Smart Media Journal / Vol.8, No.2 / ISSN:2287-1322
http://dx.doi.org/10.30693/SMJ.2019.8.2.74

음절 단위 임베딩과 딥러닝 기법을 이용한 복합명사 분해

(Compound Noun Decomposition by using Syllable-based Embedding and Deep Learning)

이현영*, 강승식**
(Hyun Young Lee, Seung Shik Kang)

요 약

기존의 복합명사 분해 알고리즘은 미등록어 단위명사들이 포함된 복합명사를 분해할 때 미등록어를 분리하기 어려운 문제가 발생한다. 이는 현실적으로 모든 고유명사, 신조어, 외래어 등의 모든 단위 명사를 사전에 등록하는 것은 불가능하다는 한계가 존재하기 때문이다. 이 문제를 해결하기 위하여 복합명사 분해 문제를 태그 열 부착(sequence labeling) 문제로 정의하고 음절 단위 임베딩과 딥러닝 기법을 이용하는 복합명사 분해 방법을 제안한다. 단위명사 사전을 구축하지 않고 미등록 단위명사를 인식하기 위하여 복합명사를 구성하는 각 음절들을 연속적인 벡터 공간에 표현하여 LSTM과 선형체인(linear-chain) CRF를 이용하는 방식으로 복합명사를 단위명사들로 분해한다.

■ 중심어 : 복합명사 분해 ; bigram ; 음절 임베딩 ; LSTM ; 선형체인 CRF

Abstract

Traditional compound noun decomposition algorithms often face challenges of decomposing compound nouns into separated nouns when unregistered unit noun is included. It is very difficult for those traditional approach to handle such issues because it is impossible to register all existing unit nouns into the dictionary such as proper nouns, coined words, and foreign words in advance. In this paper, in order to solve this problem, compound noun decomposition problem is defined as tag sequence labeling problem and compound noun decomposition method to use syllable unit embedding and deep learning technique is proposed. To recognize unregistered unit nouns without constructing unit noun dictionary, compound nouns are decomposed into unit nouns by using LSTM and linear-chain CRF expressing each syllable that constitutes a compound noun in the continuous vector space.

■ keywords : compound noun decomposition ; bigram ; syllable embedding ; LSTM ; Linear-Chain CRF

Ⅰ. 서 론

한국어 문장에서 명사는 중요한 의미 정보를 갖는 성분으로 검색엔진에서 색인어 추출, 질의어 분석 등에 사용되므로 자동 띄어쓰기, 오타교정 및 복합명사 분해를 통해 명사 성분을 추출한다[1,2,3]. 하지만 둘 이상의 명사들이 결합된 복합명사의 경우 띄어쓰기가 자유롭기 때문에 중의적인 표현으로 인해 색인어와 질의어 간의 용어 불일치가 발생하고 이는 검색 성능을 저하시키는 요인이 되기도 한다[4,5]. 이러한 문제점을 해결하기 위해 복합명사 분해에 대한 기존의 연구는 단위명사 사전을 구축한 후 복합명사를 단위명사 형태로 분해하는 방식으로 연구를 진행하였다[6,7,8,9]. 하지만 사전에 수록해야 하는 복합명사 개수가 너무 많기 때문에 현실적으로 모든 복합명사를 사전으로 관리하는데 어려움이 있다.

복합명사의 분해는 분해 기준에 따라 중의성이 발생하여 이는 복합명사의 구문적 불일치 문제가 발생한다. 예를 들어, “국어정보처리”라는 복합명사를 검색할 때 질의어 처리 과정에서 “국어 정보처리”, “국어정보 처리”, “국어 정보 처리” 또는 “한미동맹”의 경우에는 “한 미 동맹”, “한미 동맹”과 같이 다양한 형태로 분해가 가능하다. 이처럼 복합명사를 단위명사로 분해하는 방식은 다양하고 사전 기반의 복합명사 분해 방식은 고유명사, 외래어, 신조어 등 사전에 등록되지 않은 미등록 단위명사를 처리해야 하는 어려운 점이 있다[6]. 한국어의 단위명사간의 결합은 자유로워 복합명사는 수는 다양하고 이에 따라 단위명사 사전의 용량은 커지게 된다. 그리하여 명사 기반의 어절 단위에

