ISSN 2383-6318(Print) / ISSN 2383-6326(Online)
KIISE Transactions on Computing Practices, Vol. 28, No. 3, pp. 190-195, 2022. 3
https://doi.org/10.5626/KTCP.2022.28.3.190

복사 매커니즘을 이용한 한국어-제주어 기계번역

(Machine Translation of the Korean-Jeju Language using the Copy Mechanism)

박 다 솔 †　　차 정 원 ††
(Da-Sol Park)　　(Jeong-Won Cha)

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 타겟 언어의 토큰을 우선순위로 선택하게 결정하며 이를 이용한 보상(reward)을 적용하였다. 데이터는 카카오브레인에서 구축한 JIT(Jejueo Interview Transcripts)를 이용하였고, BLEU를 통해 어절 단위 성능을 측정하였다. 한국어→제주어 실험은 검증 데이터는 47.85, 평가 데이터는 47.12를 보였고, 제주어→한국어 실험은 검증 데이터는 69.01, 평가 데이터는 68.54를 보였다.

키워드: 기계번역, 기계학습, 자연어 처리, 복사 매커니즘, 한국어-제주어

Abstract In this paper, we focused on the fact that there isare a lot of overlapping vocabulary in the Jeju language and Korean language due to the characteristics of writing in the Korean language, and we used a copying mechanism in consideration of the data characteristic of word substitution in the Korean-Jeju language. The copying mechanism used a pointer generation network, which determines whether to generate by selecting an input sentence or word in the dictionary. In addition, it was decided to select the token of the target language as a priority and applied a reward was applied using this it [오전1] . Fordata, JIT (Jejueo Interview Transcripts) built by Kakao Brain was used, andperformance per eojeol was measured through BLEU. In the Korean-Jeju language experiment, the validation data was 47.85, → and the test data was 47.12, and in the Jeju language Korean experiment, the validation data was → 69.01, and the test data was 68.54.

Keywords: machine translation, machine learning, natural language processing, copy mechanism, korean-jeju language

· 이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No.2021-0-00354, 비정형 텍스트를 학습하여 쟁점별 사실과 논리적 근거 추론이 가능한 인공지능 원천기술)
· 이 논문은 2020 한국소프트웨어종합학술대회에서 '복사 매커니즘을 이용한 한국어-제주어 기계번역'의 제목으로 발표된 논문을 확장한 것임

† 비 회 원 : 창원대학교 친환경해양플랜트FEED공학과 학생
dasol_p@changwon.ac.kr
†† 종신회원 : 창원대학교 컴퓨터공학과 교수(Changwon Nat'l Univ.)
jcha@changwon.ac.kr
(Corresponding author임)

논문접수 : 2021년 4월 1일
(Received 1 April 2021)
논문수정 : 2021년 12월 13일
(Revised 13 December 2021)
심사완료 : 2021년 12월 28일
(Accepted 28 December 2021)

정보과학회 컴퓨팅의 실제 논문지 제28권 제3호(2022. 3)

1. 서 론

딥러닝의 등장으로 최근 주로 신경망 기계번역(Neural Machine Translation) 모델에 대한 많은 연구가 진행되고 있으며[1,2], 많은 성능 향상을 이루어냈다. 기계번역은 주로 소스 언어와 타겟 언어가 다른 데이터 셋을 이용한다. 예를 들어 영어를 불어로 번역하거나, 독어를 영어로 번역하는 태스크 등이 있다. 기존의 기계번역 데이터는 입력 문장과 출력 문장이 다른 언어이므로 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 언어의 토큰으로 보상(reward)을 계산하고, 이를 이용한 학습을 통해 한국어-제주어와 제주어-한국어 기계번역의 성능 향상을 목표로 하였다.

