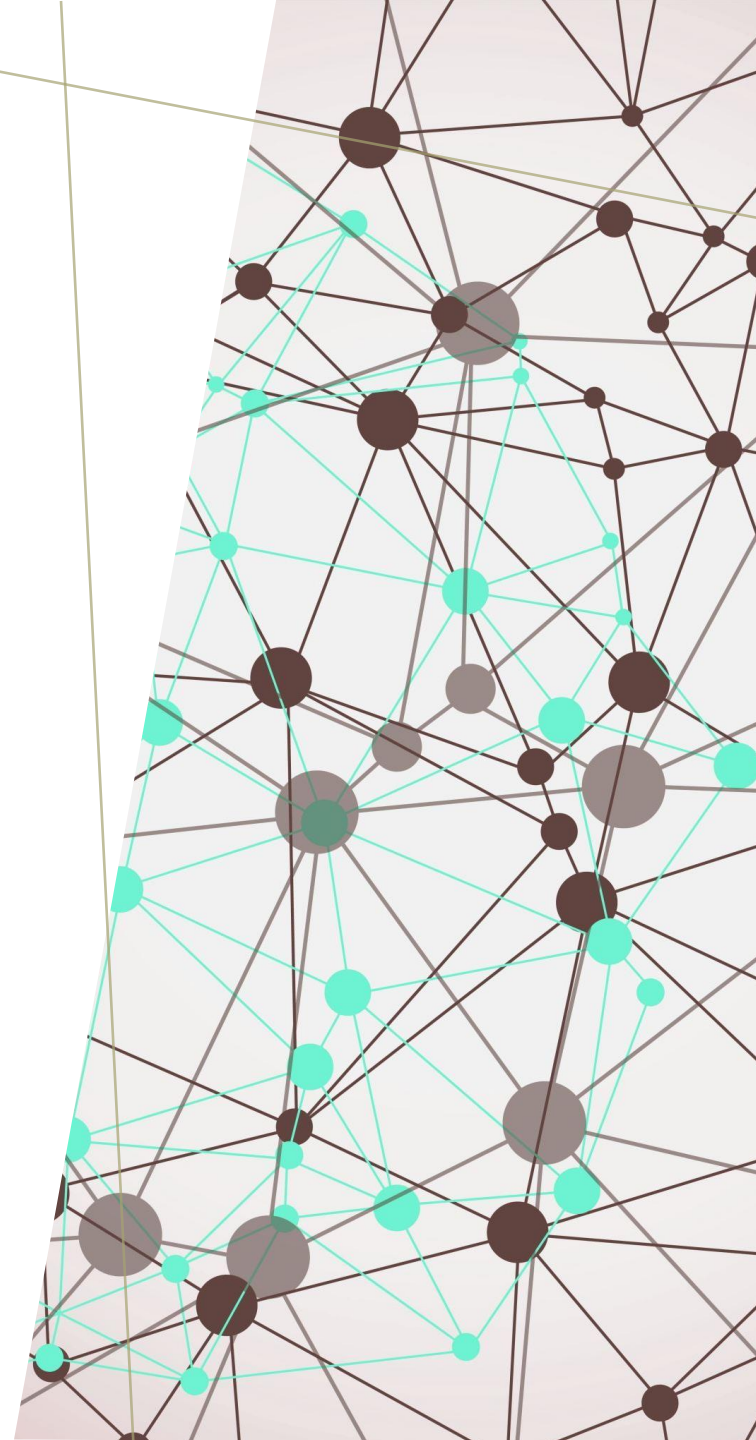
논문 리뷰

Neural Machine Translation of Rare Words with Subword Units

사이트(HTTPS://ARXIV.ORG/PDF/1508.07909)



소개

기존의 신경망 기계 번역(NMT) 모델은 고정된 어휘집을 사용해서 그 데이터에 존재하는 단어만 번역이 가능

-> 어휘는 제한되어 있어 어휘에 포함되지 않은 단어, 즉 희귀 단어를 번역하는 데 어려움이 존재

-> 드문 단어와 새로운 단어를 서브워드 유닛의 시퀀스로 인코딩하여 더 유연하게 다양한 단어를 번역하는 방법이 필요

==> 서브워드 유닛의 시퀀스로 인코딩 방법을 다뤄
하위 단어 단위로 희귀 단어를 분할하여
번역 모델의 성능을 향상시키는 것이 목표

서브워드 유닛의 시퀀스로 인코딩이란?

