MASS : MAsked Sequence to Sequence Pre-training for Language Generation

Kaitao Song et al.

ICML 2019

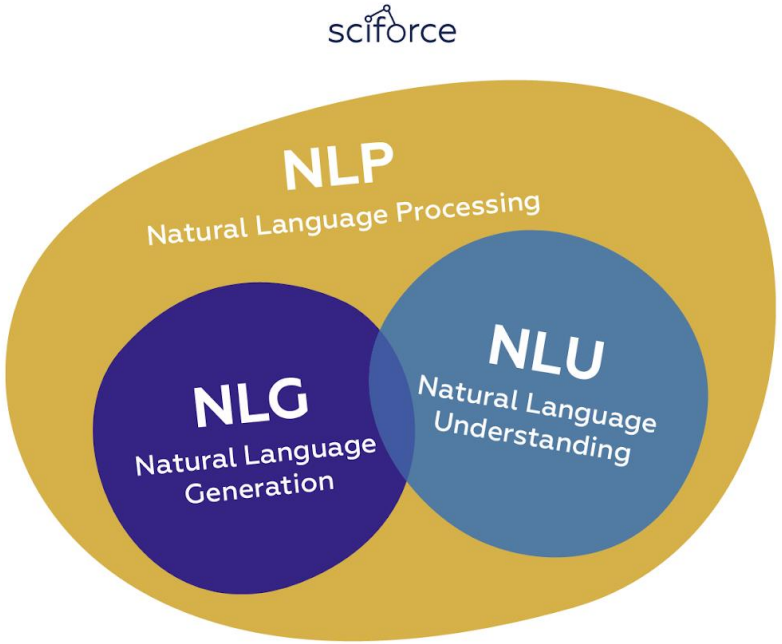
Introduction

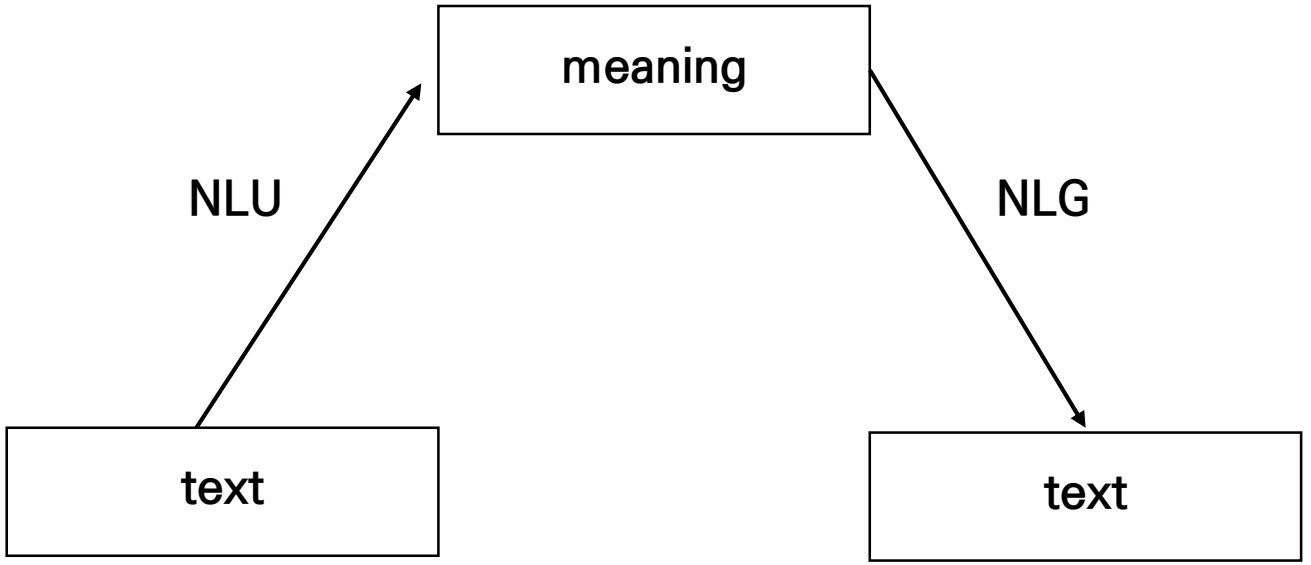
Introduction

- 최근 BERT가 등장해 좋은 성능을 냈음에도 불구하고, BERT는 원래 Natural Language Understanding task를 위해 만들어진 모델
- 보통 BERT-like method들은 encoder 또는 decoder 하나만을 사용하기 때문에 encoder와 decoder 둘 모두가 중요한 Natural Language Generation task에 곧바로 적용하기 어려움
- 그러므로 **Natural Language Generation task를 위한** pre-training method를 만들 필요가 있음
  - NLG의 특성: 일반적으로 데이터가 부족하고 많은 경우 학습 데이터가 low-resource 또는 zero-source
