오픈소스 최종보고서

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 지도교수 | 컴퓨터공학과 | | | 정유철 교수님 |
| 구분 | 학년 | 학번 | 성명 | 역할 분담 |
| 팀장 | 3 | 20160519 | 박유성 | 기계 번역, 자료 조사 분석 |
| 팀원 | 3 | 20160328 | 김주원 | 데이터 수집, 기계 번역 |
| 팀원 | 3 | 20160726 | 엄진웅 | 기계 번역, 자료 조사 분석 |
| OpenNMT를 사용한 한-영/영-한 기계번역 | | | | |

**0. 목차**

1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

2. 관련 연구. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

3. 제안 방법 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

4.실험 결과 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

5. 분석 & Discussion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

6. 결론 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

7. 참고 문헌 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

1. **Introduction**

한글은 현존하는 문자 중 가장 많은 발음을 표기할 수 있는 문자라고 한다. ‘빨강'이라는 단어를 가지고도 ‘붉은', ‘빨간’, ‘불그스름’, ‘벌건’ 등 여러가지로 표현될 수 있는 우수한 문자이다. 반면 우리의 일상에서 가장 가까운 영어를 가지고 표현할 수 있는 단어들은 ‘red’가 전부이다. 이로 보아 영어의 표현이 한글의 의미를 완전히 표현하는 것은 불가능하다. 그렇기에 한글은 표음 문자이며 위대한 문자라는 평을 받는 것이다. 한글에서 영어로 번역을 한다면 그 표현이 기존의 한글에 비해서는 의미 전달이 덜 한 감이 있을 수 있다. 이전에 지금처럼 스마트폰과 인터넷이 보편적이지 않을때 번역을 하려면 필요한 단어 하나하나를 사전을 통해 찾아보았다. 이는 돌이켜 생각해본다면 굉장히 불편하고 시간이 많이 소비되며 번역은 어려운 일로 인식이 되었다. 하지만 스마트 폰과 인터넷이 보편적으로 된 일상에서 우리는 일상적으로 한글을 영어로 또는 영어를 한글로 번역하여 사용할 일이 많다. 그때마다 보통 인터넷을 통한 검색을 사용하게 된다. 그렇지만 그 번역 또한 위의 ? 처럼 영어로 표현하기 힘든 번역에서는 얼마나 뛰어난 성능을 가질지 의문이 된다. 만약 제대로 표현이 되지 않는다면 한글에서 영어로의 번역에서 완벽한 문장 번역은 어떻게하면 이루어지게 될지 생각해보았다.

기계번역은 학습량을 늘리고 학습 데이터의 양을 늘려준다면 성능 향상이 가능해진다. 그렇다면 다음과 같은 의문을 가질 수 있다. ‘만약 기계 번역을 통해 많은 데이터를 학습시킨다면 이러한 의미에 더욱 가깝게 사용할 수 있게 번역해주지 않을까?’라는 생각을 가질 수 있다. 우리는 일반적으로 일상속에서 번역을 해야 한다면 보통 구글 번역기 또는 파파고 등의 성능이 좋은 번역기를 사용하게 된다. 이 번역기를 사용한 번역에서 한글을 영어로 번역하기 어려운 문장을 번역한 것과 우리의 기계번역을 통해 학습된 모델에서의 번역에서는 어떤 결과가 나타나고 차이를 가지는지 그리고 얼마나 많은 양의 데이터와 학습을 할 경우 이 번역 만큼의 성능 혹은 그 이상의 성능을 낼 수 있는지에 초점을 맞추어 보았다. 위와 같은 실험을 하기위해서는 많은 데이터를 사전에 가지고 있어야한다. 그래서 데이터 셋은 AI Hub에 있는 한글의 자연스러운 구어체 문장, 상황/시나리오 기반 대화 세트, 정부/지자체 홈페이지, 간행물, 행정 규칙, 자치 법규,한국 역사, 문화 콘텐츠, 뉴스 텍스트에 대한 한영 병렬 자료를 약 160만개 수집하였고, 이 데이터들을 기계학습 시킨 모델에서 한글과 영어 사이의 관계는 서로의 번역에서 어떤 영향을 미치는지 조금이나마 알아보도록 몇가지 조건을 달리하여 학습시켜 보았다.

