA에 매핑하는 단계와, 상기 매핑 단계에서 타겟 키워드가 단일 문서 내에 복수 등장할 경우 가장 높은 관련도 점수를 가지는 데이터 스키마의 속성 혹은 NA에 매핑하는 단계와, 상기 추론 단계에서 타겟 키워드의 관련도 점수가 한계점을 넘지 못할 경우 NA에 매핑하는 단계를 더 포함할 수 있다.

[0047] 본 발명의 일실시예에 따르면, 상기 신경망은, 컨벌루션 신경망을 이용하여 계산한 타겟 키워드를 포함하는 문장의 벡터, 타겟 키워드의 임베딩 벡터, 타겟 키워드의 형태소 분석값/개체명 인식값의 임베딩 벡터, 문장 내 타겟 키워드 직전 단어의 형태소 분석값/개체명 인식값의 임베딩 벡터, 및 문장 내 타겟 키워드 직후 단어의 형태소 분석값/개체명 인식값의 임베딩 벡터를 특징점으로 포함할 수 있다.

[0048] 본 발명의 일실시예에 따르면, 매핑하는 단계에서 데이터 스키마의 속성 중에 단일 레코드를 저장하는 속성의 경우, 해당 속성에 매핑된 타겟 키워드가 여러 개일 경우 관련도 점수로 타겟 키워드를 랭킹하여 가장 높은 관련도 점수를 가지는 타겟 키워드만 저장할 수 있다.

[0049] 본 발명의 일실시예에 따르면, 상기 특징점에서 타겟 키워드의 임베딩 벡터는, 타겟 키워드가 2개 이상의 단어로 이루어진 경우 타겟 키워드 각 단어의 임베딩 벡터에 어텐션 신경망을 적용하여 계산될 수 있다.

[0050] 본 발명의 일실시예에 따르면, 상기 특징점에서 컨벌루션 신경망을 이용하여 문장의 벡터를 계산하는 과정은, 문장 내의 각 단어를 임베딩 벡터로 변환한 값과 문장 내의 타겟 키워드에 대한 문장 내의 각 단어의 상대적인 위치를 임베딩 벡터로 변환한 값을 입력으로 받는 단일 레이어 컨볼루션 신경망을 이용할 수 있다.

[0051] 본 발명의 일실시예에 따르면, 상기 특징점에서 타겟 키워드 및 타겟 키워드 전후 단어의 형태소 분석값/개체명 인식값의 임베딩 벡터는 신경망 학습 과정에서 가변성을 가질 수 있다.

[0052] 본 발명의 일실시예에 따르면, 상기 컨벌루션 신경망의 입력값 중 단어의 상대적인 위치를 임베딩 벡터로 변환한 값은 신경망 학습과정에서 가변성을 가질 수 있다.

발명의 효과

[0053] 본 발명 컨벌루션 신경망 기반 영문 텍스트 정형화 방법은, 3개 이상의 속성을 가지는 데이터를 정형화 할 수 있는 효과가 있다.

[0054] 또한, 텍스트가 여러 개의 문장으로 구성되고, 타겟 키워드가 입력으로 주어지지 않는 경우에도 데이터를 효과적으로 정형화할 수 있는 효과가 있다.

도면의 간단한 설명

[0055] 도 1은 비정형 데이터를 정형 데이터로 처리하는 예시도이다.

도 2는 본 발명의 바람직한 실시예에 따른 컨벌루션 신경망 기반 영문 텍스트 정형화 방법의 순서도이다.

도 3은 도 2에서 전처리 단계의 예시도이다.

도 4는 본 발명에 적용되는 신경망 기반 정형화 알고리즘이다.

도 5는 키워드 c_i의 특징점 설명도이다.

도 6은 문장 S_i의 특징점 설명도이다.

발명을 실시하기 위한 구체적인 내용

[0056] 이하, 본 발명 컨벌루션 신경망 기반 영문 텍스트 정형화 방법에 대하여 첨부한 도면을 참조하여 상세히 설명한다.

[0057] 본 발명의 실시 예들은 당해 기술 분야에서 통상의 지식을 가진 자에게 본 발명을 더욱 완전하게 설명하기 위해 제공되는 것이며, 아래에 설명되는 실시 예들은 여러 가지 다른 형태로 변형될 수 있으며, 본 발명의 범위가 아래의 실시 예들로 한정되는 것은 아니다. 오히려, 이들 실시 예는 본 발명을 더욱 충실하고 완전하게 하며 당업

자에게 본 발명의 사상을 완전하게 전달하기 위하여 제공되는 것이다.