* 학생회원, 국민대학교 컴퓨터공학과
** 정회원, 국민대학교 소프트웨어학부
이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No.NRF-2017M3C4A7068186).
접수일자 : 2018년 10월 10알
수정일자 : 1차 2019년 01월 17일, 2차 2019년 02월 14일
게재확정일 : 2019년 02월 17일
교신저자 : 강승식 e-mail : sskang@kookmin.ac.kr

서 복합명사를 분해하지 않고, 음절 단위에서 복합명사를 분해하는 방식으로 공백문자가 없는 복합명사를 단위명사로 분해하는 시스템을 제안한다.

Ⅱ. 관련 연구

한국어의 복합명사 분해에 관한 연구는 규칙기반 방식과 말뭉치 기반의 확률 및 통계적 방식으로 나누어진다. 규칙기반 방식은 복합명사의 음절 길이에 따른 선호 음절 패턴 규칙을 이용하는 방식으로 이현민(2001)은 단위명사 사전, 접사 사전, 한국어 복합명사의 구조적 특성을 기반으로 복합명사를 분해한다[6]. 강승식(1998)은 형태소 분석 결과로 추정된 4음절 복합명사의 2+2, 5음절 복합명사의 3+2, 2+3 등의 음절 패턴 유형과 두 가지 예외 규칙을 사용하여 복합명사를 분해하는 방법을 제안하였다[7]. 이러한 규칙 기반의 방식은 음절 길이의 제한이 없이 결합이 가능한 모든 복합명사를 대상으로 하기에는 어려움이 있다[8].

확률 기반 방법은 빈도수를 이용하는 방식으로 이용훈(2012)은 복합명사를 구성하는 단위명사의 위치별 명사 빈도 데이터를 이용하여 복합명사를 분해한다[8]. 심광섭(1997)은 이웃하는 두 개 음절간의 합성 상호 정보(composite mutual information)를 이용하여 띄어쓰기가 전혀 되어 있지 않은 복합명사를 단위명사로 분해하는 알고리즘을 제시하였다[9]. 이용훈(2012)은 세종 말뭉치에서 추출한 명사 사전, 심광섭(1997)은 2음절 기반 명사 사전 등의 빈도수 기반의 확률 및 통계적 정보를 이용하여 복합명사를 단위 명사로 분해한다[8,9]. 통계적 방식에서도 단위명사 사전을 이용하여 복합명사를 분해하기 때문에 미등록어 단어 처리해야하는 문제가 발생한다. 기존의 단위명사 사전을 이용하는 연구에서는 복합명사 분해의 성능은 사전에 의존하므로 미등록어가 사전에 적을수록 복합명사 분해 성능이 향상함을 보이고 있다[6,7,8,9].

본 논문에서는 단위명사 사전을 이용하지 않고 음절을 연속적인 벡터 공간에 표현하고 음절 unigram과 음절 bigram 벡터를 이용하여 미등록어 처리에 유연하고, 순차적인 데이터의 길이에 유연한 딥러닝 기법인 단방향 LSTM 또는 양방향 LSTM 이용하여 복합명사를 분해하는 두 가지 방법을 제안한다.

Ⅲ. 음절 단위 임베딩과 딥러닝 기법을 이용한 복합명사 분해

1. 복합명사 분해 태그(BI) 열 부착

기존의 공백 삽입 형태의 복합명사 분해 문제는 BI 태그 열 부착 문제로 정의할 수 있다. 복합명사의 각 음절들에 B(beginning)와 I(inside) 태그를 부착한 후에 음절 사이에 공백이 없는 복합명사를 단위명사들로 분해하는 방법이다. 복합명사 분해 태그는 BI 태그를 사용하여 복합명사를 구성하는 단위명사의 시작 음절은 B 태그를 부착하고, I 태그는 단위명사의 시작음절 이외의 음절에 태깅(tagging)을 한다. BI 태그 부착이 완료된 후에는 그림 1과 같이 B 태그 앞에 공백을 삽입하여 복합명사를 단위명사로 분해한다.