- 예시) 독일어 단어 'Abwasserbehandlungsanlage'는 '하수 처리장'을 의미하는데, 이 단어는 'Abwasser' (하수), 'behandlungs' (처리), 'anlage' (시설)처럼 나눌 수 있음
- 이 단어를 고정된 길이로 저장하는 것보다, 이렇게 여러 부분으로 나누어 변할 수 있는 길이로 저장하는 것이 더 자연스럽고 이해하기 쉬움
- 서브워드 유닛을 통하여 다음 2가지를 보여주려함

1. 개방 어휘 신경망 기계 번역이 가능함을 보여줘, 글쓴이의 아키텍처가 큰 어휘집과 사전 대체 기법을 사용하는 것보다 더 간단하고 효과적이라는 것

2. 단어를 더 작은 단위로 나누기 위해 'byte pair encoding' (BPE)이라는 압축 방법을 사용

NMT(신경망 기계 번역)이란?

- 순환 신경망을 갖춘 인코더-디코더 네트워크로 구현

인코더 : 입력 시퀀스 x = (x1, …, xm)를 읽는 gated recurrent units로 구성된 양방향 신경망

디코더 : 대상 시퀀스 y = (y1, …, yn)를 예측하는 순환 신경망

요약 : 입력 문장을 읽는 부분은 양방향으로 정보를 수집하여 중요한 단어들을 찾아내고, 번역 문장을 생성하는 부분은 이 정보를 기반으로 번역을 수행

Subword Translation

- Subword(하위 단어)의 필요성 :

  단어보다 작은 단위로 텍스트를 분할하여 번역의 유연성을 높임

ex) "unhappiness"라는 단어를 "un", "happi", "ness"로 분할하면, "unhappy"와 "happiness"에서도 동일한 하위 단위를 재사용 가능

BPE(Byte Pair Encoding)

- BPE는 하위 단어 단위를 생성하는 데 사용되는 방법
- BPE는 가장 빈번하게 함께 나타나는 문자 쌍을 반복적으로 병합하여 새로운 하위 단어 단위를 만듦

먼저, 문자 어휘를 사용하여 심볼 어휘를 초기화

각 단어는 문자들의 시퀀스로 표현되며, 각 단어의 끝에는 특별한 '·' 기호가 추가되는데, 이 기호는 번역 후에 원래의 단어 구조를 되돌릴 때 사용함.

다음으로, 모든 심볼 쌍을 세어 가장 많이 나타나는 쌍을 찾음.

예) 'A'와 'B'가 많이 나타나면 이 쌍을 'AB'라는 새로운 심볼로 대체

이런 방식으로, BPE 알고리즘은 특별한 목록이 필요하지 않고, 자주 나타나는 부분들을 합쳐서 새로운 심볼로 만듦.

최종적으로 생성된 심볼 어휘의 크기는 초기 문자 어휘의 크기에 추가된 합병 작업의 수와 같다.

BPE 알고리즘

```python
import re, collections

def get_stats(vocab): # 주어진 어휘집(vocab)에서 각 단어의 문자열 쌍 빈도를 계산하는 함수
    pairs = collections.defaultdict(int)
    for word, freq in vocab.items():
        symbols = word.split() # 단어를 공백 기준으로 분할하려 리스트로 저장
        for i in range(len(symbols)-1): # 단어를 구성하는 각 문자열 쌍에 대해 빈도를 계산
            pairs[symbols[i],symbols[i+1]] += freq # 각 문자열 쌍의 빈도를 누적하여 기록
    return pairs

def merge_vocab(pair, v_in): # 합병된 문자열 쌍을 새로운 단위로 대체해 어휘집을 업데이트하는 함수
    # pair: 합병할 문자열 쌍
    # v_in: 입력으로 받은 어휘집
    v_out = {}
    bigram = re.escape(' '.join(pair)) # pair로 주어진 문자열 쌍을 정규 표현식 패턴으로 변환
    p = re.compile(r'(?<!\S)' + bigram + r'(?!\S)')
    # 정규 표현식 패턴을 컴파일하여 문자열에서 pair를 찾음.

    for word in v_in:
        w_out = p.sub(''.join(pair), word)
        # 각 단어에서 pair를 찾아 새로운 단위로 합병하여 새로운 단어를 생성
        v_out[w_out] = v_in[word]
        # 새로운 어휘집에 합병된 단어를 추가하고, 빈도를 유지
    return v_out
```

```python
vocab = {'l o w </w>' : 5, 'l o w e r </w>' : 2, 'n e w e s t </w>':6, 'w i d e s t </w>':3}
num_merges = 10  # 합병 작업을 10번
for i in range(num_merges):
    pairs = get_stats(vocab) # 문자열 쌍의 빈도를 계산하고, 가장 빈도가 높은 문자열 쌍(best)을 선택
    best = max(pairs, key=pairs.get)
    vocab = merge_vocab(best, vocab) # 선택된 문자열 쌍을 합병하고, 어휘집을 업데이트
    print(best) # 각 합병 작업에서 선택된 가장 빈번한 문자열 쌍을 출력
```

