복사 매커니즘을 이용한 한국어-제주어 기계번역

박다솔 ○, 차정원
창원대학교
dasol_p@changwon.ac.kr, jcha@changwon.ac.kr

Machine Translation of Korean-Jeju Language Using Copy Mechanism

Da-Sol Park ○, Jeong-Won Cha
Changwon National University

요 약

본 논문에서 한글로 표기하는 특성상 제주어와 한국어에는 겹치는 어휘가 상당히 많다는 점에 집중하였고, 한국어-제주어의 단어 치환이라는 데이터 특성을 고려하여 복사 매커니즘을 이용한다. 복사 매커니즘은 포인터 생성 네트워크를 사용했으며, 이는 입력 문장 또는 사전 내 단어를 선택하여 생성할지를 결정한다. 또 목표 언어의 토큰을 우선순위로 선택하게 결정하며 이를 이용한 보상(reward)을 적용하였다. 데이터는 카카오브레인에서 구축한 JIT(Jejueo Interview Transcripts)를 이용하였고, BLEU를 통해 토큰 단위 성능을 측정하였다. '한국어→제주어 실험'은 검증 데이터는 47.85, 평가 데이터는 47.12를 보였다.

1. 서 론

딥러닝의 등장으로 최근 주로 신경망 기계 번역(Neural Machine Translation) 모델에 대한 많은 연구가 진행되고 있으며[1,2], 많은 성능 향상을 이루어냈다. 기존의 기계번역 데이터는 소스 문장과 타겟 문장의 교집합이 존재하지 않는다. 하지만 한국어-제주어 기계번역 데이터셋은 한글을 이용해 글자를 표기하는 특성상 겹치는 어휘가 상당히 많다.

따라서 우리는 이렇게 겹치는 어휘가 많다는 점에 집중하였고, 한국어-제주어의 단어 치환이라는 데이터 특성을 고려하여 포인터 생성 네트워크(Pointer-Generator Network)[3]를 복사 매커니즘으로 이용한다. 또 목표 언어(제주어)의 토큰으로 보상(reward)을 계산하고, 이를 이용한 학습을 통해 한국어-제주어 기계번역의 성능 향상을 목표로 하였다.

최근 트랜스포머[4]라는 모델이 발표되었으며 기존 방법론들에 비해 좋은 성능을 보이고 있어, [1,2]와 같이 트랜스포머 기반 연구가 이루어지고 있다.

2. 제안 방법

그림 1은 제안 모델 구조이다. 해당 구조는 [5]를 활용하였으며, 해당 논문 구조 중 타겟 어텐션 분포 부분을 제외하였다. 제안 모델은 제주어 학습 코퍼스에만 있는 토큰을 특정 토큰 사전으로 추가적용하여 토큰 분포를 생성하는 데 영향을 준다.