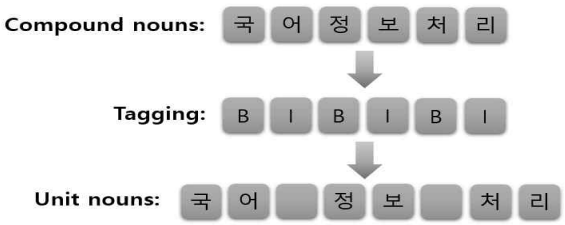

그림 1. BI 태그에 의한 복합명사 분해

2. 음절 사전

단위명사가 등록된 사전을 기반으로 복합명사를 단위명사로 분해하는 기존의 복합명사 분해 알고리즘들은 모든 단위명사를 사전에 등록하는 것이 불가능하므로 등록되지 않은 미등록 단위명사를 포함한 복합명사의 분해 문제가 발생한다. 복합명사는 두 개 이상의 명사들이 결합되지만 명사를 구성하는 성분은 음절단위이고, 명사를 구성하는 음절 종류는 한정적이다. 따라서 사전에 미등록 단위명사를 구성하는 음절이 사전에 등록된 단위명사를 구성하는 음절로 사용되는 경우가 단위명사 자체로 사전에 존재하는 확률보다 높다. 본 논문에서는 복합명사들에 대해 음절 unigram과 bigram 형태의 음절 사전을 구축하고 음절 임베딩과 신경망(neural network)을 이용하여 복합명사를 단위명사로 분해한다.

음절 사전을 구축할 때 복합명사를 구성하는 마지막 음절 표시를 위하여 마지막 음절과 “<WORD_END>” 토큰을 사용하여 음절 bigram을 구축한다. 예를 들어, “국어정보처리”라는 복합명사를 사전에 등록을 한다면 unigram은 “국”, “어”, “정”, “보”, “처”, “리”이고, bigram은 “국어”, “어정”, “정보”, “보처”, “처리”, “리<WORD_END>”라는 형태로 등록한다. 그리고 예외 처리를 위해 사전에 없는 음절을 벡터 공간에 표현하기 위해 “<UNK>” 토큰을 사용하였다.

3. 음절 임베딩과 LSTM를 이용한 인코딩

자연어 처리를 위한 딥러닝 모델은 단어의 의미를 이해하고 표현하기 위해서 단어를 연속적인 벡터 공간에 표현하는 벡터

공간 모델을 사용한다[1,11,12,13]. 단어 벡터를 사용하는 모델은 말뭉치에 존재하는 단어를 연속적인 벡터 공간에 표현하기 때문에 말뭉치에 존재하지 않는 단어를 처리해야하는 문제가 발생한다. 그리하여 단어를 벡터 공간에 표현하기보다는 음절 사전에 등록된 음절 unigram과 bigram을 연속적인 벡터 공간에 표현하였다. 그리고 복합명사를 구성하는 음절들의 의존정보를 고정된 길이의 새로운 벡터로 표현하기 위해 그림 2와 그림 3과 같이 음절 벡터를 단방향 LSTM 또는 양방향 LSTM으로 인코딩하는 방법을 두 가지 모델로 각각 구성하여 복합명사를 구성하는 음절에 복합명사 분해 태그를 부착한다.