- MASS는 인코더와 인코더를 **jointly train**하는 방식으로 NLG task에 대응

Language Understanding & Language Generation

- NLU: 작성된 텍스트의 의미를 이해 (text to structured data)
  - named-entity recognition, question answering, sentiment analysis…
- NLG: 특정한 입력을 조건으로 텍스트 생성 (structured data to text)
  - machine translation, conversational response generation…





Related Works

Related Works

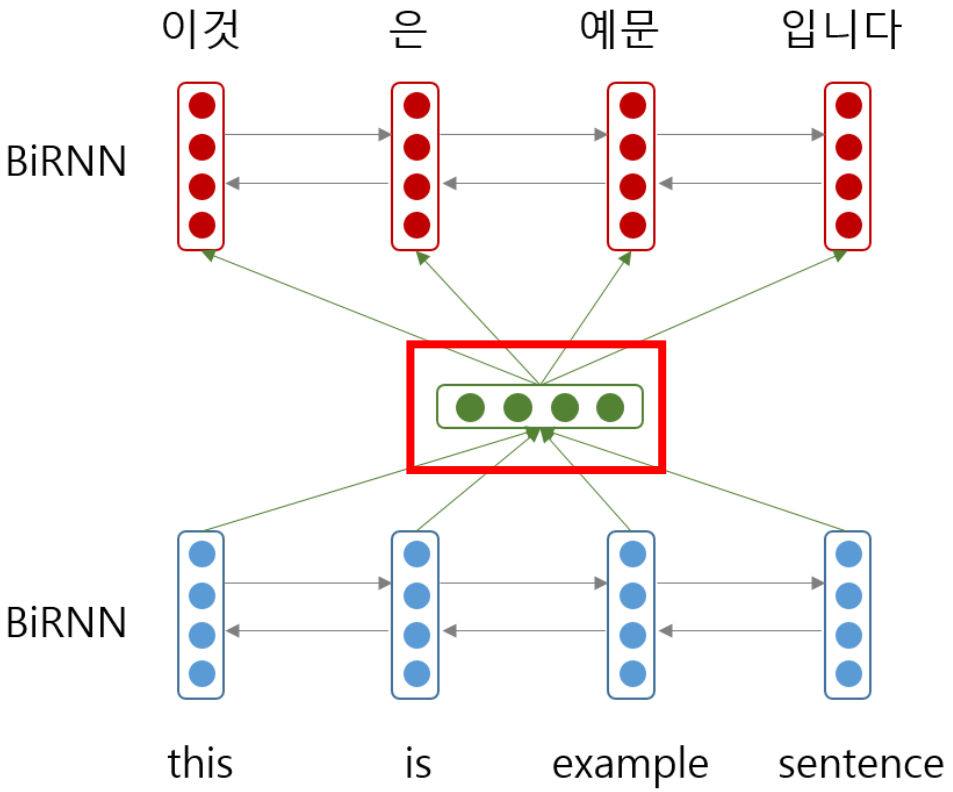
1. Sequence to Sequence Learning (encoder-decoder framework)
2. Pre-training
   - GPT (**only decoder**)
   - BERT (**only encoder**)
   - XLM (encoder and decoder, **직전 SOTA**)