해외에서는 이미 방언 관련 연구들이 많이 진행되어 왔다. 아랍어에 관련된 방언 기계번역 연구로는 [4,5]가 존재한다. [4]는 Neural Machine Translation(NMT)와 멀티태스크(multitask) 방식으로 진행하였으며, 순환신경망(Recurrent neural network) 인코더(Encoder)-디코더(Decoder) 모델에 기반하고, 아랍어 방언에서 표준 아랍어로 번역하는 작업과 세그먼트 단위 품사 태깅 작업을 공유하는 통합된 모델을 적용하여 방언 기계 번역에 적용하였다. [5]는 입력 소스와 타겟 언어에 같은 벡터 공간을 사용하여 표준어를 기반으로 방언들의 동의어가 비슷한 벡터를 갖도록 하였다.

2. 제안 방법

2.1 문제 정의

기계번역은 한 자연어를 다른 자연어로 자동 변환하고 입력 문장의 의미를 보존하며 타겟 언어로 유창한 텍스트를 생성하는 태스크이다. 본 논문에서 고려한 기계번역의 정의는 다음과 같다. n개의 단어로 구성된 입력문장 $S=\{w_{s_0}, w_{s_1}, \cdots, w_{s_n}\}$이 주어지면, 우리는 입력 문장의 의미를 보존하는 출력 문장 $T=\{w_{t_0}, w_{t_1}, \cdots w_{t_m}\}$ 이때, n과 m은 동일하지 않을 수 있다.

2.2 제안구조

그림 1은 제안 모델 구조이다. 해당 구조는 [6]을 활용하였으며, 해당 논문 구조 중 타겟 어텐션 분포 부분을 제외하였다. 제안 모델은 타겟 언어의 학습 코퍼스에만 있는 토큰을 특정 토큰 사전으로 추가 적용한다. 타겟 언어를 위한 토큰 분포를 통해 타겟 언어에 대한 생성에 영향을 주도록 설정하였다. 실험 내 적용된 설정 값은 3.2에 명시하였다. 본 연구에서는 타겟 언어의 생성 빈도를 더 높이기 위해서 각 타겟 언어 토큰에 빈도에 따른 리워드를 부여하여 타겟 언어 내 토큰 생성 빈도의 확률은 높이고 이를 더하여 타겟 언어 토큰이 생성되는 확률을 낮추도록 손실 함수에 리워드의 영향을 받도록 설정하였다.

2.3 인코더와 디코더

인코더와 디코더 모두 워드피스(wordpiece)[7] 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.

2.4 문맥 벡터 및 생성 게이트 확률

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 연결하여 2개의 MLP(Multi-Layer perceptron)를 거친다. 이 때 $P_{src}+P_{gen}+P_{special}=1$을 만족해야 한다. 이때, P_{src}는 소스 문장에 대한 토큰 확률 값이고, P_{gen}는 전체 사전 내 토큰 확률 값이고, $P_{special}$는 타겟 언어에 대한 토큰 사전 확률 값을 의미한다.

우리는 우선순위를 가지고 선택하도록 지역 사전(Local vocab)과 전역 사전(Global vocab)을 이용한다. 전역 사전은 토크나이저의 토큰 사전을 의미하고, 지역 사전은 정답 문장에 나타난 토큰 사전, 특정 토큰 사전은 타겟 언어의 학습 코퍼스에만 있는 토큰을 가진다. 단어 선택 시 지역 사전, 특정 토큰 사전, 전역 사전 순으로 우선순위를 가진다.

2.5 손실 함수

우리는 모델이 현재 시간의 토큰을 예측한 확률을 이용하여 보상을 적용하고자 하였다. 현재 시간의 토큰을 예측한 확률($\hat{y_t}$)은 식 (1)과 같다. W_V와 b_V는 시스템이 학습 데이터를 통해 학습되는 가중치를 뜻하고, V는 사전 크기를 말한다. 그리고 h_{t-1}은 t-1번째 은닉 상태를 의미하고, x_t는 t 번째 단어를 의미한다. 특정 토큰 마스킹 벡터($Mask$)는 특정 토큰 사전 내 존재하지 않은