그림 2와 같이 단방향 LSTM은 입력이 왼쪽에서 오른쪽으로 순차적으로 입력되어 현재 입력은 과거의 입력과의 의존정보를 고려하여 새로운 자질 벡터를 생성한다. 그림 3과 같이 양방향 LSTM은 단방향 LSTM과 달리 현재 입력에 과거 입력과 미래 입력들의 의존정보를 고려하여 새로운 자질 벡터를 생성한다. 단방향 LSTM을 이용하여 음절 unigram 벡터와 음절 bigram 벡터를 인코딩 할 때에는, 그림 2와 같이 음절 unigram 벡터와 음절 bigram 벡터에 각각 하나의 단방향 LSTM으로 인코딩하고 음절 unigram과 음절 bigram의 왼쪽에서부터 오른쪽으로 대응되는 순서로 단방향 LSTM의 출력 값을 결합하여 새로운 자질 벡터를 생성하였다. 양방향 LSTM을 이용하는 경우에도 그림 3과 같이 음절 unigram 벡터와 음절 bigram 벡터에 각각 하나의 양방향 LSTM으로 인코딩하고 음절 unigram과 음절 bigram의 왼쪽에서부터 오른쪽으로 대응되는 순서로 양방향 LSTM의 출력 값을 더하거나 이어붙이는 형태로 새로운 자질 벡터를 생성하였다. 단방향 LSTM 또는 양방향 LSTM을 통한 음절 인코딩 후에는 두 모델은 복합명사를 구성하는 음절에 태그 부착을 위해 선형체인 CRF를 사용한다.

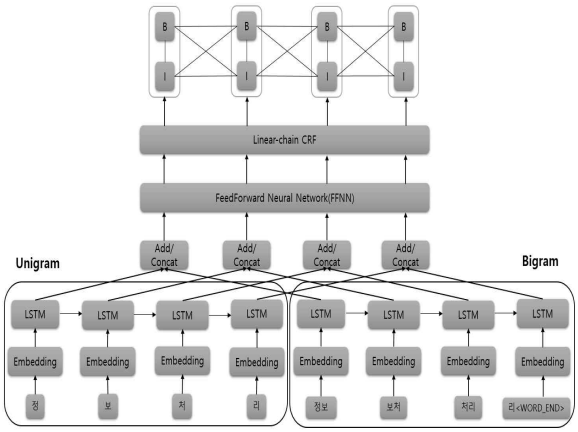

그림 2. 단방향 LSTM과 선형체인 CRF를 이용한 복합명사 분해

선형체인 CRF는 순차적인 입력 열의 태그 점수와 각 열과 이웃하는 태그의 의존성을 함께 고려하여 최적의 태그 열을 예측한다[10]. 본 논문에서 그림 2와 그림 3과 같이 단방향 LSTM과 양방향 LSTM을 통해 인코딩 된 자질 벡터를 전방향 신경망(feedforward neural network)의 입력으로 하여 복합명사의 각 음절에 해당하는 태그 클래스(B 또는 I) 점수를 계산하고 이를 바탕으로 선형체인 CRF를 통해 최적의 태그 열을 예측하여 복합명사를 단위명사들로 분해한다.

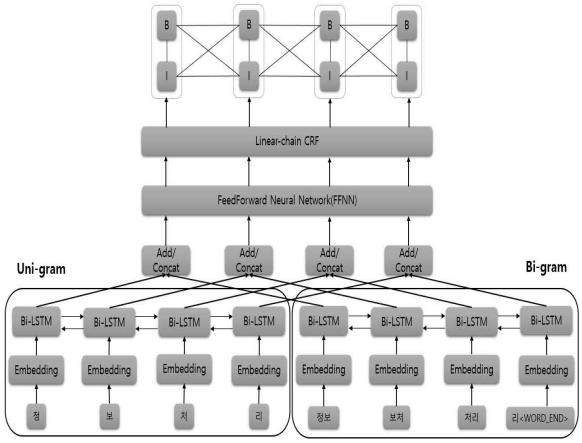

그림 3. 양방향 LSTM과 선형체인 CRF를 이용한 복합명사 분해

Ⅳ. 실험 및 평가

복합명사 분해 실험 및 평가를 위한 말뭉치 데이터는 “차세정 언어처리 경진대회 2018”의 복합명사 분해 태스크(task)에서 제공하는 복합명사 말뭉치를 사용하였다. 이 말뭉치는 표준국어대사전 복합명사 리스트, 우리말샘 복합명사 리스트, 세종 형태소 말뭉치에서 추출한 복합명사 리스트, 국립국어원의 “한국어 복합명사 용례 분석 프로그램”에서 추출한 복합명사 리스트로 표 1과 같이 구성되어 있다.