[0058] 본 명세서에서 사용된 용어는 특정 실시 예를 설명하기 위하여 사용되며, 본 발명을 제한하기 위한 것이 아니다. 본 명세서에서 사용된 바와 같이 단수 형태는 문맥상 다른 경우를 분명히 지적하는 것이 아니라면, 복수의 형태를 포함할 수 있다. 또한, 본 명세서에서 사용되는 경우 "포함한다(comprise)" 및/또는"포함하는(comprising)"은 언급한 형상들, 숫자, 단계, 동작, 부재, 요소 및/또는 이들 그룹의 존재를 특정하는 것이며, 하나 이상의 다른 형상, 숫자, 동작, 부재, 요소 및/또는 그룹들의 존재 또는 부가를 배제하는 것이 아니다. 본 명세서에서 사용된 바와 같이, 용어 "및/또는"은 해당 열거된 항목 중 어느 하나 및 하나 이상의 모든 조합을 포함한다.

[0059] 본 명세서에서 제1, 제2 등의 용어가 다양한 부재, 영역 및/또는 부위들을 설명하기 위하여 사용되지만, 이들 부재, 부품, 영역, 층들 및/또는 부위들은 이들 용어에 의해 한정되지 않음은 자명하다. 이들 용어는 특정 순서나 상하, 또는 우열을 의미하지 않으며, 하나의 부재, 영역 또는 부위를 다른 부재, 영역 또는 부위와 구별하기 위하여만 사용된다. 따라서, 이하 상술할 제1 부재, 영역 또는 부위는 본 발명의 가르침으로부터 벗어나지 않고서도 제2 부재, 영역 또는 부위를 지칭할 수 있다.

[0060] 이하, 본 발명의 실시 예들은 본 발명의 실시 예들을 개략적으로 도시하는 도면들을 참조하여 설명한다. 도면들에 있어서, 예를 들면, 제조 기술 및/또는 공차에 따라, 도시된 형상의 변형들이 예상될 수 있다. 따라서, 본 발명의 실시 예는 본 명세서에 도시된 영역의 특정 형상에 제한된 것으로 해석되어서는 아니 되며, 예를 들면 제조상 초래되는 형상의 변화를 포함하여야 한다.

[0061] 도 2는 본 발명의 바람직한 실시예에 따른 컨벌루션 신경망 기반 영문 텍스트 정형화 방법의 순서도이다.

[0062] 도 2를 참조하면 본 발명의 바람직한 실시예에 따른 컨벌루션 신경망 기반 영문 텍스트 정형화 방법은, 입력 텍스트를 전처리하여 입력 텍스트를 실수 벡터 형식으로 변환하는 단계(S10)와, 상기 입력 텍스트로부터 후보 키워드를 추출하는 단계(S20)와, 각 후보 키워드를 포함하는 문장을 색인하는 단계(S30)와, 상기 후보 키워드에 대한 특징점을 추출하는 단계(S40)와, 추출한 특징점을 입력 데이터로 하여 키워드의 라벨을 예측하는 단계(S50)와, 상기 산출된 키워드 속성을 네거티브 샘플링 기반 신경망 학습처리를 수행하여 각 속성에 키워드를 매핑하는 단계(S60)를 포함한다.

[0063] 이하, 상기와 같이 구성되는 본 발명의 바람직한 실시예에 따른 컨벌루션 신경망 기반 영문 텍스트로부터 관계형 데이터 베이스의 튜플을 추출하는 방법의 구성과 작용에 대하여 좀 더 상세히 설명한다.

[0064] 먼저, S10단계에서는 처리대상 데이터를 전처리한다.

[0065] 도 3은 상기 S10단계를 설명하기 위한 예시도이다.

[0066] 본 발명에서 사용하는 전처리 단계(S10단계)는 토큰화 단계(S11)와 토큰화된 데이터를 임베딩하는 단계(S12)를 포함한다.

[0067] 상기 토큰화 단계(S11)는 비정형 데이터인 영문 텍스트를 문장 단위로 분리한 뒤 문장 내의 각 단어를 토큰화 기술을 통해 식별하는 것이다.

[0068] 문장의 분리와 토큰화를 위하여 Stanford CoreNLP(Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, David McClosky: The Stanford CoreNLP Natural Language Processing Toolkit. ACL (System Demonstrations) 2014: 55-60)의 토큰 분석기(tokenizer)를 이용할 수 있다.

[0069] 토큰 이러한 과정을 통해, n_s개의 문장을 포함하는 입력 텍스트 $T=(S_1, .., Sn_s)$에서 m_i개의 단어를 포함하는 각 문장 $S_i=(w_{(i,1)},w_{(i,2)},..,w_{(i,mi)})$를 임베딩 벡터의 시퀀스 $E_i=(v_{(i,1)},\cdots,v_{(i,mi)})$로 변형할 수 있다.