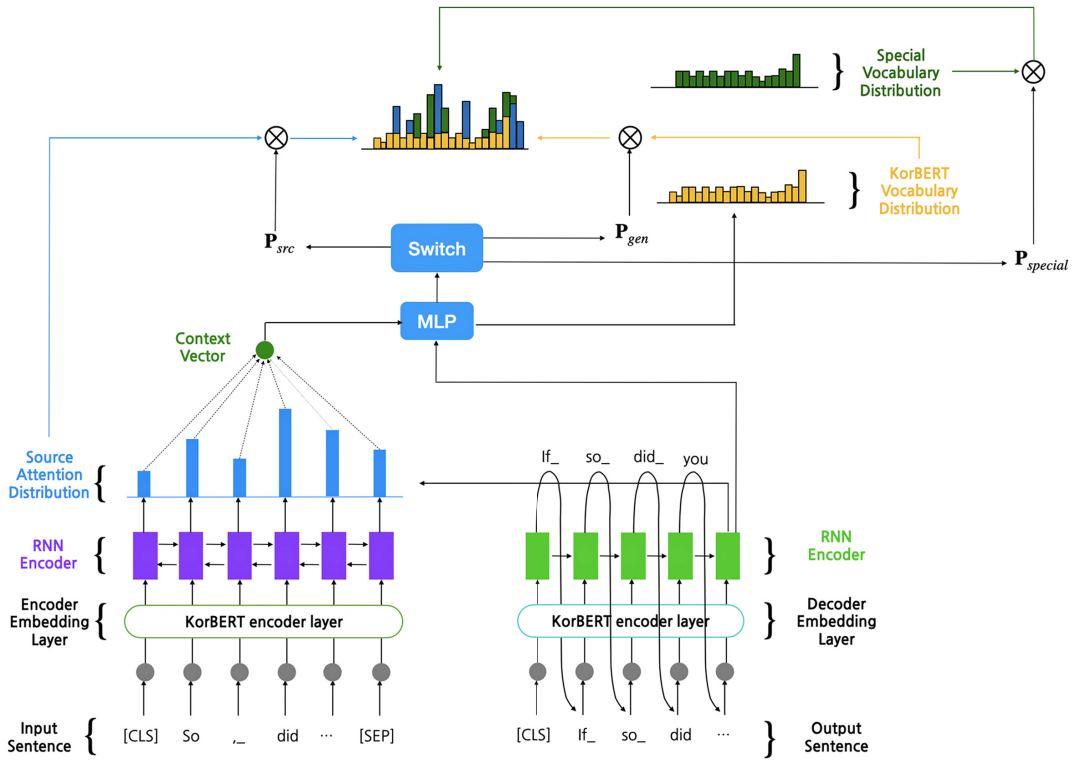

그림 1 제안 모델 구조

Fig. 1 Structure of the proposed model

토큰들의 확률을 0으로 만들어주는 역할을 한다. 식 (2)로 나타낸다.

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

$$Mask^{|V|}_{v=0} \begin{cases} 1, \text{if } v \text{ is target index} \\ 0, \text{else} \end{cases} \tag{2}$$

보상 값($reward$)은 현재 시간의 토큰을 예측한 확률과 특정 토큰 마스킹 벡터를 곱하여 합하여 스칼라 값으로 나타낸다. 그림 2는 보상 값을 계산하는 방식을 보여준다.

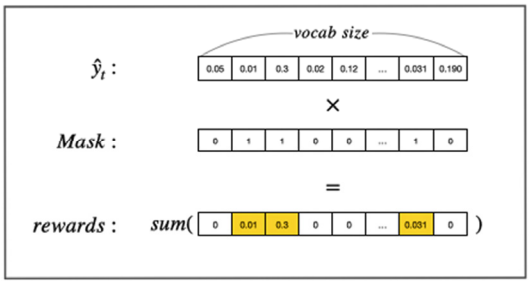

그림 2 보상 값 계산

Fig. 2 Calculation of the reward value

보상 값과 손실 함수 값은 상반 관계이므로 보상 값이 커지면 손실 함수를 작게 만들기 위해 (1-보상 값)을 손실 함수에 추가한다.

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

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

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

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. 실험 방법 및 결과

3.1 데이터셋

우리는 실험을 위해 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. 그리고 JIT 데이터셋 내 입·출력이 동일한 문장에 대한 통계는 표 2와 같다.