표 1. 복합명사 분해 학습 및 평가 데이터

	Training	Test	Total
라인수	2,600,738	288,971	2,889,709
단어 (중복 단어 포함)	5,562,304	617,825	6,180,129
단어 (중복 단어 미포함)	237,005	83,527	320,532
음절 (중복 음절 포함)	11,140,829	1,237,687	12,378,516
음절 (중복 음절 미포함)	2,489	1,904	4,393

복합명사 데이터 집합은 표 1과 같이 한 라인에 하나의 복

합명사로 구성되어 총 2,889,709개의 복합명사이며, 복합명사를 구성하는 단위명사는 320,532개(중복 단위명사 미포함), 음절은 총 4,393개(중복 음절 미포함)로 구성되어 있다. 복합명사 분해 학습 및 평가를 위해 2,889,709개의 복합명사를 2,600,738개의 학습 데이터와 288,971개의 테스트 데이터로 나누어 학습 및 평가를 수행하였다.

전방향 신경망은 복합명사의 각 음절의 태그 클래스의 점수 계산을 위해 비선형 함수를 사용하지 않고, 한 개의 출력층(output layer)만 사용하였다. 테스트용 학습 데이터 집합에는 포함되지만 테스트 데이터 집합에는 포함되지 않는 음절 벡터의 경우에는 무작위로 초기화한 벡터를 사용하였다. 복합명사를 단위명사로 분해하기 위해 사용한 단방향 LSTM과 선형체인 CRF 또는 양방향 LSTM과 선형체인 CRF를 이용하는 모델은 텐서플로우*로 구현하였다.

표 2는 학습 및 평가를 위한 음절 임베딩, LSTM의 출력 연산의 종류, 음절 임베딩 크기, LSTM 셀 유닛 크기 등을 다르게 구성한 모델의 종류를 나타내고, 각 모델들은 미니 배치 확률적 경사 하강법(mini batch stochastic gradient descent)로 학습하였다. 표 2의 모델 종류 이외의 하이퍼 파라미터(hyper parameter)의 경우에는 학습 횟수(epoch)는 5씩 증가하여 5, 10, 15, 20, 학습률(learning rate)는 0.001, 배치 사이즈(batch size)는 10, 20으로, 음절 벡터의 인코딩을 위한 단방향 LSTM 또는 양방향 LSTM 층수는 1로 모델을 학습하였다. 단방향 LSTM과 음절 unigram를 사용하는 모델의 경우에는 단방향 LSTM의 출력 결과를 그대로 전방향 신경망의 입력으로 하여 모델을 학습하였다.

표 2. 복합명사 분해 모델 종류

모델	LSTM	음절 임베딩	LSTM 출력연산 종류	임베딩 크기 및 LSTM 셀 유닛 크기
1	단방향	Unigram	–	250
2				300
3		Unigram + Bigram	Add	250
4				300
5			Concatenation	250
6				300
7	양방향	Unigram	Add	250
8				300
9			Concatenation	250
10				300
11		Unigram + Bigram	Add	250
12				300
13			Concatenation	250
14				300

* Tensorflow, Available: https://www.tensorflow.org/

모델의 정량적 성능 평가를 위해 식 (1) ~ (5)와 같이 복합명사 분해 태그 정확도(accuracy), 어절 재현율(word recall), 어절 정확도(word precision)와 공백 재현율(spacing recall), 공백 정확도(spacing precision)를 사용하였다.

$$Accuracy = \frac{The\,\mathrm{Predicted}\,Correct\ Tags}{The\ Actual\ Entire\ Tags} \times 100 \quad (1)$$

$$Word\,Recall = \frac{The\,\mathrm{Predicted}\,Correct\ Words}{The\ Actual\ Entire\ Words} \times 100 \quad (2)$$

$$Word\,\mathrm{Precision} = \frac{The\,\mathrm{Predicted}\,Correct\ Words}{The\,\mathrm{Predicted}\,Entire\ Words} \times 100 \quad (3)$$

$$Spacing\,Recall = \frac{The\,\mathrm{Predicted}\,Correct\ Spacing}{The\ Actual\ Entire\ Spacing} \times 100 \quad (4)$$

$$Spacing\,\mathrm{Precision} = \frac{The\,\mathrm{Predicted}\,Correct\ Spacing}{The\,\mathrm{Predicted}\,Entire\ Spacing} \times 100 \quad (5)$$