[0070] 상기 토큰화 단계(S11)를 수행한 후 토큰화 된 데이터에 대하여 단어 임베딩 및 태깅 임베딩(품사 태깅 및 개체명 인식 과정)을 수행한다.

[0071] 단어 임베딩은 m개의 단어를 포함한 문장 $S=(w_1,w_2,..,w_m)$이 주어졌을 때, 각 단어 w_i는 실수 값을 가지는 d^{word}차원의 벡터 v_i로 변환된다. 이때, d^{word}x|V|차원의 임베딩 매트릭스(embedding matrix) W^{word}를 이용하는데, V는 어휘 목록이며, |V|는 어휘 목록의 크기로 고정된 값이다. W^{word}를 구하는 대표적인 기술로 word2vec(Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, Jeffrey Dean: Distributed Representations of

Words and Phrases and their Compositionality. NIPS 2013: 3111-3119), GloVe(Jeffrey Pennington, Richard Socher, Christopher D. Manning: Glove: Global Vectors for Word Representation. EMNLP 2014: 1532-1543)가 있다. 본 발명에서는 W^{word}로 구글에서 word2vec을 이용하여 학습시킨 임베딩 매트릭스를 이용한다.

[0072] 이때, d^{word}는 300개, |V|는 3,000,000개이다.

[0073] 문장 내의 각 단어 w_i를 v_i로 변환하는 과정은 먼저, w_i 를 V에서 색인한 후 w_i의 인덱스(index)에 해당하는 값을 원 핫 인코딩(one-hot encoding)한 V차원 벡터 o_i를 구하고 o_i와 W^{word}간의 매트릭스-벡터 곱 연산을 수행한다.

[0074] 이를 식으로 나타내면 다음의 수학식 1과 같다.

[0075] (수학식 1)

[0076] $$v_i = W^{word} o_i$$

[0077] 이때, w_i가 V에 존재하지 않는 단어일 경우, v i 은 d word 차원의 영벡터로 설정된다.

[0078] 그 다음, 태그 임베딩 과정에서는 상기 문장 S에 대해 품사 태깅(part-of-speech tagging)과 개체명 인식(named-entity recognition) 기술을 이용하여 문장 내 각 단어 w_i의 품사 태그(part-of-speech tag) 및 개체명 클래스(named entity class)를 구한다.

[0079] (w_i, pos_i, nec_i)의 순서쌍이 구해지면, pos_i와 nec_i를 각 d^{pos}, d^{nec}차원의 벡터 $vpos_i$, $vnec_i$로 변환한다. 이때, pos_i, nec_i를 임베딩하기 위해 d^{pos}x|V^{pos}| 차원의 임베딩 매트릭스 W^{pos}와 d^{nec}x|V^{nec}| 차원의 임베딩 매트릭스 W^{nec}를 각각 이용한다.

[0080] 상기 V^{pos}는 품사 태그의 목록이며 V^{nec}는 개체명 클래스의 목록이다.

[0081] 이때 품사 태그/개체명 클래스를 벡터로 변환하기 위한 임베딩 매트릭스(W^{pos}, W^{nec})를 구하는 대표적인 기술로 word2vec(Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, Jeffrey Dean: Distributed Representations of Words and Phrases and their Compositionality. NIPS 2013: 3111-3119), GloVe(Jeffrey Pennington, Richard Socher, Christopher D. Manning: Glove: Global Vectors for Word Representation. EMNLP 2014: 1532-1543)가 있다. 본 발명에서는 W^{pos}와 W^{nec}로 구글에서 word2vec을 이용하여 학습시킨 임베딩 매트릭스를 이용한다.

[0082] pos_i와 nec_i를 각 d^{pos}, d^{nec}차원의 벡터 $vpos_i$, $vnec_i$로 변환하기 위하여, pos_i와 nec_if를 d에서 색인한 후, pos_i와 nec_i의 인덱스에 해당하는 값을 원 핫 인코딩(one-hot encoding)한 d차원 벡터를 구하고, d차원 벡터와 W^{pos}, W^{nec} 각각과 매트릭스-벡터 곱 연산을 수행하여 변환할 수 있다.

[0083] 상기 W^{pos}와 W^{nec}는 학습 과정에서 갱신되는 파라미터이며, 초기값은 Xavier 초기화 방식(Xavier initialization)을 이용하여 설정된다.

[0084] Vpos와 Vnec이 각각 모든 품사 태그 및 개체명 클래스를 포함한 목록이므로 모든 pos와 nec은 Vpos와 Vnec에 반드시 포함된다.

[0085] 그 다음, S20단계에서는 영어 텍스트 데이터에서 후보 키워드를 추출한다.