1. **관련 연구**

관련 연구로는 고려대학교 컴퓨터학과에서 진행한 병렬 코퍼스 필터링과 한국어에 최적화된 서브 워드 분절 기법을 이용한 기계번역 논문을 찾을 수 있었다.

이 연구에서는 기계번역 모델에 집중하기보다는 학습데이터를 어떤식으로 전처리를 하여 기계번역의 성능을 효과적으로 끌어올릴 수 있느냐를 중점으로 다루었다.

이 논문은 데이터를 전처리하는 과정을 병렬 코퍼스 필터링으로 시작하여 아주 흥미로운 실험 결과를 내었다.

그 중에서도 가장 눈길을 끌었던 것은 한국어에 최적화된 Tokenization 실험이었다.

Tokenization방법을 5가지 사용하는데, BPE, Sentence Piece Unigram, 한국어 형태소 분석기인 Mecab Tokenization, 조사를 분리 후 Sentence Piece Unigram, 복합명사를 분해하고 조사를 분리한 뒤, Sentence Piece Unigram등에 관한 Tokenization을 한 다음 기계 번역에 대한 결과를 보여주었다.

결과는 아주 놀라웠고 데이터 전처리를 함으로써 결과가 현저히 바뀔 수 있다는 것을 깨달았다.

1. **제안 방법**

OpenNMT 2.0 version을 이용하여 한-영, 영-한으로 machine translation을 해보았습니다.

총 5단계를 걸쳐서 진행을 하였다.

첫 번째로는, 데이터를 준비한다.

저희는 각각 한글 train data set, test data set, validation data set과 영어 train data set, test data set, validation data set을 AI Hub사이트에서 총 160만개를 가져왔고, 6:2:2로 train, test, validation data set을 나누었다.

두 번째로는, BPE를 사용하여 가져온 data set들을 Tokenization을 한다.

이 단계를 거치는 이유는 OOV 문제를 해결하기 위해서 쓰는데, OOV란 기계가 모르는 단어가 등장하면 그 단어를 단어 집합에 없는 단어로 해석하여 Out-Of-Vocabulary(또는 UNK(Unknown Token))라고 표현을 한다.

BPE를 사용함으로써, Subword를 분리시켜 하나의 단어는 더 작은 단위의 의미있는 여러 Subword들로 분리해서 단어를 임베딩을 시켜준다.

이를 통해 OOV문제를 완화시킬 수 있다.

세 번째로는, 준비된 병렬 문장들을 분석해 어휘의 빈도순으로 저장을 합니다.

만약, 1만개의 단어들을 추려내어 순위를 메기고 텍스트 파일로 저장을 하였다면, 그 텍스트 파일안에 있지 않은 단어들은 모두 제거가 됩니다.

이 단계에서 쓸데없는 단어들에 대해 제거를 하여 훈련과 번역을 수월하게 할 수 있습니다.

train data set과 validation data set들을 모델 훈련에 사용하였다면, test data set은 번역 단계에 사용되어진다.

세 번째로는, 사용할 데이터를 지정해준다.

사용할 데이터를 지정하려면 YAML 구성 파일을 빌드해야 하는데, 이 YAML 파일에는 샘플이 작성 될 위치, 어휘가 작성 될 위치, train data 경로, test data 경로, 세 번째 단계에서 만든 어휘 파일 경로, GPU 사용여부, 체크 포인트를 저장할 위치와 train steps와 valid steps가 각각 몇 번 돌 때, 저장하겠냐를 지정해줄 수 있다.

네 번째로는, 모델을 훈련시킨다.

OpenNMT 2.0 version의 기본옵션인 인코더와 디코더 모두 500개의 hidden units들의 2계층 LSTM 모델의 구성과 0.3의 dropout을 사용하였다.