식 (1)은 (올바르게 예측된 BI 태그 개수)/(실제 BI 태그 개수)*100으로 복합명사 분해 태그 정확도이고, 식 (2)는 (올바르게 예측된 단위명사 개수)/(실제 단위명사 개수)*100으로 어절 재현율이다. 식 (3)은 (올바르게 예측된 단위명사 개수)/(예측된 전체 단위명사 개수)*100으로 어절 정확도이다. 식 (4)은 공백 삽입 관점에서 (올바르게 예측된 공백 위치 개수)/(실제 전체 공백 위치 개수)*100으로 공백 재현율을 나타낸다. 식 (5)는 공백 삽입 관점에서 (올바르게 예측된 공백 위치 개수)/(예측된 전체 공백 위치 개수)*100으로 공백 정확도를 나타낸다. 표 3은 표 2의 모델 종류 및 하이퍼 파라미터 별로 성능을 평가한 실험 결과이다.

표 3. 복합명사 분해 성능 평가 (단위: %)

모델	복합명사 태그 정확도	어절 재현율	어절 정확도	공백 재현율	공백 정확도
1	92.431	84.04	83.59	86.26	85.18
2	92.478	83.87	83.78	85.93	85.56
3	97.032	93.23	93.16	94.44	94.25
4	97.002	92.95	93.23	94.03	94.50
5	97.035	93.32	93.16	94.54	94.17
6	96.983	93.01	93.15	94.14	94.34
7	93.570	86.04	86.73	87.10	88.34
8	93.522	86.23	86.44	87.54	87.84
9	93.364	85.83	86.17	87.08	87.66
10	93.394	85.75	86.31	86.90	87.89
11	97.357	94.05	94.28	94.76	95.12
12	97.319	93.90	94.24	94.56	95.17
13	**97.368**	94.02	**94.31**	94.72	**95.20**
14	97.296	**94.09**	93.98	**94.98**	94.70

단방향 LSTM과 선형체인 CRF 또는 양방향 LSTM과 선형체인 CRF 모델의 성능 평가 결과 복합명사 태그 정확도, 어절 재현율, 어절 정확도, 공백 재현율 그리고 공백 정확도에서는 단방향 LSTM과 양방향 LSTM과는 상관없이 음절 unigram 벡터만 사용한 경우보다 음절 unigram 벡터와 음절 bigram 벡터를 함께 사용했을 때가 우수함을 보여준다. 음절 unigram과 음절 bigram을 단방향 LSTM과 양방향 LSTM으로 인코딩 할 때는 양방향 LSTM과 선형체인 CRF가 복합명사 분해 태그 정확도는 97.368%, 어절 재현율 94.09%, 어절 정확도 94.31%, 공백 재현율은 94.98%, 공백 정확도 95.20%로 나타났다.

표 4. 기존 연구의 복합명사 분해 성능 (단위: %)

미등록어 존재유무	모델	분해 정확도	어절 정확도
True	이현민(2001)	77.50	
	심광섭(1997)	-	90.60
False	이현민(2001)	99.60	-
	심광섭(1997)	-	98.60