[0086] 후보 키워드 추출 과정에서는 최소 1에서 최대 k_{max} 크기를 가지는 윈도우(window)를 이용하여 텍스트 T의 모든 문장 $S_1,\cdots,S_{ns}$에 포함된 키워드를 추출한다. 총 m_i개의 단어로 구성된 S_i내의 단어 $w_{(i,j)}$부터 이후 k번째 단어를 포함하는 시퀀스 ($w_{(i,j)}$, $w_{(i,j+1)},\cdots,$ $w_{(i,j+k-1)}$)를 $w_{(i,j:j+k-1)}$로 나타낸다고 할 때, i+j-1은 m_i보다 작거나 같다.

[0087] S_i로부터 크기가 k(1<=k<=min(k_{max}, m_i))인 윈도우를 이용하여 $w_{(i,1:k)}$, $w_{(i,2:k+1)}$,..., $w_{(i,mi-k+1:mi)}$의 총 m_i-k+1개의 시퀀스를 얻을 수 있다.

[0088] 다시 말하면, S_i로부터 1에서 k_{max} 사이의 크기 k의 윈도우를 이용하여 min(0, m_i-k+1) 개의 시퀀스를 얻을 수 있다. 위와 같은 방식으로, 텍스트 T의 각 문장 S_i에 대해 모든 시퀀스 $w_{(i,j:j+k-1)}$가 후보 키워드에 포함된다.

[0089] 주어진 텍스트 T에 대해서, 각 윈도우 크기에 대해 텍스트의 각 문장을 스캔하며 후보 키워드 집합 KC=[c_1, c_2, …,c_{nk}]를 계산한다. 동일한 시퀀스가 텍스트 내에 여러 번 등장할 수 있는데, KC는 [$w_{(1,1:1)}$, $w_{(1,2:2)}$, …, $w_{(ns,mns-kmax+1:mns)}$]에서 이러한 중복 시퀀스를 제거한 집합과 같다.

[0090] 도 4는 본 발명에 적용되는 신경망 기반 정형화 알고리즘이다.

[0091] 위에서 설명한 후보 키워드 집합 KC를 구하는 과정은 먼저, 비정형 텍스트 데이터 T와 데이터 스키마 SC를 입력받는다.

[0092] 도 4에서 INITIALIZESTRUCTURE(·)는 결과 정형 데이터를 저장할 d를 초기화하는 함수로써, d는 입력받은 SC를 데이터 스키마로 가지는 빈 튜플로 초기화 된다(1번째 줄).

[0093] EXTRACTKEYWORDS(·)는 비정형 텍스트 데이터 T로부터 후보 키워드 집합 KC를 생성하는 함수이다(2번째 줄).

[0094] 상기 후보 키워드 집합 KC를 구하는 함수에 관해서는 앞서 상세히 설명한 바와 같다.

[0095] 상기 정형화 알고리즘에서 확인할 수 있는 바와 같이 본 발명은 컴퓨터로 수행되는 알고리즘의 해석이며, 수행 주체는 컴퓨터 또는 컴퓨터에 준하는 연산장치를 사용할 수 있다.

[0096] 그 다음, S30단계와 같이 후보 키워드를 포함하는 문장을 색인한다.

[0097] 상기와 같이 후보 키워드 집합 KC를 구한 후, 각 키워드 c_i를 포함하는 모든 문장의 집합 SC_i을 텍스트 T 내에서 추출하는 과정을 수행한다.

[0098] 문장 색인 과정은 KC를 구하는 과정에서 각 키워드 c_i에 대해, c_i를 포함하는 문장의 인덱스 정보를 함께 저장함으로써 효율적으로 수행될 수 있다.

[0099] 저장한 인덱스 정보를 이용하여 다음의 순서쌍 (KC,SC)=[(c_1, SC_1), …, (c_{nk}, SC_{nk})]를 구한다.

[0100] 이와 같은 순서쌍을 구하는 과정은 도 4의 알고리즘에서 4번째 줄에 기재되어 있으며, 앞서 구해진 두 함수를 호출하여 d의 초기화 및 KC를 생성한 다음, KC에 속한 각 키워드 c에 대해 c를 포함하는 T 내의 모든 문장을 찾는다.

[0101] 그 다음, S40단계와 같이 구해진 (KC, SC) 쌍과 각 문장 S_i의 임베딩 결과인 EP_i를 이용하여, 각 키워드 c_i의 특징점을 추출한다.

[0102] 앞서 하나의 키워드에 대해 여러 개의 문장이 SC_i에 포함될 수 있음을 설명하였다. 특징점 추출과정에서는, SC_i에 포함되는 각 문장 $S_{(ci,j)}$에 대해 (c_i, $S_{(ci,j)}$)의 특징점을 추출하여 신경망 모형의 입력값으로 이용한다.