- 기존 모델들은 인코더와 디코더가 따로 학습되기 때문에 최적의 성능을 낼 수 없음

→ 인코더와 디코더를 **unlabeled data**만으로 **joint training**한다는 아이디어

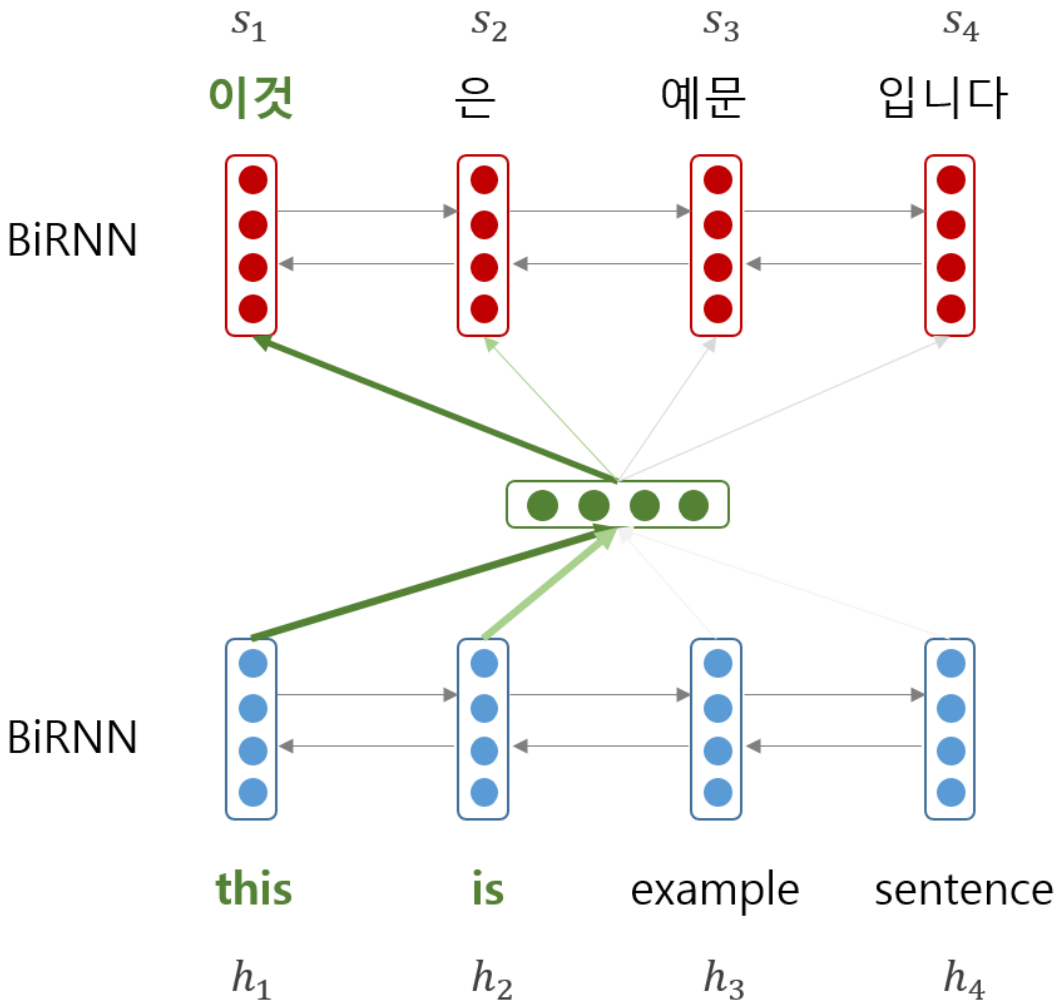
Related Works - Sequence to Sequence Learning

- **encoder**: source sentence를 읽어 알맞은 representation 생성
- **decoder**: 인코더가 생성한 representation과 앞서 등장했던 token들을 이용해 target token이 등장할 확률 계산
- 초창기에는 input sequence의 정보를 하나의 fixed-length vector에 저장했음
  - 문장의 모든 정보를 고정 벡터 하나에 밀어넣어야 한다는 문제점 (long-term dependency)
  - 각 토큰을 예측할 때마다 필요한 정보가 다른데 항상 같은 벡터를 참고한 한다는 문제점