표 4와 같이 기존 기법들은 단위명사 사전에 등록된 어휘의 수에 따라 복합명사 분해의 성능에 차이가 발생한다. 기존의 연구들 중, 이현민(2001)은 3,230개의 복합명사 중 미등록어를 포함하는 복합명사는 444개로 성능 평가를 진행하였고, 심광섭(1997)은 테스트 데이터 복합명사는 4,322개의 어절, 4,322 중 372개의 미등록어 어절로 성능 평가를 진행하였다. 이현민(2001) 미등록어가 포함된 444개 복합명사 분해 시 정확도는 77.50%, 전체 실험 데이터 3,230개 중에서 미등록어를 포함하는 복합명사 444개를 제외한 2786개 복합명사 분해 시 정확도는 99.60%이다. 그리고 이현민(2001)는 전체 실험 데이터인 3,230개의 복합명사를 분해 시 정확도는 96.60%를 보여준다[6]. 심광섭(1997)은 사전에 미등록어가 존재하는 경우 90.60%의 어절 정확도와 미등록어가 존재하지 않을 경우 98.60%의 어절 정확도를 보여준다[9]. 이 기존 연구들은 사전에 미등록어의 존재유무에 따라 복합명사 분해 성능 차이를 보여준다. 하지만 본 논문에서 제안한 단방향 LSTM과 선형체인 CRF 또는 양방향 LSTM과 선형체인 CRF는 음절 unigram과 음절 bigram의 음절 벡터를 함께 사용할 때, 단위 명사 사전을 사용하지 않고 각각 어절 정확도가 93.23%, 94.31%를 보여주어 심광섭(1997)의 미등록어가 존재한 경우인 어절 정확도 90.60% 보다 우수함을 보여주었다. 또한, 본 논문에서 288,971개의 복합명사 데이터는 기존의 연구에서 사용한 데이터양보다 많음에도 어절 재현율, 어절 정확도에서 각각 94.09%, 94.31%를 보여주어 복합명사를 분해 시 단위명사 사전 사용 없이 음절 벡터와 딥러닝 기법을 이용하는 방법이 미등록어 처리에서도 우수함을 보여준다.

표 5는 음절 unigram 벡터만 사용한 모델과 음절 unigram 벡터와 음절 bigram 벡터를 함께 사용한 모델의 복합명사 분해 오류를 보여준다. 음절 unigram 벡터만을 사용한 경우에는 “남북 회담 사무 국장”이라는 복합명사를 분해할 때, 음절 unigram 정보만을 사용하기 때문에 “남북”, “회담사무”, “국장”과 같이 음절 bigram이 하나의 단위명사로 되는 형태를 분해하지 못하는 오류가 발생하지만, 음절 uingram 벡터와 음절 bigram 벡터를 함께 사용한 경우 “남북”, “회담”, “사무”, “국장”의 형태로 분류하였다. 또한, 음절 unigram 벡터만 사용하는 경우에는 “공리주의”를 “공리” 와 “주의”라는 단위명사로 분해하여 복합명사가 하나의 명확한 의미를 갖는 경우와 같은 모호한 복합명사 처리에서 미흡한 결과를 보여주었다. 그리고 “마스크 롬”과 같은 외래어 복합명사의 분해에서는 음절 unigram만 사용한 모델과 음절 unigram과 음절 bigram을 함께 사용한 모델도 분해 오류가 발생하였다.

표 5. 복합명사 분해 결과

모델	<표준>공리주의 적
unigram	공리 주의 적
unigram + bigram	공리주의 적
모델	<표준>남북 회담 사무 국장
unigram	남북 회담사무 국장
unigram + bigram	남북 회담 사무 국장
모델	<표준>마스크 롬
unigram	마스 크롬
unigram + bigram	마스 크롬

V. 결 론

명사 사전 없이 음절 정보만을 사용하여 복합명사를 분해하는 방법으로 복합명사 분해 문제를 순차적인 태그 열 부착 문제로 정의하고 복합명사를 단위명사로 분해하기 위해 음절 임베딩 기법과 단방향 LSTM 및 선형체인 CRF 또는 양방향 LSTM 및 선형체인 CRF를 이용하는 딥러닝 모델을 제안하였다. 그 결과 단방향 LSTM과 양방향 LSTM의 두 가지 모델에서 음절 unigram 벡터만 사용하여 새로운 자질 벡터를 생성하기보다는 음절 unigram 벡터와 음절 bigram 벡터를 함께 연속적인 벡터 공간에 표현하고 새로운 자질 벡터를 생성하였을 때 복합명사 분해 성능인 복합명사 분해 태그 정확도, 어절 재현율, 어절 정확도, 공백 재현율, 공백 정확도가 우수함을 보여 주었다. 그리고 단방향 LSTM보다 양방향 LSTM을 이용하여 음절 unigram 벡터와 음절 bigram 벡터를 새로운 자질