[0103] 즉, 하나의 키워드 c_i는 c_i를 포함한 문장 개수인 |SC_i| 횟수만큼 신경망 모형에 입력되며, 이후 추론 과정에서 |SC_i| 개의 신경망 출력 값 중에 가장 적합한 출력값을 선택한다.

[0104] 특징점은 크게 키워드 c_i의 특징점과 문장 $S_{(ci,j)}$ 의 특징점으로 나뉜다. 먼저, c_i의 특징점은 1) c_i의 단어 임베딩 벡터, 2) c_i의 품사 태그 임베딩 벡터, 3) c_i의 개체명 클래스의 임베딩 벡터, 4) $S_{(ci,j)}$ 내에서 c_i의 직전 위치에 있는 단어의 품사 태그 임베딩 벡터 및 5) 개체명 클래스 임베딩 벡터, 6) c_i의 직후 위치의 단어의 품사 태그 임베딩 벡터 및 7) 개체명 클래스 임베딩 벡터로 총 7가지가 있다.

[0105] 각 임베딩 벡터는 단일 단어임을 가정하는 반면 c_i는 여러 개의 단어로 구성되어 있다. 따라서, 1) - 3)의 경우, 여러 단어로 된 키워드의 임베딩 벡터를 계산하는 방법을 추가로 고려해야 한다.

[0106] 본 발명에서는 어텐션(attention)을 적용하여 여러 단어로 된 키워드의 임베딩 벡터를 구하였다. $S_{(ci,j)}$의 특징점은 컨벌루션 신경망에 $S_{(ci,j)}$ 에 대응되는 단어 임베딩 벡터의 시퀀스 $E_{(ci,j)}$와 $S_{(ci,j)}$ 내 c_i와 각 단어의 상대적 위치 값을 임베딩한 상대 위치 임베딩 벡터 vposition 의 시퀀스 $Epos_{(ci,j)}$를 연결한 $EC_{(ci,j)}=[(v_{(k,1)}, vposition_{(k,1)}), (v_{(k,2)}, vposition_{(k,2)}), \cdots, (v_{(k,mk)}, vposition_{(k,mk)})]$를 통과시켜 추출한다.

[0107] 상기의 과정은 도 4의 알고리즘의 7번째 및 8번째 줄에 기재되어 있다. 특징점은 문장의 특징점과 키워드 c의 특징점으로 나뉘는데, 문장 S의 특징점은 컨벌루션 신경망을 이용하여 추출하고(7번째 줄), c의 특징점은 GETFEATURE(·) 함수를 통해 추출한다(8번째 줄).

[0108] 도 5와 도 6은 각각 'Database Administrator'와 이를 포함한 문장 'We are seeking a Senior Database Administrator.'을 예시로 특징점을 추출하는 과정의 설명도로서, 도 5는 키워드 c_i의 특징점 설명도이고, 도 6은 문장 S_i의 특징점 설명도이다.

[0109] 도 5와 도 6을 각각 참조하면, 타겟 키워드는 'Database', 'Administrator' 두 개의 단어로 이루어져 있다.

[0110] 먼저, 두 단어의 단어 임베딩, 태그 임베딩 벡터를 구한다. 그런 다음 두 벡터에 어텐션 기법을 적용하여 'Database Administrator'의 임베딩 벡터를 계산한다.

[0111] 그 결과, 키워드가 포함하는 단어 개수에 관계없이 키워드의 임베딩 벡터의 크기는 일정하게 유지된다.

[0112] 문장에서 'Database Administrator' 직전단어는 'Senior'이고, 직후 단어는 '.'이므로 두 단어의 품사 태그 임베딩 벡터와 개체명 클래스 임베딩 벡터를 구한다.

[0113] 한편, 도 6은 문장의 특징점을 구하는 과정을 보여준다.

[0114] 먼저, 각 문장의 단어를 단어 임베딩 과정을 통해 벡터로 변환하고, 문장 내 각 단어의 타겟 키워드와의 상대위치를 임베딩한 벡터를 구한다.

[0115] 도 6에 도시한 바와 같이 한 문장이 하나의 매트릭스로 표현되는데, 이 매트릭스를 컨벌루션 신경망에 입력하여, 문장의 특징점을 구한다.

[0116] 상기 컨벌루션 신경망은, 필터(filter) 3개를 사용한 단일 레이어 컨벌루션 신경망을 가정하였으나, 본 발명은 이러한 컨벌루션 신경망 구조에 의해 제한되는 것은 아니다.

[0117] 어텐션을 적용하여 c_i의 임베딩 벡터를 계산하는 과정에서 어텐션 벡터 v^{att} 가 이용된다. 이때, v^{att}는 $(d^{word}+d^{pos}+d^{nec})$ 차원의 벡터로 학습 과정에서 갱신되는 파라미터이며, Xavier 초기화 방식을 이용하여 초기값이 설정된다.

