한국어 음소열 기반 워드 임베딩 기술

정의석O, 송화전, 이성주, 박전규

한국전자통신연구원, 음성지능연구그룹
eschung@etri.re.kr, songhj@etri.re.kr, lee1862@etri.re.kr, jgp@etri.re.kr

Korean Phoneme Sequence based Word Embedding

Euisok ChungO, Hwa Jeon Jeon, Sung Joo Lee, Jeon-Gue Park
ETRI, Speech Intelligence Research Group

요 약

본 논문은 한국어 서브워드 기반 워드 임베딩 기술을 다룬다. 미등록어 문제를 가진 기존 워드 임베딩 기술을 대체할 수 있는 새로운 워드 임베딩 기술을 한국어에 적용하기 위해, 음소열 기반 서브워드 자질 검증을 진행한다. 기존 서브워드 자질은 문자 n-gram을 사용한다. 한국어의 경우 특정 단음절 발음은 단어에 따라 달라진다. 여기서 음소열 n-gram은 특정 서브워드 자질의 변별력을 확보할 수 있다는 장점이 있다. 본 논문은 서브워드 임베딩 기술을 재구현하여, 영어 환경에서 기존 워드 임베딩 사례와 비교하여 성능 우위를 확보한다. 또한, 한국어 음소열 자질을 활용한 실험 결과에서 의미적으로 보다 유사한 어휘를 벡터 공간상에 근접시키는 결과를 보여 준다.

주제어: 워드임베딩, 서브워드, 음소열

1. 서론

워드 임베딩 기술은 어휘 목록을 벡터 공간상에 배치하는 기술이다. 이는 유사한 어휘들을 근접한 벡터 공간에 위치하게 한다. Word2vec 공개툴 [1]로 보편화된 워드 임베딩 기술은 다양한 응용 분야에 적용할 수 있다. [2]는 워드 임베딩 기술을 이용하여 텍스트로부터 연결된 문장 체인을 추출하여 도메인 텍스트 확장에 대한 가능성을 보여줬다. 또한 워드 임베딩 기술을 이용하여 클래스 언어모델에 적용하여 성능개선을 시도한 연구도 있었다 [3]. [5]는 한국어 대상의 워드 임베딩 기술로 다양한 학습 패러미터 실험결과를 제시하고 있다. WS353평가셋을 한국어로 번역하여 실험하였는데, 해당 평가셋은 영어 어휘의 다양한 뉘앙스를 기반하고 있어, 해당 워드 유사도 평가셋을 독립적으로 구축하는 것이 옳다고 판단된다.

기존의 워드 임베딩 기술은 미등록어 문제를 갖고 있었다. 즉, 학습 시점에 벡터공간상에 할당할 어휘 목록을 미리 결정해야 했다. 그러나 최근 서브워드 정보 기반 워드 임베딩 연구 [4]는 미등록어 문제 해결 방법을 제시하였다. 이는 각 어휘를 구성 '문자 n-gram'으로 표현하는 방법이다. 학습되는 벡터 값은 '문자 n-gram'을 대상으로 학습되며, 각 어휘는 '문자 n-gram' 벡터 값의 합으로 벡터 공간에 할당된다. 이는 FastText라는 오픈툴로 공개되어 있고, 미리 학습된 결과들을 제공하고 있어, 유용하게 활용할 수 있다. [6]의 경우 한국어 미등록어 워드 임베딩 처리를 위해 음절열 대상 CNN기반 워드 임베딩 기술을 제안하였다. 해당 연구는 응용 DNN 구조에 내재 되었을 경우와 분리되어 단어 단위 입력 레이어로 적용되었을 경우에 대한 비교 후속 연구가 필요하다고 판단된다.

본 논문은 기존 FastText를 재현하여 다양한 자질들을 이용한 워드 임베딩 기술 개발을 위해, 우선 한국어 음소열 기반 워드 임베딩 기술을 검토한다.