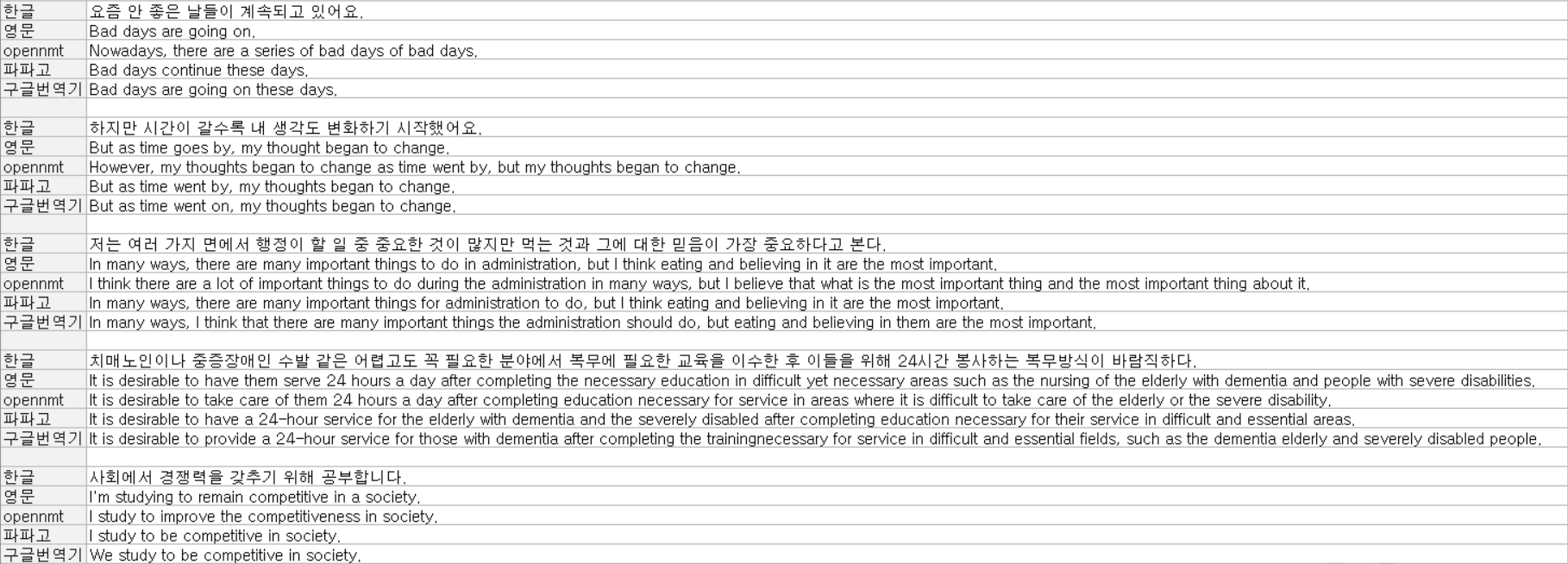
다섯 번째로는, 번역 단계이다. train data set과 validation data set을 사용하여 학습되어진 모델로 test data set을 번역한다.

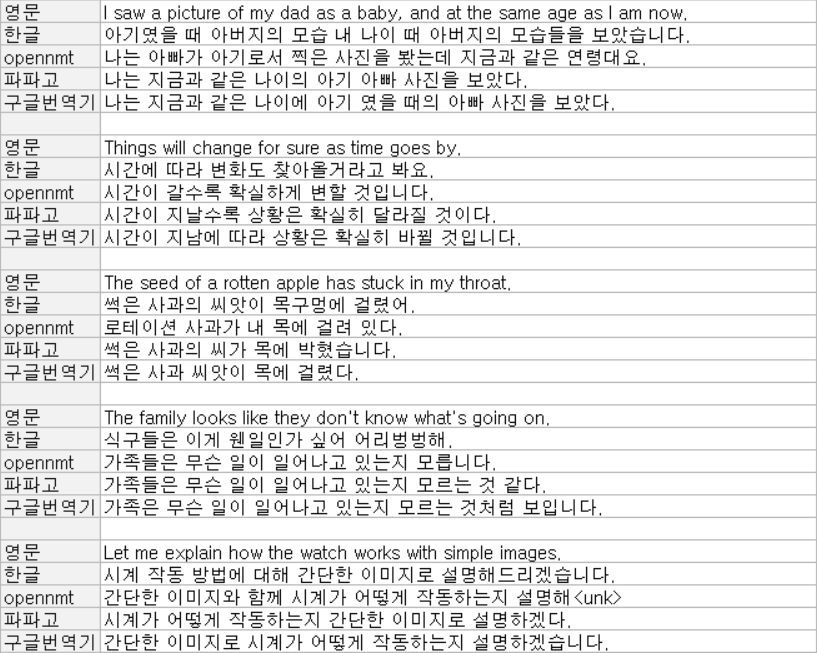
마지막으로는, 번역된 test파일에 대한 Detokenization을 하여 BPE를 제거해주고, 원본 test 파일과 비교를 하여 BLEU Score를 평가한다.