[0118] 키워드 c_i가 k개의 단어 $(w_1, w_2,..,w_k)$로 구성되어 있다고 하고, 단어 w_j의 임베딩 벡터는 $vp_j=(v_j, vpos_j, vnec_j)$라고 할 때, w_j의 어텐션 $aj = \frac{vp_j^T \cdot v^{att}}{\sum_{l=1}^{k} vp_l^T \cdot v^{att}}$ 이 된다. 어텐션을 적용한 c_i의 임베딩 벡터는 $\sum_{j=1}^{k} a_j \cdot vp_j$ 가 된다.

[0119] $S_{(ci,j)}$ 의 특징점을 추출하기 위해 이용되는 컨벌루션 신경망은 1개의 컨벌루션 레이어(convolution layer), 1개의 맥스 폴링 레이어(max-pooling layer)로 구성될 수 있다. 컨벌루션 레이어에서는 크기가 w(w=3)인 윈도우를 이용하여 문장 내 연속된 w개의 단어들의 국소 특징점(local feature)을 추출한다.

[0120] 단어 시퀀스 $w_{k:k+w-1}$의 특징점을 $conv_k$라고 할 때, $conv_k=EC_{(ci,j)}[k:k+w-1]^T \times W^{conv}$으로 나타내 질 수 있다. 여기서 W^{conv}은 $EC_{(ci,j)} \cdot w$ 차원의 벡터로 학습 과정에서 갱신되는 파라미터이고, $EC_{(ci,j)}[k]$을 $EC_{(ci,j)}$의 k번째 벡터값이라고 할 때, $EC_{(ci,j)}[k:k+w-1]$은 $EC_{(ci,j)}[k]$, $EC_{(ci,j)}[k+1],...,EC_{(ci,j)}[k+w-1]$를 모두 이은 벡터이다.

[0121] 이렇게 얻어진 $conv_k$는 스칼라 값이며, 길이가 $m_{(ci,j)}$인 문장의 경우 $m_{(ci,j)-w}+1$ 개의 서로 다른 윈도우로 부터 스

칼라 값을 얻을 수 있다.

[0122] 즉, $m_{(ci,j)-w}+1$ 차원의 벡터 conv가 계산된다. 본 발명에서는 제로 패딩(zero padding) 기술을 적용하여 $m_{(ci,j)}$ 차원의 벡터 conv를 계산하였다.

[0123] 맥스 풀링 레이어에서는 벡터 conv 의 각 필터마다 가장 큰 값을 추출하면 $S_{(ci,j)}$의 특징점이 된다. 본 발명에서는 특징점의 다양성을 보장하기 위하여 d^{conv}개의 서로 다른 필터를 적용하였고, 이에 따라 최종적으로 d^{conv}차원의 특징점 벡터를 추출하였다. 이때, d^{conv}는 정수 값을 갖는 하이퍼 파라미터이다.

[0124] 컨벌루션 신경망의 입력값으로 이용되는 상대 위치 임베딩 벡터는 상대 위치 임베딩 매트릭스를 이용하여 계산한다. 먼저, 상대 위치는 키워드 c_i에 속하는 단어의 경우 모두 0이고, 문장 내 키워드 $c_i=[w_1, \cdots, w_k]$에서 w_1보다 좌측에 위치한 단어는 w_1에서 멀어질수록 상대 위치 값이 1씩 감소하고, w_k보다 우측에 위치한 단어는 w_k에서 멀어질수록 1씩 증가하도록 정의된다.

[0125] 도 6의 예시에서는 이러한 방식으로 설정된 상대 위치 값을 보여주고 있다. ‘We’ 는 가장 좌측에 있는 단어로, ‘Database’ 로부터의 상대 위치가 -5이다. ‘Administrator’ 의 우측에 위치한 단어는 ‘.’ 로, ‘.’ 의 상대 위치는 1이 된다.

[0126] 상대 위치 값은 품사 태그, 개체명 클래스와 같은 방법으로 $d^{position}$차원의 벡터로 임베딩된다. 이 과정에서 사용되는 $d^{position} \times |V^{position}|$ 차원의 상대 위치 임베딩 매트릭스 $W^{position}$은 학습 과정에서 갱신되는 파라미터이다.

[0127] 그 다음, S50단계에서는 키워드 특징점을 이용하여 키워드 c_i의 라벨을 예측한다.

[0128] 본 발명에서는 키워드 c_i의 라벨을 예측하기 위하여 소프트맥스 회귀(Softmax regression)를 이용한다. 상기 추출한 특징점은 소프트맥스 회귀의 입력 값이 된다. N항 관계의 데이터 스키마를 $(r_1, r_2, \cdots, r_N)$으로 나타낼 때, 클래스의 이산 집합 Y는 총 N+1개의 클래스 $[r_1, r_2, \cdots, r_N, NA]$로 구성된다.