2. 서브워드 기반 워드 임베딩

이 장에서는 기존의 워드 임베딩 기술과 서브워드 임베딩 기술의 차이점을 [4]를 참조하여 기술한다. 워드 임베딩 기술의 경우 T 개의 어휘로 구성된 입력 텍스트에 대하여 하나의 단어 w_t의 컨텍스트 어휘 w_c에 대하여 다음의 로그 우도값 (1)을 최대화는 방식으로 기술 할 수 있다. 이는 [1]에 따르면 Skip-gram모델로 명명 된다.

$$\sum_{t=1}^{T}\sum_{c\in C_t}\log p(w_c|w_t) \qquad (1)$$

여기서 $p(w_c|w_t)$는 컨텍스트 어휘의 확률값을 나타내는 소프트맥스 (2)로 기술될 수 있다. 즉, 일종의 W개의 어휘 셋에 대한 언어모델 값이 된다.

$$p(w_c|w_t)=\frac{e^{s(w_t,w_c)}}{\sum_{i=1}^{W}e^{s(w_t,i)}} \qquad (2)$$

(2)에서 $s(w_t,w_c)$는 각 어휘에 해당하는 특정 벡터값들의 스칼라 곱, $u_{w_t}^{\mathrm{T}}v_{w_c}$로 표현할 수 있다. 서브워드 워드 임베딩의 경우, 하나의 어휘를 '문자 n-gram'으로 표현한다고 하였다. "she"의 경우, 최소 3-gram에서 4-gram으로 제한하였을 때, { <sh, <she, she, she>, he> }의 문자 n-gram 집합으로 표현할 수 있고, 해당 문자 n-gram은 벡터값을 갖게 되고, 모든 벡터 값의 합으로 "she"의 벡터값이 결정 된다. 여기서 "<,>"는 단어의 시작과 끝을 표현한다. [4]의 표현을 빌리자면, G개의 문자 n-gram 사전이 주어졌을 때, 어떤 어휘 w가 갖는 문자 n-gram의 집합을 C_w 라고 한다면, C_w 는 문자 n-

gram 사전의 부분 집합이 되고, s(w,c)는 w의 구성 문자 n-gram의 벡터 표현을 z_g라고 했을 때 다음 (3)과 같이 표현된다.

$$s(w,c) = \sum_{g \in C_w} z_g^{\mathrm{I}} v_c \qquad (3)$$

3. 음소열 기반 한국어 워드 임베딩

한국어를 위한 서브워드 접근 방법으로는 문자 n-gram, 자소 n-gram, 음소 n-gram 접근 방법이 가능하다고 판단된다. 본 논문은 문자 n-gram과 음소 n-gram만을 다룬다. 문자 n-gram은 한국어 단어를 구성하는 각 '음절'을 이용하는 반면, 음소 n-gram은 단어의 '발음열'을 이용한다. 직관적으로 음소 n-gram의 경우 자소 n-gram과 차이는 없으리라 본다. 특징을 논의해 보자면, 발음에 대한 표현으로 '독립문'의 경우 '동닙문'이라는 음소열로 표현된다. 즉, 발음기호 'd o N n i xm m u xn'으로 기술 할 수 있다. 음소열은 음성인식에서 사용되는 한국어 g2p (grapheme to phoneme)의 표준 음소셋을 이용한다.[1] 이 접근방법의 장점은 '독선'의 경우 발음은 '독썬'이 되므로, 서브워드 '독'에 대한 분리 표현이 가능하다는 점을 들 수 있겠다. 표 1은 '독립문'의 서브워드 집합 (min=3, max=4)을 보여 준다.

표 1. '독립문'의 서브워드 집합 예

	서브워드 집합
문자 n-gram	{ <_독_립, <_독_립_문, 독_립_문, 독_립_문 _>, 립_문_> }
음소 n-gram	{ <_d_o, <_d_o_N, d_o_N, d_o_N_n, o_N_n, o_N_n_i, …, xm_m_u, xm_m_u_xn, m_u_xn, m_u_xn_> }

4. 실험

한국어 서브워드 워드 임베딩 실험을 위해 c++로 구현된 FastText를 텐서플로우 환경으로 재현하는 작업을 진행하였다. 텐서플로우 튜토리얼에서 제공되는 word2vec 코드를 이용하여, 문자 n-gram 자질을 반영하는 부분과 analogy 평가셋을 사용한 평가를 진행하는 부분을 추가하였다.