표 1 JIT 데이터셋 통계
Table 1 Statistics of the JIT dataset

Classification	Number of pair (Korean-jeju language)
Train dataset	160,356
Validation dataset	5,000
Test dataset	5,000

표 2 JIT 데이터셋 내 입·출력이 동일한 문장 통계
Table 2 Statistics of Input·Output identical sentence in the JIT dataset

Classification	Number of sentences (including white-spaces)	Number of sentences (not including white-spaces)
Train dataset	41,729	41,799
Validation dataset	183	190
Test dataset	208	209

표 3 각 데이터별 한국어와 제주어 중복어절 비율
Table 3 Ratio of duplicate words in Korean and Jeju language for each data

Classification	Duplicate Eojeol rate
Train dataset	51.81%
Validation dataset	47.57%
Test dataset	47.83%

한국어 문장-제주어 문장의 예시는 아래와 같다. 한국어 문장인 '볏짚도 소 먹이고.'와 제주어 문장인 '노록찍도 쉐 먹이고.'는 '볏짚'이 '노록찍'으로, '소'가 '쉐'로, '먹이고'가 '멕이고'로 번역되어야 한다. 또 한국어 문장인 '그럼 고추장은 안 담가서 먹었습니까?'와 제주어 문장인 '게민 고추장은 안 담강 먹엇수과?'는 '그럼'이 '게민'으로, '담가서'가 '담강'으로 '먹었습니까?'가 '먹엇수과?'로 번역되어야 한다. 위의 예시 중 굵은 글씨로 표기 해놓은 부분은 한국어 문장과 제주어 문장이 중복되는 어절이며, 이런 중복 결과에 대한 통계를 진행하였다. 표 3은 JIT 데이터셋의 한국어 문장과 제주어 문장의 중복어절 통계이다.

3.2 실험 설정

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, 한국어 토큰 사전은 5,460개의 토큰으로 구성되어 있다.

비교 모델은 표준 트랜스포머 모델인 'JIT'와 JIT 데이터와 한국어 위키 데이터셋을 함께 학습한 모델인 'JIT+KorWiki'이다. 두 모델은 [10]에 작성된 성능을 명시하였다. 'Ours(-reward)'는 제안 모델에서 $P_{special}$과 특정 토큰 분포(Special token distribution) 모듈이 제외된 모델이다. 'Ours'는 제안 모델로써, 타겟 언어의 특정 사전 토큰을 우선순위와 보상에 이용한 모델이다. 표준 트랜스포머 모델을 사용한 비교 모델(JIT, JIT+KorWiki)과 달리, 제안 모델 중 Ours(-rewards)는 포인터 생성 네트워크를 적용한 모델이고, Ours는 포인터 생성 네트워크와 타겟 언어에 대한 보상을 함께 적용한 모델을 의미한다.

'JIT'와 'JIT+KorWiki' 모델은 토크나이저를 바이트 페어 인코딩(Byte-PairEncoding, BPE)[11]을 적용하였고, 제안 모델은 워드피스(wordpeice)를 적용하였다.

3.3 한국어→제주어 실험 및 분석

성능 평가 지표는 BLEU[12]로 설정한다. JIT와 JIT+KorWiki는 BLEU로 성능 평가를 하였고, Ours(-reward)와 Ours는 BLEU 1~4의 평균을 이용하였다. 표 4는 어절 단위의 실험 성능 표이다. 실험에서 제안하는 두 모델이 비교 모델에 비해 검증 및 평가 데이터에서 모두 높은 성능을 보였다.