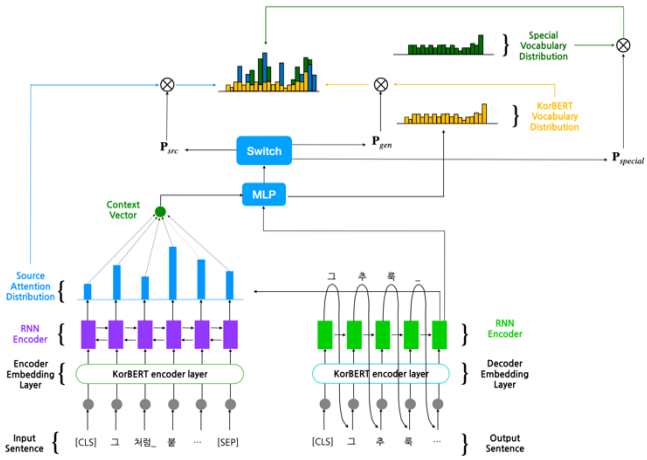

그림 1 제안 모델 구조

인코더와 디코더 모두 워드피스(wordpiece)[6] 단위로 토큰화(tokenization)하고, ETRI에서 공개한 한국어 BERT 언어 모델인 KorBERT[7]의 임베딩을 거친다. 인코더는 순서를 고려한 양방향 LSTM(bi-directional LSTM)[8]을 이용하고, 순방향과 역방향 인코더의 은닉 상태(hidden state)를 연결(concatenate)하여 사용한다. 디코더는 단방향 LSTM(uni-directional LSTM)을 사용하며 인코더 마지막 상태를 디코더의 초기 상태(initial state)로 설정한다.

입력 문장의 토큰에 대한 확률인 소스 어텐션 분포는 인코더와 디코더의 은닉 상태를 입력으로 하여 바다나우 어텐션(Bahdanau attention)을 이용하여 계산한다. 문맥 벡터(context vector)는 소스 어텐션 벡터의 가중합(weighted sum)을 적용한다. 문맥 벡터와 인코더의 은닉 상태를 결합하여 전체 사전 내 생성 확률을 구한다. 생성 게이트 확률은 문맥 벡터와 디코더 은닉 상태의 최종 출력 벡터를 연결(concatenate)하여 2개의 MLP(Multi-Layer perceptron)를 거친다. 이 때 $P_{src} + P_{gen} + P_{special} = 1$을 만족해야 한다.

우리는 우선순위를 가지고 선택하도록 지역 사전(Local vocab)과 전역 사전(Global vocab)을 이용한다. 타켓 문장의 토큰들을 지역 사전으로, KorBERT의 토큰들을 전역 사전으로 생성하였다. 단어 선택 시 지역 사전, 입력 문장 내 사전, 전역 사전 순으로 우선순위를 가진다.

$$\hat{y_t} = W_V \cdot LSTM(h_{t-1}, x_t) + b_V \quad (1)$$

$$\text{Mask}_{v=0}^{|V|} = \begin{cases} 1, \text{if } \alpha \le v \le \beta \\ 0, \text{ else} \end{cases} \quad (2)$$

$$\text{reward} = \sum(\hat{y} * \text{Mask}) \quad (3)$$

$$Loss_{sentence} = CE\left(y^{(t)}, \hat{y}^{(t)}\right) = -\sum_{j=1}^{|V|} y_j^{(t)} log\, \hat{y}_j^{(t)} \quad (4)$$

$$Loss_{total} = Loss_{sentence} + (1 - \text{reward}) \quad (5)$$

우리는 모델이 현재 시간의 토큰을 예측한 확률을 이용하여 보상을 적용하고자 하였다. 현재 시간의 토큰을 예측한 확률은 식 (1)과 같다. W_V 와 b_V 는 시스템이 학습 데이터를 통해 학습되는 가중치를 뜻하고, V 는 사전 크기를 말한다. 그리고 $\mathrm{h_{t-1}}$ 은 t-1번째 은닉 상태를 의미하고, x_t 는 t 번째 단어를 의미한다.

특정 토큰 마스킹 벡터는 특정 토큰 사전 내 존재하지 않은 토큰들의 확률을 0으로 만들어주는 역할을 한다. 식 (2)로 나타내며, 특정 토큰 사전의 시작 인덱스는 α 이고, 끝 인덱스는 β 이다. 보상 값은 현재 시간의 토큰을 예측한 확률과 특정 토큰 마스킹 벡터를 곱하여 합하여 스칼라 값으로 나타낸다. 보상 값과 손실 함수 값은 상반 관계임으로 보상 값이 커지면 손실 함수를 작게 만들기 위해 (1-보상값)을 손실 함수에 추가한다.

최종 손실 함수(식 (5))는 1:1의 비율을 적용하였으며, 출력된 문장과 정답 문장을 이용한 손실 함수(식 (4))와 보상 값(식 (3))을 더한 값을 통해 모델을 업데이트시킨다.