Related Works - Sequence to Sequence Learning (Attention)

- Fixed-length의 벡터가 아닌 인코더의 모든 hidden state 참조
- 각 토큰을 생성할 때마다 다른 정보를 참조할 수 있음
- source representation 중에 어떤 부분의 정보에 더 **집중**할 것인가를 나타냄
- Translation의 경우: 이 단어가 target language의 어떤 단어와 **가장 비슷**한가



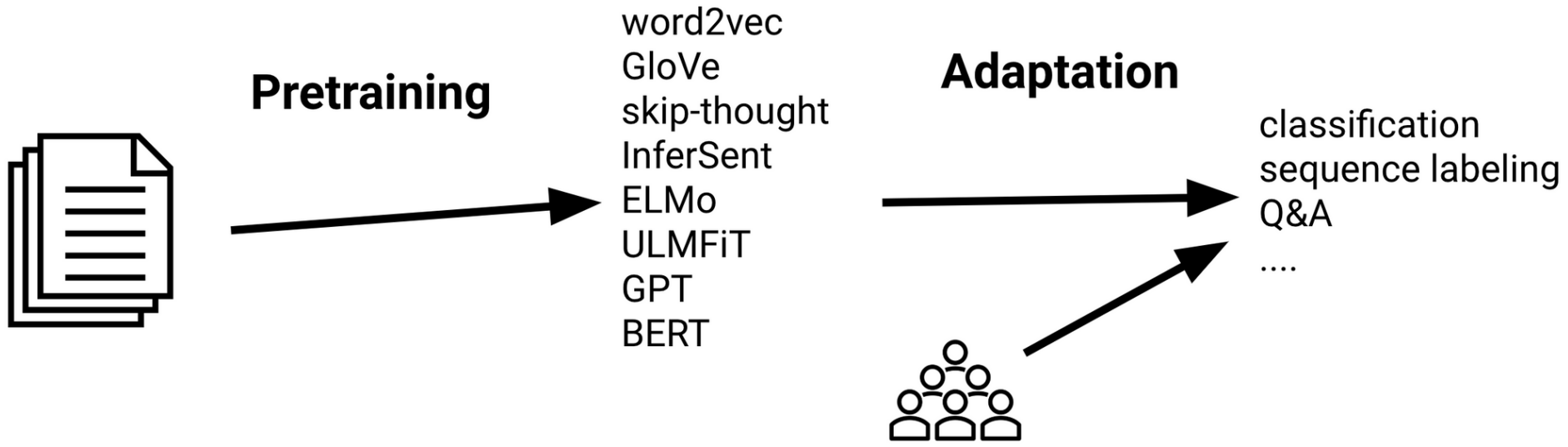
Related Works – Pretraining

- machine translation task는 원래 supervised
  - Parallel corpora 필요
  - {“나는 학생입니다” : “I am a student”}
- 그러나 parallel corpora를 구하기 어렵고 구축 비용이 많이 듦
- Low-resource language의 경우 parallel corpora 자체가 적거나 없을 수 있음 (영어-프랑스어 vs 한국어-네팔어)
- 반면 monolingual data는 찾기 쉽고 많이 있음

→ monolingual data만으로 모델을 먼저 훈련시키고 그 다음 번역(downstream task)을 위한 학습을 한다면?

