5.Machine Translation with Transformer

정규헌

Index

Machine Translation with the Transformer

1.Defining machine translation

Machine translation : reproducing human translation by machine transductions and outputs

1. Sentence to Translate
2. Learn Parameters : 단어들이 서로 어떤 관계를 갖고있는 지에 대해 수많은 파라미터로 학습
3. Machine Transductions : 새로운 문장에 대해 parameter를 적용
4. 적절한 Candidate translation을 선택함

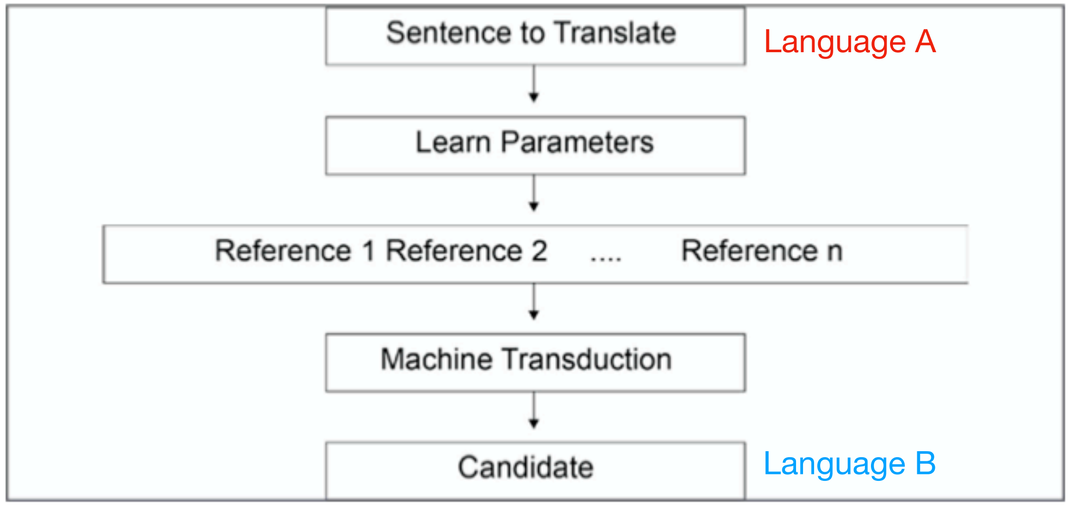

Figure 5.1: Machine translation process

2.transductions and translations

Human transductions and translations

- 인간 통역사
  - 문장을 단어 단위로 번역하지 않음
  - 단어간의 문맥이 무시되어 적절한 번역을 생성하지 못함
- Human transduction : A언어의 문장을 갖고와서 의미를 내포한 cognitive reference를 생성
- Reference 문장 : 통/번역가들에 의해 B언어로 번역한 것
  - 각각 다른 스타일의 번역가들이 A언어 문장을 번역한 여러 스타일의 B언어 문장이 존재
  - $Reference = \{ref\ 1, ref\ 2, ref\ 3...., ref\ n\}$
  - 기계도 인간의 방식으로 생각하는 방법을 찾음

Machine transductions and translations

- transformer Architecture
  1. Encoder stack
  2. Decoder stack
  3. Model’s Parameters
  4. Reference sequence : output sequence
- Evaluation
  - 여러 모델들의 solution을 비교하기 위해서는 동일한 데이터 셋을 사용해야함 - WMT dataset

3.WMT datasets

WMT'14 datasets

- WMT(Workshop on Machine Translation)
  - Vaswani et al.*(2017)* - transformer의 성능 측정에 WMT2014 en-ge, en-fr을 사용, 높은 BLEU를 얻음
  - Version 7 of European language corpus
  - https://www.statmt.org/europarl/v7/fr-en.tgz
    - europarl-v7.fr-en.en
    - europarl-v7.fr-en.fr

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

3.WMT datasets

process

- Tokenizer
  - 임베딩을 하기 위해서는 토큰화가 필요
  - 각 나라의 언어마다 토크나이징 하는 방법을 통일 할 수 는 없음
    - Spacy에서 언어에 따른 토크나이저 제공
- Field : 각 문장이 어떻게 전처리 할 것인지 정의
- Build Vocabulary
- Iterator
- Build Model : Attention is All You Need
- Train
- Test & Evaluation
- 참조 : bentrevett GitHub (https://github.com/bentrevett/pytorch-seq2seq/blob/master/6%20-%20Attention%20is%20All%20You%20Need.ipynb)

4. Evaluating machine translation

Why Bleu ?

Language Model evaluation metric

- Perplexity(PPL)
- 헷갈리는 정도. 자연스러운 문장을 만들어내는 가에 대한 수치
- 번역 성능을 직접적으로 반영하는 수치가 필요

BLEU : 기계 번역의 결과(candidate)와 사람의 번역(reference)이 얼마나 유사한지 비교해 성능을 측정하는 방법

1. n-gram을 통한 순서쌍들이 얼마나 겹치는지 측정(precision)
2. 문장 길이에 대한 과적합 보정
3. 같은 단어가 연속적으로 나올때 과적합 되는 것을 보정

4. Evaluating machine translation

Unigram Precision

Unigram Precision = Candidate의 단어 중에 Reference에 존재하는 단어의 수 / Candidate의 총단어수

Candidate : the the the the the the the
Ref : the cat is on the mat

Unigram Precision = 7/7 = 1

precision의 분자 계산을 위해 Ref와 매칭하여 카운트
Candidate의 unigram이 중복을 고려해야함

4. Evaluating machine translation

Modified Unigram Precision(Clipping)