[0129] 이때, 키워드 c_i와 키워드 c_i를 포함하는 문장 $S_{(i,j)}$로부터 추출한 특징점 f를 이용하여 최종적으로 계산한 점수 $score(c_i, S_{(i,j)})=Softmax(W^{(S)}f)$이다. 이때 $W^{(S)}$ 는 $(N+1)\times(d^{word}+3d^{pos}+3^{dnec}+d^{conv})$ 차원의 소프트맥스 매트릭스이다.

[0130] $W^{(S)}r$을 $W^{(S)}$의 r번째 행 벡터라고 했을 때, 키워드 c_i의 최종 클래스 $r(c_i)= argmax_r(score(W^{(S)}{}_rf))$가 된다.

[0131] 이와 같은 과정은 도 4의 알고리즘의 9번째 줄과 같이 추출한 특징점을 소프트맥스 회귀(Softmax regression)의 입력 값으로 하여, 키워드 c가 속하는 속성과 점수를 계산하는 과정이며, 6번째 줄 내지 11번째 줄과 같이 키워드 c를 포함하는 모든 문장 S에 대해서 특징점 추출 및 소프트맥스 회귀 과정을 반복한다.

[0132] 이때 문장에 따라 키워드 c의 속성과 점수가 달라질 수 있다.

[0133] 그 다음, S60단계에서는 신경망 학습을 통해 매핑한다.

[0134] 신경망 학습의 학습 데이터로는 텍스트 데이터와 텍스트 데이터에 대응되는 튜플을 이용한다. 이때, 튜플에 저장된 각 엔티티는 텍스트 데이터에 등장하는 키워드여야 한다.

[0135] 튜플의 각 엔티티 값과 동일한 텍스트 데이터 내의 키워드에 대해서, 해당 키워드의 라벨을 엔티티의 속성으로 설정한다. 이외의 키워드는 라벨을 NA로 설정한다.

[0136] 신경망 학습 과정에서 윈도우를 이용하여 구한 모든 후보 키워드를 학습 데이터로 이용할 경우, 대부분의 후보 키워드의 라벨이 NA인 문제가 발생할 수 있다. 본 발명에서는 이러한 문제를 해결하기 위해 네거티브 샘플링 기법을 학습 과정에 적용하였다.

[0137] 후보 키워드 중 라벨이 NA가 아닌 키워드를 양성데이터(positive data)로 정의하고, 라벨이 NA인 키워드는 음성데이터(negative data)로 정의한다.

[0138] 양성데이터인 키워드는 모두 학습 데이터로 이용하고, 음성 데이터인 키워드 개수는 양성 데이터와의 비율이

Po:Ne을 만족하도록 균일 분포(uniform distribution)를 따라 전체 음성 데이터에서 무작위 추출한다. 이때, Po와 Ne는 실수 값을 갖는 하이퍼 파라미터이다.

[0139] 교차 엔트로피 오차 함수(cross-entropy cost function)를 기반으로 하는 본 발명의 학습 비용 함수는 Θ를 갱신 가능한 모든 파라미터라고 할 때 다음의 수학식 2와 같다.

[0140] (수학식 2)

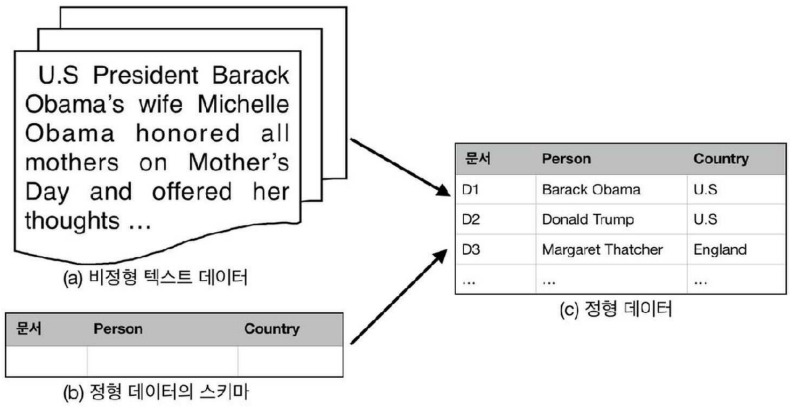
$$J(\theta) = \frac{1}{nk}\sum_{i=1}^{nk} log\, p(r_i|c_i, S_{(ci,j)};\ \theta)$$

[0141]