Related Works – Pretraining

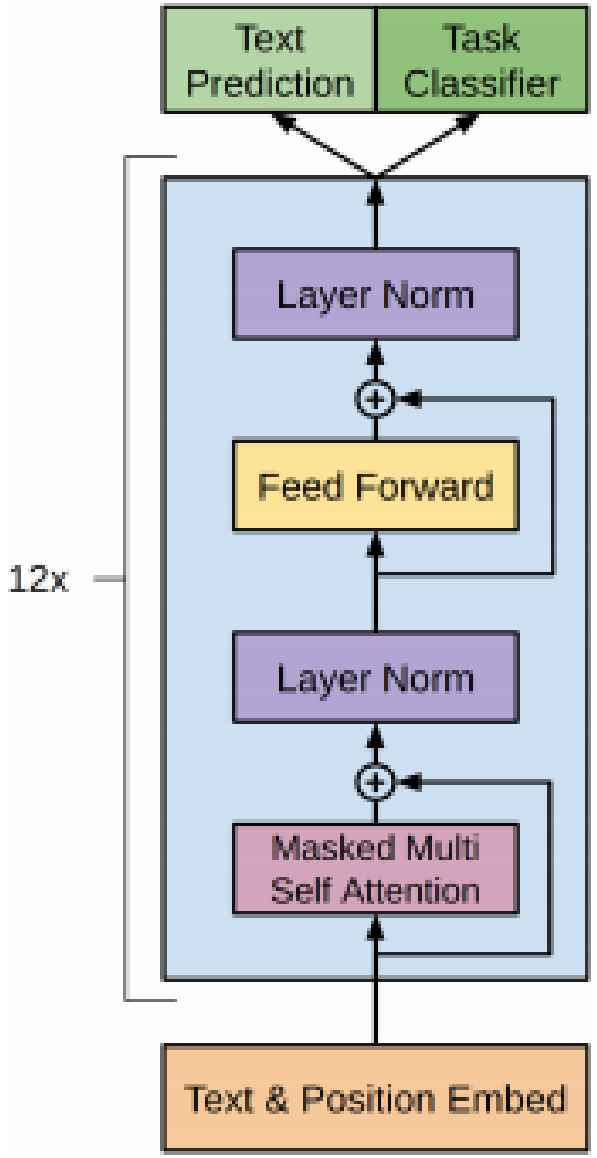
- 어떤 문제를 위해 학습한 모델을 다른 문제를 푸는 데 재사용하는 것
- general language modeling task로 모델을 학습시킨 후 각 downstream task에 맞게 업데이트
- 모델은 이전 학습에서 얻은 지식을 downstream task에서 활용할 수 있음



Related Works – GPT: Standard Language Model

- 문장에서 이전 단어들이 주어졌을 때 다음 단어가 나올 확률 계산 (standard language model)
- Transformer **decoder** 사용
- “I ate a delicious hot ___”