Unigram Precision = Reference중 존재하는 Candidate의 단어수 / Candidate의 총단어수

Candidate : the the the the the the the the
Ref : the cat is on the mat

Unigram Precision = 7/7 = 1

$Count_{clip}$ = min(Count, Max_Ref_Count)
* max_ref_count : Unigram이 하나의 Ref에서의 최대 몇번 등장하는지 count

Modified Unigram Precision = $Count_{clip}$ / Candidate의 총 단어 수 = 2/7

Unigram Precision은 단어의 빈도수 접근 방식이므로 단어의 순서는 고려하지 않음

4. Evaluating machine translation

N-gram Precision

Candidate : the cat the cat on the mat
Ref : the cat is on the mat

2-gram Candidate	The cat	Cat the	The cat	Cat on	On the	The mat	Sum
Count	1	0	1	0	1	1	4

2-gram precision = 일치하는 2-gram의 수 / 모든 2-gram 쌍 = 4 / 6

Bleu에서는 n-gram(1~4) precision의 기하 평균을 냄

$$(\prod_{i=1}^{4} precision_i)^{\frac{1}{4}}$$

4. Evaluating machine translation

Brevity Penalty

Candidate : 빛이 쐬는 노인은 완벽한 어두운 곳에서 잠듬
Ref : 빛이 쐬는 노인은 완벽한 어둠에서 잠든 사람과 비교할 때 우울증이 심해질 가능성이 훨씬 높았다.

Candidate의 총 단어수가 작을 경우 precision이 높게 측정될 수 있으므로 보정이 필요

Brevity Penalty(BP)

$$min(1, \frac{\text{예측된 } sentence\text{의 길이(단어의 갯수)}}{true\ sentence\text{의 길이(단어의 갯수)}}) = min(1, \frac{6}{14}) = \frac{3}{7}$$

4. Evaluating machine translation

With BLEU

- **BLEU score**

$$\mathrm{BLEU}(N, T, R) = P(N, T, R) \times \mathrm{BP}(T, R)$$

$$BLEU = min(1, \frac{output\ length(\text{예측 문장})}{reference\ length(\text{실제 문장})})(\prod_{i=1}^{4} precision_i)^{\frac{1}{4}}$$

- Bleu score측정 방법
  - torchtext.data.metrics.bleu_score
  - nltk.translate.bleu_score
  - Sacrebleu

5. Applying a smoothing technique

A systematic Comparison of Smoothing Techniques for Sentence - Level BLEU

- Chen and Cherry et al.(ACL, 2014)
- **Label smoothing**의 원리와 같음
- 논문에서는 sentence-level의 BLEU에 적용할 수 있는 7개의 smoothing technique을 비교
  - 3개는 논문에서 새롭게 제안한 방법
  - 기존 방법에 비해 human judgment와 비교했을 때 더 좋은 상관관계를 가짐
- Tuning task에 대해서도 실험을 하지만 , 별다른 결과를 얻지 못함

	Into-English		
smooth	seg τ	sys γ	sys ρ
crp	–	0.720	0.887
0	0.165	0.759	0.887
1	0.224	0.760	0.887
2	0.226	0.757	0.887
3	0.224	0.760	0.887
4	0.228	0.763	0.887
5	0.234	0.765	0.887
6	0.230	0.754	0.887
7	**0.236**	**0.766**	0.887

Table 2: Correlations with human judgment on WMT data for Into-English task. Results are averaged on 4 test sets. "crp" is the origianl IBM corpus-level BLEU.

	Out-of-English		
smooth	seg τ	sys γ	sys ρ
crp	–	0.712	0.744
0	0.119	0.715	0.744
1	0.178	0.722	**0.748**
2	0.180	0.725	0.744
3	0.178	0.724	0.744
4	0.181	0.727	0.744
5	0.184	0.731	0.744
6	0.182	0.725	0.744
7	**0.187**	**0.734**	0.744

Table 3: Correlations with human judgment on WMT data for Out-of-English task. Results are averaged on 4 test sets. "crp" is the origianl IBM corpus-level BLEU.

6.Machine Translation

- 번역 서비스의 가장 중요한 목표 : **번역 품질의 향상**
- 번역 task에서 고려해야할 여러가지 문제와 연구들
    1. 고유 명사(ex. 이하늘, 유연석) : Data Augmentation
    2. 문맥을 고려한 번역 : 기존 학습 데이터 (원문 : 번역문) -> 새로운 학습 데이터(원문+컨텍스트문 : 번역문)
    3. 문체 : 한국어의 경우 예사/높임말의 구분이 존재
    4. 데이터의 부족 : 영어 데이터는 많은 편이지만 다른 나라 끼리의 번역 데이터는 부족함, 영어로 연결
    5. 문화 : 각 언어에는 해당 국가나 민족의 특성 및 문화가 반영
        - 한국어의 경우 일본어와 유사, 영어의 경우 서유럽권의 언어들과 유사
        - 문법의 차이 : 태국어의 경우 띄어쓰기가 존재하지 않고, 문장 단위를 명시적으로 표시하지 않음
            - 한국어의 경우 논리적으로 맞으면 네, 틀리면 아니오
            - 영어의 경우 긍정이면 yes, 부정이면 no

Reference

- Transformers for Natural Language
- Bentrevett github(https://github.com/bentrevett)
- BLEU: a Method for Automatic Evaluation of Machine Translation(2002)
- Chen&Cherry et al.(2014)
- Vaswani et al.(2017)
- WMT
- Pytorch 공식문서 : https://tutorials.pytorch.kr/beginner/transformer_tutorial.html
- Kakao enterprise : https://tech.kakaoenterprise.com/22?category=909203

감사합니다!