정성 평가에 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, '입력 문장 내 단어로 출력한 문장'은 타겟 언어가 아닌 소스 언어 내 단어를 출력한 경우이다. '문법적 오류'는 코퍼스 내 한번도 나타나지 않은 단어를 생성하거나 잘못된 토큰의 조합 또는 문법적 에러가 포함된 문장을 의미한다. '문장 도중 생성 실패'는 문장 생성을 하다가 도중에 중단된 문장을 의미한다.'정답과 동일하게 출력한 문장'은 정답과 동일하게 출력한 문장이다. '특수 기호 제외 시, 정답과 동일한 문장'은 설정한 특수 기호를 제외하였을 때 정답과 동일한 문장을 의미한다. 이때 특수 기호는 온점, 쉼표,

표 4 한국어-제주어 번역 성능(어절 단위)
Table 4 Performance of the Korean-Jeju language translation (Eojeol unit)

Classification	Validation dataset	Test dataset
JIT	44.85	43.31
JIT+KorWiki	45.25	44.19
Ours(-rewards)	46.40	44.66
Ours	**47.85**	**47.12**

표 5 한국어-제주어 번역의 정성 평가 통계
Table 5 Qualitative evaluation statistics of Korean-Jeju language translation

Classification	Ours (-reward)	Ours
Include OOV in output sentence	62	59
Output as words in the input sentence	22	10
Grammatical error	20	17
Sentence generation failure	19	9
Output the same as the correct answer	53	69
Same as correct answer, except for special symbols	321	332
Generated in other words of target language	3	4

물음표, 느낌표 총 4가지로 설정하였다. '타겟 언어의 다른 단어로 생성한 문장'은 정답 문장과는 달리 타켓 언어의 다른 단어로 생성한 문장을 말한다.

표 5는 한국어→제주어 실험 내 평가 코퍼스 결과에 대한 정성평가 통계를 보여준다. 한국어→제주어 실험에서 보상을 사용한 모델이 정답과 동일하게 생성한 문장 수가 증가하였고, 또 다른 단어로 생성한 문장 수 또한 증가하였다. 문법적 에러의 문장 생성 수와 문장 도중 생성 실패의 문장 수가 감소하여 성능 향상에 도움을 주었다.

제안 모델 중 Ours(-reward)와 Ours의 결과를 비교해보았다. 예를 들어, '무슨 김치해요?'라는 표준어 문장을 '무신 짐치헤마씨?'라는 제주어 문장으로 번역해야하나, Ours(-reward)의 결과는 '무슨 김치헤마씨?'로 번역하는 반면, Ours의 결과는 '무슨 짐치헤마씨?'로 번역하였다. 그리고 '많이, 그러니 그때들 막 거기 무서워 했었지.'라는 표준어 문장을 '하영, 게난 그때들 막 그디 무서완 헤낫주게 .'로 제주어 문장으로 번역해야하나 Ours(-reward)의 결과는 '많이, 게난 그때덜 막 그디 무서워서 헤낫주게'로 번역하는 반면, Ours의 결과는 '하영, 게난 그때들 막 그디 ㅁ·서완 헤낫주'로 번역하였다.

타겟 언어의 보상을 사용하기 전에는 한국어에서 제주어로 번역이 제대로 되지 않은 경우들이 존재했으나, 보상을 적용한 모델은 타겟 언어에 대한 토큰을 생성하는 경우와 아래아(·)를 많이 사용하는 제주어의 특성을 반영한 결과를 보여 성능 향상을 보였다고 할 수 있다.

3.4 제주어→한국어 실험 및 분석

표 6은 어절 단위의 실험 성능표이다. 제주어→한국어 실험은 검증 데이터셋에서 JIT+KorWiki 모델이 가장 높은 성능을 보였으나, 평가 데이터셋에서는 제안하는 모델이 가장 높은 성능을 보였다. 표 7은 제주어→한국어 실험 내 평가 코퍼스의 결과에 대한 정성평가 통계를 보여준다. 제주어→한국어 실험에서 정답과 동일하게 출력한 문장 수로 인해 정량평가 성능이 향상되었으며,

표 6 제주어→한국어 번역 성능(어절 단위)
Table 6 Performance of Jeju→Korean language translation (Eojeol unit)

Classification	Validation dataset	Test dataset
JIT	69.39	67.70
JIT+KorWiki	**69.59**	67.94
Ours(-rewards)	68.94	68.38
Ours	69.01	**68.54**

표 7 제주어-한국어 번역의 정성 평가 통계
Table 7 Qualitative evaluation statistics of Jeju-Korean language translation

Classification	Ours (-reward)	Ours
Include OOV in output sentence	4	4
Output as words in the input sentence	64	61
Grammatical error	36	31
Sentence generation failure	17	13
Output the same as the correct answer	154	170
Same as correct answer, except for special symbols	192	201
Generated in other words of target language	33	20

한국어의 다른 단어로 생성한 문장 수가 증가하였다. 그리고 입력 내 제주어 단어로 생성한 출력의 문장 수와 문법적 에러 문장과 문장 도중 생성에 실패한 문장의 수가 감소하여 성능 향상에 도움을 주었다.