Python

```
('e', 's')
('es', 't')
('est', '</w>')
('l', 'o')
('lo', 'w')
('n', 'e')
('ne', 'w')
('new', 'est</w>')
('low', '</w>')
('w', 'i')
```

- BPE 외에도 문자 n-그램, 형태소 분할 등 다양한 방법 존재
- BPE는 특히 NMT 모델에 적합하여 널리 사용

실험(모델에 쓰일 세분화 지정)

segmentation	# tokens	# types	# UNK
none	100 m	1 750 000	1079
characters	550 m	3000	0
character bigrams	306 m	20 000	34
character trigrams	214 m	120 000	59
compound splitting△	102 m	1 100 000	643
morfessor*	109 m	544 000	237
hyphenation◇	186 m	404 000	230
BPE	112 m	63 000	0
BPE (joint)	111 m	82 000	32
character bigrams (shortlist: 50 000)	129 m	69 000	34

- Tokens: 토큰 수
- Types: 고유 단어 수
- UNK: 미지의 토큰 수

글자 세분화

복합어 세분화: 책가방 -> 책+가방

Morfessor: 설치된 -> 설치+된

하이픈 세분화

BPE

빈도 높은 50000만개 단어만 원래 형태 유지

높은 고유 단어수와 **낮은** 미지의 토큰수를 가진 것이 best

실험(모델 설명)

- WUnk(백오프 사전 없이)
- WDict(백오프 사전 사용)
- C2-50k
- BPE-60k(병합 연산 6만)
- BPE_J90k(병합 연산 9만)

name	segmentation	shortlist
syntax-based (Sennrich and Haddow		
WUnk	-	-
WDict	-	-
C2-50k	char-bigram	50 000
BPE-60k	BPE	-
BPE-J90k	BPE (joint)	-

✓ 백오프사전(back-off dictionary): 기계 번역에서 특정 단어나 구문이 번역 모델의 훈련 데이터에 존재하지 않거나 모델이 그 단어나 구문을 번역할 수 없는 경우에 사용되는 보조 사전

실험

영어
->
독일어

name	segmentation	shortlist	vocabulary source	vocabulary target	BLEU single	BLEU ens-8	CHRF3 single	CHRF3 ens-8	unigram F_1 (%) all	unigram F_1 (%) rare	unigram F_1 (%) OOV
syntax-based (Sennrich and Haddow, 2015)					24.4	-	55.3	-	59.1	46.0	37.7
WUnk	-	-	300 000	500 000	20.6	22.8	47.2	48.9	56.7	20.4	0.0
WDict	-	-	300 000	500 000	22.0	24.2	50.5	52.4	58.1	36.8	**36.8**
C2-50k	char-bigram	50 000	60 000	60 000	**22.8**	**25.3**	51.9	53.5	58.4	40.5	30.9
BPE-60k	BPE	-	60 000	60 000	21.5	24.5	**52.0**	53.9	58.4	40.9	29.3
BPE-J90k	BPE (joint)	-	90 000	90 000	**22.8**	24.7	51.7	**54.1**	**58.5**	**41.8**	33.6