실험은 우선 영어환경에서 진행하였다. 100메가 분량의 text8코퍼스를 이용하였고, 어휘셋 빈도수 5이상, 71,290단어, 서브워드 범위 3gram~6gram, 서브워드 총 비율 90%를 이용하였다. 7만여 어휘에 대하여 총 3만 수준의 서브워드 목록이 추출되었다. 배치 크기 64, 임베딩 크기 128, 윈도우 크기 4, 스킵빈도 2, 네거티브 샘플링 개수 10, 서브 샘플링값 1e-3을 적용하였다. 다음 표 2는 word analogy 평가셋을 사용한 평가 결과를 보여준다. 성능 상으로는 기존 word2vec보다 우수한 결과를 보여 준다. 그러나 좀더 대용량 텍스트에 대하여 검증해 볼 필요가 있다고 본다.

표 2. word analogy task 실험 결과

	Accuracy
word2vec	35.8%
sub-word word2vec	39.0%

한국어의 경우 평가셋을 찾을 수 없어, 어휘 유사도, 문장 유사도, 어휘 analogy셋을 직접 구축하기로 시도하는 중이고, 본 논문의 경우 정성 평가와 [6]의 접근방법에 따라 기존 WS353평가셋[2]을 한국어로 변환하여 테스트해 보았다. 사용된 코퍼스는 [2]에서 실험용으로 사용된 82cook코퍼스 중 일부인 35M바이트 분량을 사용하였다. 코퍼스는 word-segmentation 도구를 적용한 상태로 형태소 분석된 결과와 유사하나 어휘 변형이 없는 상태로 보면 된다. 하이퍼 패러미터는 영어 실험과 동일하고, 어휘 셋만 1만 5천 단어로 제한하였다. 정성 평가는 특정 어휘와 벡터 공간상에 근접해 위치한 어휘 출력으로 하였다. 본 논문에서 제안된 음소열 기반 sub-word word2vec (p-sub-word word2vec)와 기존의 word2vec의 유사 어휘 목록 중 차이가 있는 항목에 대하여 표 3에서 기술한다. 또한 기존 문자 n-gram (s-sub-word word2vec) 결과도 포함하였다.

표 3. 유사 어휘 실험 결과 (top5)

어휘	p-sub-word word2vec	s-sub-word word2vec	word2vec
요구_하	요구 요구_했 밝히 지급_하 밝히_고	요구 요청_하 밝히 지키 거부_하	거부_하 적용_하 강조_하 밝히 강화_하
어르신	어르신들 노인 부모_님 부모님 할아버지	노인 어르신들 노인들 할머니 연세	여성_분 노인 분 회원_님 선배_님
시어머니	시어머님 어머니 시아버지 시모 며느리	시아버지 어머니 시모 시어머님 어머님	어머니 시아버지 시모 시엄니 아버지
깨끗_한	깨끗_하 깨끗_해 깨끗 깔끔_한 깔끔_하	깨끗_하 깔끔_한 깔끔 안전_한 깔끔_하	깔끔_한 조용_한 오래된 깨끗_하 어두운
선택	결정 판단 선택_할 선택_한 선택_하	결정 선택_할 선택_했 선택_한 판단	판단 결정 노력 성공 행동

[1] 에트리 음성인식 엔진 내부 dd-g2p, https://itec.etri.re.kr/itec/sub02/sub02_01_1.do

[2] WordSim353 - Sim. and Rel. http://alfonseca.org/eng/research/wordsim353.html

표 3에서 알 수 있는 점은 p-sub-word 실험의 경우 유사 의미의 어휘가 더 근접해 있다는 데 있다. '요구_하'의 경우, word2vec은 유사한 컨텍스트를 보이는 어휘들이 벡터 공간상 근접해 있으므로, 실제 반대 되는 의미를 가진 어휘 들이 유사 어휘로 선정되었으나, p-sub-word의 경우 발음의 유사성이 유사 컨텍스트 클러스터링 현상을 회피할 수 있게 한다고 볼 수 있다. s-sub-word 실험의 경우 표 3 두 실험의 중간 정도 수준의 출력 결과를 보였다.

서론에서 전술했듯이 WS353을 한국어로 변환하는 접근 방법은 부적절하다고 판단된다. 해당 평가셋은 영어 어휘의 다양한 의미를 고려하여 설계 되었다. 따라서 번역 어휘의 선정은 단어 쌍의 할당 점수를 무의미 하게 한다. 그러나, 한국어에 대한 적절한 평가셋을 구할 수 없어, 본 논문은 [6]의 접근방법에 따라, 번역을 통하여 WS353 평가를 진행해 보았다.