BLEU Score란 언어 모델의 성능 측정을 위한 평가 방법으로, 기계 번역 결과와 사람이 직접 번역한 결과가 얼마나 유사한지 비교하여 번역에 대한 성능을 측정하는 방법이다.

BLEU Score는 3가지 지표로 측정되는데, 보통 n-gram(1~4)을 통한 순서쌍들이 얼마나 겹치는지 측정한 것이다.

1. **실험 결과**

****

****

위 결과들은 데이터셋에 있는 영문과 한글 5개를 무작위로 선택하여 OpenNMT로 한영/영한 번역을 한 것과 구글의 구글번역기와 네이버의 파파고를 비교하여 엑셀에 표로 만들어 보았다. 위에서 확인할 수 있듯이 한-영/영-한 번역이 대체로 뜻이 맞는 경우가 대부분임을 확인할 수 있다. 하지만 어느 부분에선 <unk>가 나오기도 하고 같은 단어가 반복하여 사용되기도 한다.

1. **분석 & Discussion**

**영-한**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Step** | **train** | **valid** | **test** | **BPE** | **BLEU** | **1-gram** | **2-gram** | **3-gram** | **4-gram** |
| **영-한** | **10만** | **127만** | **30만** | **3만** | **O** | **5.36** | **28.2** | **9.5** | **3.4** | **1.3** |
| **영-한** | **10만** | **154만** | **3만** | **3만** | **O** | **4.90** | **27.9** | **9.4** | **3.4** | **1.2** |
| **영-한** | **20만** | **96만** | **32만** | **32만** | **O** | **6.30** | **29.6** | **10.7** | **4.2** | **1.7** |
| **영-한** | **20만** | **96만** | **32만** | **32만** | **X** | **1.97** | **14.3** | **4.2** | **1.3** | **0.5** |
| **영-한** | **20만** | **127만** | **30만** | **3만** | **O** | **5.18** | **29.2** | **9.8** | **3.5** | **1.3** |
| **영-한** | **20만** | **154만** | **3만** | **3만** | **O** | **5.08** | **29.2** | **9.8** | **3.5** | **1.3** |

**한-영**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Step** | **train** | **valid** | **test** | **BPE** | **BLEU** | **1-gram** | **2-gram** | **3-gram** | **4-gram** |
| **한-영** | **5천** | **96만** | **32만** | **32만** | **O** | **1.46** | **12.0** | **2.5** | **0.7** | **0.2** |
| **한-영** | **5천** | **96만** | **32만** | **32만** | **X** | **1.26** | **12.7** | **2.2** | **0.5** | **0.2** |
| **한-영** | **2만** | **96만** | **32만** | **32만** | **O** | **7.82** | **26.4** | **10.9** | **5.2** | **2.5** |
| **한-영** | **2만** | **96만** | **32만** | **32만** | **X** | **5.25** | **20.8** | **7.6** | **3.3** | **1.5** |
| **한-영** | **10만** | **12만** | **4만** | **4만** | **O** | **13.97** | **36.9** | **18.3** | **10.0** | **5.6** |
| **한-영** | **10만** | **96만** | **32만** | **32만** | **O** | **19.07** | **47.7** | **24.4** | **13.8** | **8.2** |