→ 빈칸에 들어가는 단어가 ‘dog’일 것을 예측

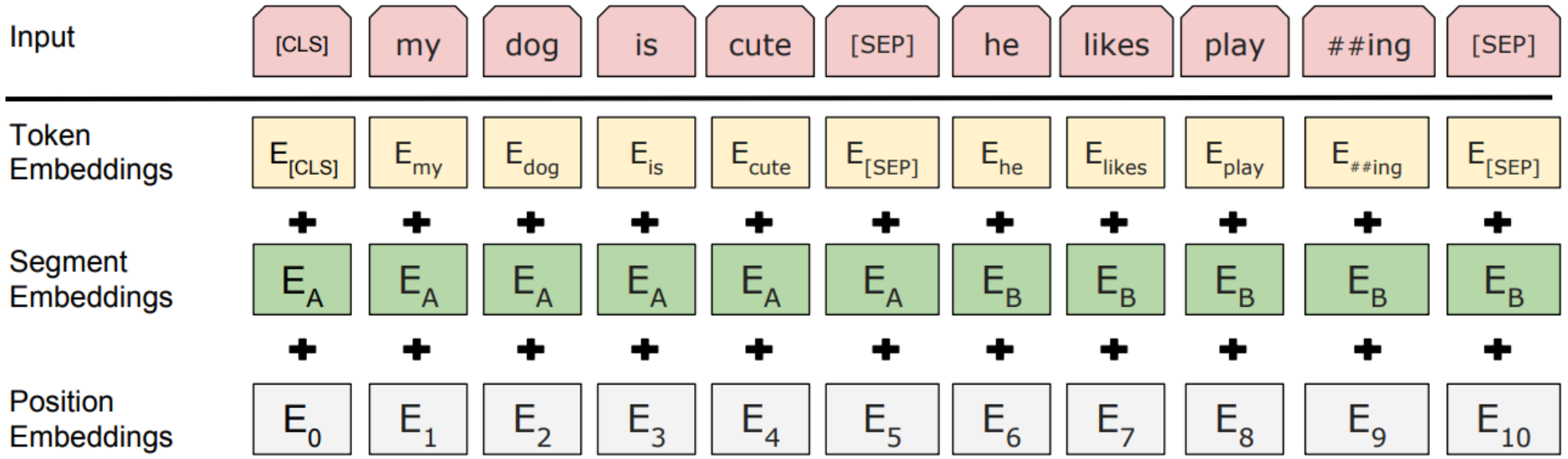


Related Works – BERT: Masked Language Model

- Transformer **encoder** 사용
- 문장에서 가려진 임의의 단어 예측 (masked language model)
- I [MASK] a delicious hot dog

→ [MASK] 자리에 들어갈 단어가 'ate'일 것을 예측

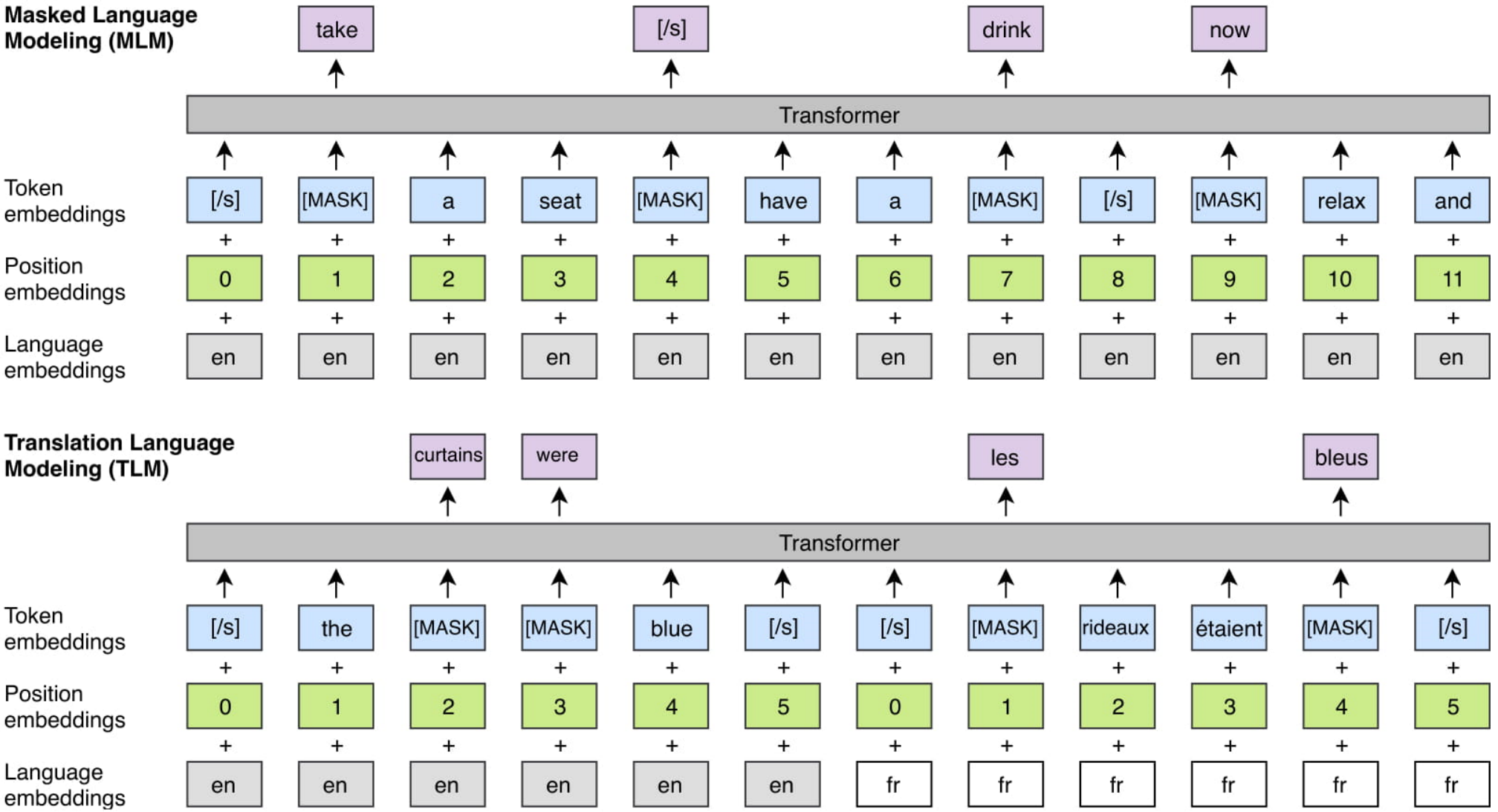
- GPT와 달리 bidirectional 함: 예측해야 할 토큰의 이전에 나온 토큰과 이후에 나온 토큰을 모두 활용하여 예측 수행