표 4. 한국어 WS353 평가

	(min, max) / 총 subword수 / word당 고정 subword 수	WS353 -S	WS353 -R
[6]	-	0.67	0.49
문자 n-gram	(2, 4) / 15k / 21	0.40	0.32
음소 n-gram	(2, 5) / 10k / 56	0.40	0.30
	(2, 4) / 5k / 29	0.34	0.27
	(3, 6) / 24k / 30	0.35	0.33

표 4는 한국어 WS353 평가 결과를 보여 준다. 기존 연구 [6]에 비해서 성능 결과는 좋지 않았다. [6]의 경우 학습 데이터도 어휘 3k 단어, 텍스트 427k 단어로 구성된 소량 뉴스 텍스트로 도출한 결과이다. 본 연구의 경우는 15k단어, 텍스트 8000k 단어를 이용한 결과이다. 두 실험의 WS353 평가셋이 다른 점과 본 연구의 텍스트 도메인이 인터넷 게시판 도메인이라는 차이점이 있다. 향후 다양한 경로로 검토와 보완을 진행할 예정이다.

표 4에서 (min, max)는 서브워드 자질의 (최소 길이, 최장 길이)이다. '총 subword 수'는 1만 5천 단어로부터 추출된 subword 개수를 말한다. '고정 subword 수'는 워드를 구성하는 서브워드 개수의 최장 길이가 된다. 실험 결과는 문자 n-gram과 음소 n-gram 결과의 큰 차이를 보여 주지 못하고 있다. 음소 n-gram 실험 결과들의 차이점은 다양한 패러미터 최적화 작업의 필요성을 보여준다. 실험의 epoch당 성능 개선 수준은 문자 n-gram이 더 안정적인 결과를 보여 줬다. 음소 n-gram의 경우, g2p결과를 이용하므로, 다중 발음이라는 문제점을 갖고 있다. 즉 한 단어를 다양한 발음으로 표현할 수 있다는 점인데, 본 논문에서는 최소 길이의 발음열을 적용하였다. 향후 이 부분에 대하여도 검토해 볼 예정이다.

5. 결론 및 향후 연구

본 논문은 한국어 서브워드 기반 워드 임베딩 기술 개발을 위해 음소열 기반 서브워드 접근 방법을 제시하였다. 이를 위해 FastText를 텐서플로우 환경으로 재현하였고, 영어 환경에서 성능 검증을 진행하였다. 한국어의 경우 음소열 기반 워드 임베딩 도구를 개발하였고, 이를 이용하여 기존의 word2vec와 비교하였다. 정성 실험 결과 좀 더 의미적으로 유사한 어휘가 근접 어휘로 추출되는 결과를 보였다. 반면 한국어 WS353평가에서는 기존 연구보다 좋지 않은 결과를 보였다. 향후 한국어 어휘 유사도, 문장 유사도, 어휘 analogy 평가셋 구축을 진행하여 정량 평가를 시도하여, 다양한 서브워드 자질에 대하여 연구를 진행할 계획이다. 또한, 영어 환경에 음소열 서브워드 자질 실험도 진행하여, 본 연구의 일반성 검증을 진행할 예정이다.

감사의 글

이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임. (R0126-15-1117, 언어학습을 위한 자유발화형 음성대화처리 원천기술 개발)

참고문헌

[1] T. Mikolov et al., "Distributed Representations of Words and Phrases and their Compositionality", Int. Conf. NIPS, pp. 3111-3119, 2013

[2] E. Chung and G. Park, "Sentence-Chain Based Seq2seq Model for Corpus Expansion", ETRI Journal, Vol. 39, Num. 4, pp. 455-466, Aug. 2017.

[3] 정의석, 박전규, "워드 임베딩과 품사 태깅을 이용한 클래스 언어모델 연구", 정보과학회 컴퓨팅의 실제 논문지, 제22권, 제7호, pp. 315-319, 2016.7.

[4] P. Bojanowski et al., "Enriching Word Vectors with Subword Information", arXiv:1607.04606v2, Jun. 2017.

[5] 최상혁, 설진선, 이상구, "한국어에 적합한 단어 임베딩 모델 및 파라미터 튜닝에 관한 연구", 제28회 한글 및 한국어 정보처리 학술대회 논문집, 2016.

[6] 최상혁, "음절 기반 한국어 단어 임베딩 모델 및 학습 기법", 서울대학교 공학석사학위논문, 2017.