3. 데이터

실험은 제주어 학습 데이터셋인 JIT(Jejueo Interview Transcripts)를 이용하였다. 카카오브레인에서 구축한 데이터[9]로써, '한국어 문장-제주어 문장'으로 구성된 17만 개의 병렬 데이터셋이다. 해당 데이터는 학습, 검증, 평가 데이터로 분류되어 있으며 데이터셋의 통계는 표 1과 같다.

표 2는 JIT 데이터셋 예시이고, 표 내 굵은 글씨는 치환되는 대상 단어와 목표 단어를 표시하였다. JIT 데이터셋은 이러한 예시들과 같이 단어 치환으로 된 문장으로 구성되어 있다.

표 1 JIT 데이터셋 통계

분류	문장 수
학습 데이터	160,356
검증 데이터	5,000
평가 데이터	5,000

표 2 JIT 데이터셋 예시

분류	한국어 문장	제주어 문장
(1)	**그럼** 고추장은 안 **담가서 먹었습니까** ?	**게민** 고추장은 안 **담강 먹엇수과** ?
(2)	**맞수다** . 아이고 . **게문 이거는** .	**맞습니다** . 아이고 . **그럼 이것은** .

4. 실험 및 분석

학습에 사용한 하이퍼 파라미터는 문장의 토큰 임베딩은 768차원이고, 문자 임베딩은 100차원이다. 인코더 은닉층 크기는 512차원이며 디코더 은닉층 크기는 1024이며, 각각 2층을 사용한다. 인코더와 디코더에 동일하게 드롭 아웃(dropout)은 0.33이다. 학습 배치 사이즈는 256이고, 옵티마이저는 Adam을 이용하였다. 학습률은 0.001로, 조기 종료(early stop)는 5로 설정하였다. 보상을 위해 제주어 토큰 사전은 5,305개의 토큰으로 구성되어 있다. 성능 측정 방법은 BLEU[10]로 설정한다.

표 3은 어절 단위의 실험 성능표이다. 비교 모델은 한국어를 제주어로 번역하는 트랜스포머 모델인 'JIT'와 JIT 데이터와 한국어 위키 데이터셋을 함께 학습한 모델인 'JIT+KorWiki'이다. 두 모델은 [11]에 작성된 성능을 명시하였다. 'Ours'는 제안 모델에서 $P_{special}$ 과 특정 토큰 분포(Special token distribution) 모듈이 제외된 모델이다. 'Ours(+reward)'는 제안 모델로써, 제주어 특정 사전 토큰을 우선순위와 보상에 이용한 모델이다. 'JIT'와 'JIT+KorWiki' 모델은 토크나이저를 바이트 페어 인코딩(Byte-Pair Encoding, BPE)[12]을 적용하였고, 제안 모델은 wordpeice를 적용하였다.

표 3 모델 성능(어절 단위)

분류	검증 코퍼스	평가 코퍼스
JIT	44.85	43.31
JIT+KorWiki	45.25	44.19
Ours	46.40	44.66

Ours(+reward)	47.85	47.12

실험에서 제안하는 두 모델이 비교 모델에 비해 검증 및 평가 데이터에서 모두 높은 성능을 보였다. 정성평가에 대한 데이터는 평가 데이터 중 10%인 500문장을 랜덤 샘플링하여 진행하였다.

정성평가 기준은 총 7개로 분류하며, '출력 문장 내 OOV가 포함된 문장'은 시스템 출력 문장에 OOV(out-of-vocabulary)가 포함되어 있는 문장이고, '정답과 동일하게 출력한 문장'은 정답과 동일하게 출력한 문장이다. '다른 단어로 생성한 문장'은 정답 문장과는 달리 다른 단어로 생성한 문장이며 '특수 기호 제외 시, 정답과 동일한 문장'은 설정한 특수기호(온점,쉼표, 물음표,느낌표)를 제외하였을 때 정답과 동일한 문장이다. '잘못된 사투리'는 코퍼스 내 한번도 나타나지 않은 사투리를 생성하거나 잘못된 토큰의 조합 또는 문법적 에러가 포함된 문장을 의미한다. '문장 도중 생성 실패'는 문장 생성을 하다가 도중에 중단된 문장을 의미한다.