벡터로 표현했을 때가 복합명사 분해 성능이 우수함을 알 수 있었다. 그리하여 양방향 LSTM으로 음절 unigram 벡터와 음절 bigram 벡터를 인코딩하여 새로운 고정된 길이의 자질 벡터를 연속적인 벡터 공간에 표현하고 이를 선형체인 CRF의 입력으로 하여 복합명사의 음절에 복합명사 분해 태그를 부착한 결과 분해 태그 정확도는 97.368%, 어절 재현율 94.09%, 어절 정확도 94.31%, 공백 재현율은 94.98%, 공백 정확도 95.20%를 보여 주었다.

REFERENCES

[1] 이현영, 강승식, "워드 임베딩과 딥러닝 기법을 이용한 SMS 문자 메시지 필터링," *스마트미디어저널*, 제7권, 제4호, 24-29쪽, 2018년 12월

[2] 웅윤지, 강승식, "터치스크린 환경에서 쿼티 자판 오타 교정을 위한 n-gram 언어 모델", *스마트미디어저널*, 제7권, 제2호, 54-59쪽, 2018년 6월

[3] 박승현, 이은지, 김판구, "한글 편집거리 알고리즘을 이용한 한국어 철자 오류 교정 방법," *스마트미디어저널*, 제6권, 제1호, 16-21쪽, 2017년 3월

[4] 원형석, 박미화, 이근배. "복합명사 분할과 명사구 합성을 이용한 통합 색인 기법," *정보과학회논문지 : 소프트웨어 및 응용*, 제27권, 제1호, 84-95쪽, 2000년 1월

[5] Jae Hoon Kim, "Korean Base-Noun Extraction and its Application," *The KIPS Transactions: Part B*, vol. 15, no. 6, pp. 613-620. Dec. 2008.

[6] Hyun Min Lee, Hyuk Ro Park. "Artificial Intelligence: A Reverse Segmentation Algorithm of Compound Nouns," *The KIPS Transactions: Part B*, vol. 8, no. 4, pp. 357-364. Aug. 2001.

[7] Seung-Shik Kang, "A Decomposition Algorithm of Korean Compound Nouns," *Journal of KISS(B): Software and Applications*, vol. 25, no. 1, pp. 172-182, Jan. 1998.

[8] Yong Hoon Lee, Cheol Young Ock, Eung Bong Lee, "Korean Compound Noun Decomposition and Semantic Tagging System using User-Word Intelligent Network," *The KIPS Transactions : Part B*, vol. 19, no. 1, pp. 63-76, Feb. 2012.

[9] Kwangseob Shim, "A Compound Noun Segmentation using Composite Mutual Information," *Journal of KISS(B): Software and Applications*, vol. 24, no. 11, pp. 1307-1317, Nov. 1997.

[10] Huang, Zhiheng, Wei Xu, and Kai Yu. "Bidirectional LSTM-CRF Models for Sequence Tagging," arXiv preprint arXiv:1508.01991. 2015.

[11] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, "Distributed Representations of Words and Phrases and their Compositionality," *In Advances in neural information processing systems*, Lake Tahoe, the United States, pp. 3111-3119, Dec. 2013.

[12] Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov, T., "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol 5, pp. 135-146. Jun, 2017.

[13] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv preprint arXiv:1301.3781, 2013.

저 자 소 개

이현영(학생회원)

2016년 국민대학교 컴퓨터공학부 학사 졸업(공학사).
2016년 ~ 2017년 SK hynix memory solutions inc. Intern
2017년 ~ 현재 국민대학교 컴퓨터공학과 석사과정.

<주관심분야 : 자연어처리, 머신러닝, 딥러닝, 빅데이터 분석, 텍스트 마이닝>

강승식(정회원)

1986년 서울대학교 전자계산기공학과 학사 졸업.
1988년 서울대학교 전자계산기공학과 학과 석사 졸업.
1993년 서울대학교 전자계산기공학과 학과 박사 졸업.

<주관심분야 : 자연어처리, 텍스트 마이닝, 빅데이터 분석, 상황인지 컴퓨팅>