동일한 데이터로 진행한 기계번역 태스크임에도 불구하고 제주어를 한국어로 번역하는 모델의 BLEU 성능이 더 높은 이유에 대해서 분석을 진행한다. 한국어 문장에서 나타나는 '어머니'라는 단어는 '어멍', '어무니', '어므니' 등과 같이 다양한 제주어 단어로 나타난다. 즉, 다양한 어휘 수가 나타나는 제주어 문장을 생성하는 경우에 정답과 동일한 단어 대신 다른 표현인 단어로 생성할 가능성이 높다. 위와 같은 특성으로 인해 정답과 비교하는 정량 평가에 대해서는 한국어→제주어 실험의 성능이 제주어→한국어 실험 성능에 비해 낮은 성능을 보인다고 분석된다.

4. 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-제주어와 제주어-한국어 기계번역에 적용해보았다. 그리고 모델 개선을 위하여 우선적으로 생성하고자 하는 타겟 문장에 나타나는 특정 토큰에 대하여 보상을 적용하였다. 실험 성능 지표의 비교는 평가

코퍼스 기준으로 어절 단위의 성능 향상을 측정하였다. 한국어-제주어 실험 결과, 비교 모델에 비해 2.93의 성능 향상을 보였다. 그리고 제주어-한국어 실험 결과, 비교 모델에 비해 0.6의 성능향상을 보였다.

향후 연구는 손실함수에 대한 가중치를 설정하여 확장하고자 한다. 최종 손실함수는 보상 값과 손실 함수 값이 1:1의 비율을 가지도록 설정하였다. 보상 값에 가중치를 성능 향상을 위해 적용해보며, 또 모델 구조와 어순 및 유창성에 대한 보상 수식을 개선하고 기계번역을 위한 다른 데이터셋 또는 다른 언어에 대해서도 실험할 계획이다.

References

[1] Duan, Sufeng, Hai Zhao, JunruZhou, and Rui Wang, "Syntax-aware transformer encoder for neural machinetranslation," *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," *Proc. 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] Baniata, Laith Hasan Okleh, "A multitask-based neural machine translation model for arabic dialects," *Domestic PhD thesis Kyungpook National University Graduate School*, 2019. Dea-gu.

[5] W. Farhan, B. Talafha, A. Abuammar, R. Jaikat, M. AlAyyoub, A. B. Tarakji, and A. Toma, "Unsupervised dialectal neural machine translation," *Information Processing Management*, Vol. 57, No. 3, p. 102181, 2020.

[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, 2021-03-12.

[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," *Proc. of The 12th Language Resources and Evaluation Conference*, pp. 2615-2621, 2020.

[10] [Online]. Available:https://www.kakaobrain.com/blog/119, (2022-03-14)

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

[12] Papineni, Kishore, SalimRoukos, Todd Ward, and Wei-Jing Zhu, "BLEU: a method for automatice-valuation of machine translation," *Proc. of the 40th annualmeeting of the Association for Computational Linguistics*, pp. 311-318, 2002.

박 다 솔

2014년 창원대학교 학사. 2017년 창원대학교 석사. 2017년~현재 창원대학교 친환경해양플랜트. FEED공학과(정보통신·컴퓨터전공) 박사수료. 관심분야는 자연어처리, 딥러닝, 기계학습

차 정 원

숭실대학교(학사). 포항공과대학교(석사, 박사). USC/ISI(박사후연수). 2004년~현재 창원대학교 컴퓨터공학과 교수. 관심분야는 자연어처리, 기계학습, 정보검색