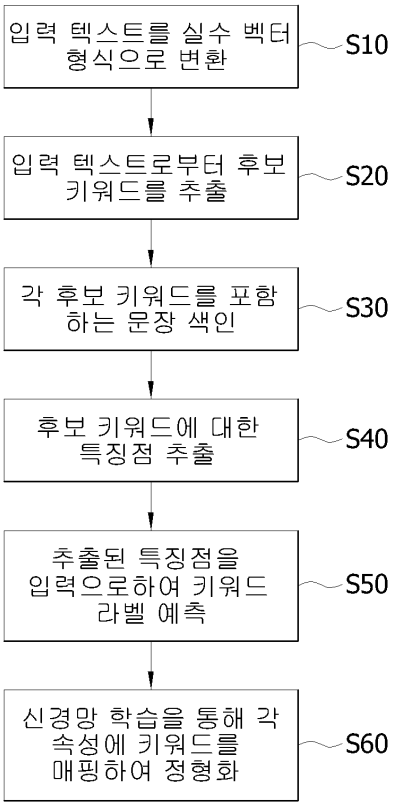
[0142] 본 발명은 상기 실시예에 한정되지 않고 본 발명의 기술적 요지를 벗어나지 아니하는 범위 내에서 다양하게 수정, 변형되어 실시될 수 있음은 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 있어서 자명한 것이다.

도면

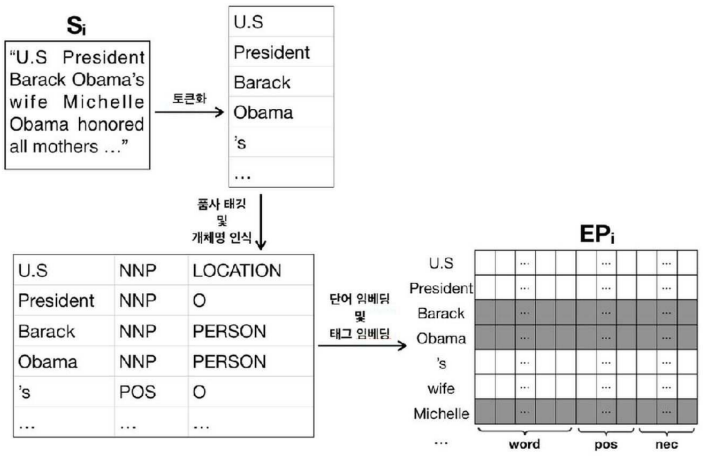
도면1



도면2


입력 텍스트를 실수 벡터 형식으로 변환
S10
입력 텍스트로부터 후보 키워드를 추출
S20
각 후보 키워드를 포함하는 문장 색인
S30
후보 키워드에 대한 특징점 추출
S40
추출된 특징점을 입력으로하여 키워드 라벨 예측
S50
신경망 학습을 통해 각 속성에 키워드를 매핑하여 정형화
S60

도면3


Si
"U.S President Barack Obama's wife Michelle Obama honored all mothers ..."
토큰화
U.S
President
Barack
Obama
's
...
품사 태깅 및 개체명 인식
U.S NNP LOCATION
President NNP O
Barack NNP PERSON
Obama NNP PERSON
's POS O
... ...
단어 임베딩 및 태그 임베딩
EPi
U.S
President
Barack
Obama
's
wife
Michelle
...
word
pos
nec

도면4

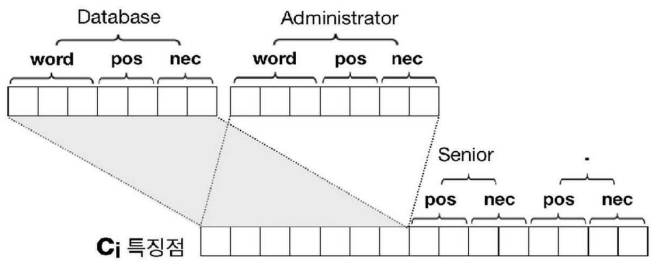
Algorithm 1 RelationExtraction(t, sc)

Input: unstructured textual data t, a data schema sc

Output: structured data d of t

```
d := INITIALIZESTRUCTURE(SC);
kC := EXTRACTKEYWORDS(t);                    //kC : Candidates of keywords
for each c in k_c do
    s_c := GETSENTENCES(t, c);               // s_c : Sentences which contain a candidate keyword c
    score_max := 0; r := NA;
    for each s in s_c do
        cv_s := CONVNN(s);                   // cv_s : Convolution output of a sentence s
        f_(s,c) := GETFEATURE(s, c);         // f_(s,c) : Features of c in s
        (r_s, score_s) := SOFTMAXREGRESSION(sc, (f_(s,c)), cv_s));
        if r_s != NA and score_max < score_s then
            (r, score_max) := (r_s, score_s);
    if r != NA and score_max > GETTHRESHOULD(r) then
        d := ADDKEYWORD(r, c);
```

도면5



도면6