위 표에서 볼 수 있듯이 영-한 기계번역보다는 한-영 기계번역의 BLEU 점수가 훨씬 높게 나온 것을 볼 수 있는데 이는 한국어 해석의 다양성으로 인한 것으로 판단된다. BLEU 점수를 높이기 위해 다양하게 시도를 해봤는데 첫 번째로 step수를 늘렸다. step수가 높아질수록 BLEU 점수가 올라간 것을 볼 수 있었고, 두 번째로 BPE의 유무를 달리했다. BPE가 있고 없고는 정말 큰 차이가 있음을 위 영-한 표에 색칠되어 있는 부분에서 명확히 확인할 수가 있다. 마지막으로 train 데이터셋, validation 데이터셋, test 데이터셋의 수를 96, 32, 32에서 154, 3, 3 그리고 127, 30, 3으로 바꿔가며 돌려봤는데 96, 32, 32에서 제일 큰 BLEU 점수가 나오는 것을 확인할 수 있었다.

1. **결론**

OpenNMT에 대한 정보를 나누는 공식 홈페이지 OpenNMT-forum을 참고해 본다면 독일어와 러시아어에 번역을 통해 잘 나왔다는 기준의 BLEU 점수는 대략 30 ~ 50 정도였다. 하지만 우리의 번역 결과는 한-영일 경우의 BLEU 점수는 19.07, 영-한일 경우의 BLEU 점수는 6.30이었다. 위의 OpenNMT를 사용한 다른 해외 이용자들과 비교하면 매우 낮은 수치라고 볼 수 있다. 하지만 4번의 실험결과 비교를 보듯이 다른 번역을 결과를 참고하여 비추어 볼 때 번역은 잘 되었다고 보여진다. 성능이 뛰어난 번역기인 구글 번역기와 파파고와 비교해도 좋은 번역이라고 보여진다. 하지만 근본적인 한글과 영어의 언어적 차이로 인하여 BLEU 점수가 낮게 나왔다고 보여진다. 이를 확인할 수 있는 부분은 1-gram에서의 점수와 2-gram에서의 차이 즉 동일한 단어의 연속일치 횟수가 다른 언어 번역에 비해 확연히 낮아진다는 것을 알 수 있다. OpenNMT-forum에서의 한 이용자의 Eng-Viet 번역에서의 결과는 BLEU = 24.20, 58.7/32.8/19.3/11.6로 출력되어지는데 1-gram에서 2-gram은 약 ½ 정도로 낮아지는데, 우리의 번역에서의 영-한에서의 1-gram과 2-gram사이의 차이는 약 ⅓ 으로 낮아지고, 한-영 번역에서의 평균적인 차이는 약 ¼ 정도로 낮아진다. 이는 아무래도 문법적 구조에서 나오는 차이라고 보여진다.

1. **참고 문헌**

데이터 셋 참조(한국정보화진흥원) - <https://www.aihub.or.kr/>

OpenNMT-py github - <https://github.com/OpenNMT/OpenNMT-py>

OpenNMT 공식홈페이지 - <https://opennmt.net/>

OpenNMT 포럼 - <https://forum.opennmt.net/>

BLEU 점수에 대한 참고사이트 - <https://m.blog.naver.com/bcj1210/221581980376>

<unk>에 대한 참고블로그 - <https://m.blog.naver.com/bcj1210/221581980376>

논문(병렬 코퍼스 필터링과 한국어에 최적화된 서브 워드 분절 기법을 이용한 기계번역) - <http://hiai.co.kr/wp-content/uploads/2019/12/%EB%85%BC%EB%AC%B8%EC%A6%9D%EB%B9%99_2019_06.pdf>

논문(공공 한영 병렬 말뭉치를 이용한 기계번역 성능 향상 연구) - <http://203.250.217.22/article/JAKO202018853211604.pdf>

Eng-Viet BLEU 결과 비교 - <https://forum.opennmt.net/t/bleu-score-of-word-features-model-is-lower-than-base-model/2934>