표 4는 실험의 정성평가 통계를 보여준다. 실험에서 보상을 사용한 모델이 정답과 동일하게 생성한 문장 수가 증가하였고, 또 다른 단어로 생성한 문장 수 또한 증가하였다. 잘못된 사투리 또는 문법적 에러의 문장 생성 수와 문장 도중 생성 실패의 문장 수가 감소하여 성능 향상에 도움을 주었다.

표 4 평가 코퍼스의 정성 평가 통계

분류	기준	Ours	Ours (+reward)
(1)	출력 문장 내 OOV가 포함된 문장	62	59
(2)	입력 문장 내 단어로 출력한 문장	22	10
(3)	잘못된 사투리 또는 문법적 에러	20	17
(4)	문장 도중 생성 실패	19	9
(5)	정답과 동일하게 출력한 문장	53	69
(6)	다른 단어로 생성한 문장	321	332
(7)	특수 기호 제외 시, 정답과 동일한 문장	3	4

5. 결론 및 향후 연구

본 논문에서는 한글을 이용해 글자를 표기하는 특성상 제주어와 한국어에는 겹치는 어휘가 상당히 많다는 점에 집중하였다. 이러한 특성을 고려하여 포인터 생성 네트워크를 한국어-제주어 기계번역에 적용해보았다. 그리고 모델 개선을 위하여 우선적으로 생성하고자 하는 특정 토큰에 대하여 보상을 적용하였다. 실험 결과, 비교 모델에 비해 평가 코퍼스 기준 2.93의 성능 향상을 보였다. 향후 연구로는 성능 향상을 위해 모델 구조와 어순 및 유창성에 대한 보상 수식을 개선하고 기계 번역을 위한 다른 데이터셋 또는 다른 언어에 대해서도 실험할 계획이다.

참 고 문 헌

[1] Duan, Sufeng, Hai Zhao, Junru Zhou, and Rui Wang. "Syntax-aware transformer encoder for neural machine translation." In 2019 International Conference on Asian Language Processing (IALP), pp. 396-401. IEEE, 2019.

[2] Xia, Yingce, Tianyu He, Xu Tan, Fei Tian, Di He, and Tao Qin. "Tied transformers: Neural machine translation with shared encoder and decoder." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 5466-5473. 2019.

[3] See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with pointer-generator networks." arXiv preprint arXiv:1704.04368, 2017.

[4] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." In Advances in neural information processing systems, pp. 5998-6008. 2017.

[5] Zhang, S., Ma, X., Duh, K., and Van Durme, B. "AMR Parsing as Sequence-to-Graph Transduction", In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 80-94, 2019.

[6] Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Neural machine translation of rare words with subword units." arXiv preprint arXiv:1508.07909, 2015.

[7] http://aiopen.etri.re.kr/service_dataset.php, 2020-11-01.

[8] Hochreiter, Sepp, and Jürgen Schmidhuber, "Long short-term memory." , Neural computation 9.8 pp. 1735-1780, 1997.

[9] Park, Kyubyong, Yo Joong Choe, and Jiyeon Ham. "Jejueo Datasets for Machine Translation and Speech Synthesis." In Proceedings of The 12th Language Resources and Evaluation Conference, pp. 2615-2621, 2020.

[10] Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. "BLEU: a method for automatic evaluation of machine translation." In Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pp. 311-318, 2002.

[11] https://www.kakaobrain.com/blog/119, 2020-11-01.

[12] Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Neural machine translation of rare words with subword units." arXiv preprint arXiv:1508.07909, 2015.