영어
->
러시아어

name	segmentation	shortlist	vocabulary source	vocabulary target	BLEU single	BLEU ens-8	CHRF3 single	CHRF3 ens-8	unigram F_1 (%) all	unigram F_1 (%) rare	unigram F_1 (%) OOV
phrase-based (Haddow et al., 2015)					24.3	-	53.8	-	56.0	31.3	16.5
WUnk	-	-	300 000	500 000	18.8	22.4	46.5	49.9	54.2	25.2	0.0
WDict	-	-	300 000	500 000	19.1	22.8	47.5	51.0	54.8	26.5	6.6
C2-50k	char-bigram	50 000	60 000	60 000	**20.9**	**24.1**	49.0	51.6	55.2	27.8	17.4
BPE-60k	BPE	-	60 000	60 000	20.5	23.6	**49.8**	52.7	55.3	29.7	15.6
BPE-J90k	BPE (joint)	-	90 000	100 000	20.4	**24.1**	49.7	**53.0**	**55.8**	**29.7**	**18.3**

BLEU: 기계 번역된 결과와 사람이 제공한 기준 번역 사이의 일치 정도를 측정(단일 / 앙상블)

CHRF3: 문자 n-gram의 정밀도와 재현율을 결합하여 평가(단일 / 앙상블)

Unigram F1 score: 단일 단어의 정밀도와 재현율의 조화 평균을 계산하여 측정(모든 단어 / 희귀 단어 / 사전에 없는 단어)

실험

영어 -> 독일어

name	segmentation	shortlist	vocabulary source	vocabulary target	BLEU single	BLEU ens-8	CHRF3 single	CHRF3 ens-8	unigram F_1 (%) all	unigram F_1 (%) rare	unigram F_1 (%) OOV
syntax-based (Sennrich and Haddow, 2015)					24.4	-	55.3	-	59.1	46.0	37.7
WUnk	-	-	300 000	500 000	20.6	22.8	47.2	48.9	56.7	20.4	0.0
WDict	-	-	300 000	500 000	22.0	24.2	50.5	52.4	58.1	36.8	**36.8**
C2-50k	char-bigram	50 000	60 000	60 000	**22.8**	**25.3**	51.9	53.5	58.4	40.5	30.9
BPE-60k	BPE	-	60 000	60 000	21.5	24.5	**52.0**	53.9	58.4	40.9	29.3
BPE-J90k	BPE (joint)	-	90 000	90 000	**22.8**	24.7	51.7	**54.1**	**58.5**	**41.8**	33.6

영어 -> 러시아어

name	segmentation	shortlist	vocabulary source	vocabulary target	BLEU single	BLEU ens-8	CHRF3 single	CHRF3 ens-8	unigram F_1 (%) all	unigram F_1 (%) rare	unigram F_1 (%) OOV
phrase-based (Haddow et al., 2015)					24.3	-	53.8	-	56.0	31.3	16.5
WUnk	-	-	300 000	500 000	18.8	22.4	46.5	49.9	54.2	25.2	0.0
WDict	-	-	300 000	500 000	19.1	22.8	47.5	51.0	54.8	26.5	6.6
C2-50k	char-bigram	50 000	60 000	60 000	**20.9**	**24.1**	49.0	51.6	55.2	27.8	17.4
BPE-60k	BPE	-	60 000	60 000	20.5	23.6	**49.8**	52.7	55.3	29.7	15.6
BPE-J90k	BPE (joint)	-	90 000	100 000	20.4	**24.1**	49.7	**53.0**	**55.8**	**29.7**	**18.3**

결론

1. **신경망 기계 번역 시스템**이 드문 단어와 알려지지 와 알려지지 않은 단어를 서브워드 단위의 연속으로 표현함으로써 개방 어휘 번역이 가능하다는 것을 보여주었고, 백오프 번역 모델보다 간단하면서도 효과적이다.
2. 단어 세분화를 위한 **바이트 페어 인코딩(BPE)**의 변형을 소개하였으며, 이는 변수 길이의 서브워드 단위의 조밀한 기호 어휘로 개방 어휘를 인코딩할 수 있다.
3. BPE 세분화와 간단한 문자 바이그램 세분화 모두를 사용하여 **고정된 어휘 문제를 해결**하고 기존에 비해 **향상된 성능**을 보았다.
4. 언어 쌍에 따라 어휘 크기와 같은 언어 특정 요소에 상대적 효과는 달라질 수 있지만, 서브워드 세분화가 대부분의 언어 쌍에 적합하다.