Related Works - XLM: Cross Lingual Language Model

1. Causal Language Model (CLM)
   - Monolingual
   - 이전 단어들로부터 다음 단어 예측
   - Transformer 사용
2. Masked Language Model (MLM)
   - Monolingual
   - BERT의 sentence pair 대신 임의 개수의 text stream 사용
3. **Translation Language Model** (TLM)
   - **Cross-lingual**
   - **MLM의 확장**
   - parallel sentence를 concatenate하여 학습
   - Ex) 모델은 mask된 영어 토큰을 예측하기 위해 프랑스어 정보를 살필 수 있음

Related Works – XLM



MASS

MASS – method

- **MA**sked **S**equence to **S**equence
    1. Masking 사용
    2. Sequence to Sequence 구조

- Input에서 **k개**의 토큰을 임의로 마스킹
- 마스킹된 토큰을 디코더가 예측
- 인코더에서 마스킹되지 않은 토큰이 디코더에서 마스킹

MASS – objective function

마스킹된 fragment u부터 v까지가 마스킹된 문장

$$L(\theta;\mathcal{X}) = \frac{1}{|\mathcal{X}|}\Sigma_{x\in\mathcal{X}} \log P(x^{u:v}|x^{\backslash u:v};\theta)$$

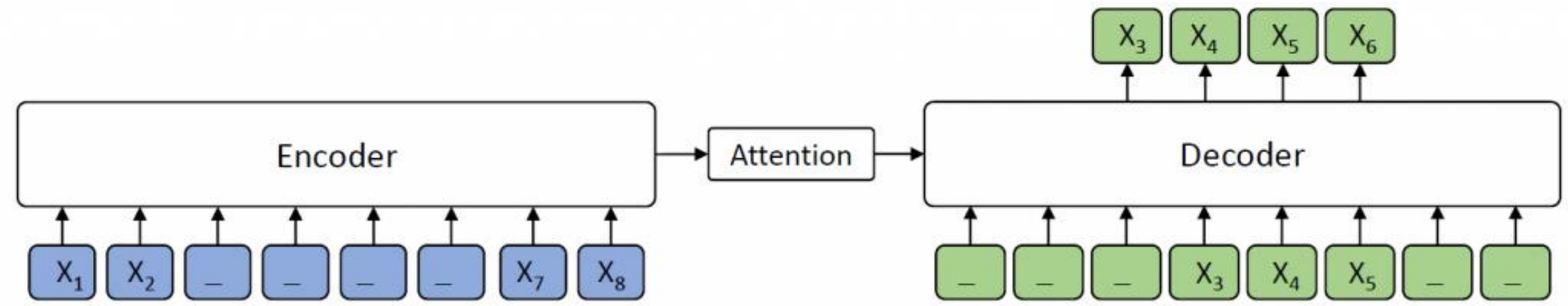
$$= \frac{1}{|\mathcal{X}|}\Sigma_{x\in\mathcal{X}} \log \prod_{t=u}^{v} P(x_t^{u:v}|x_{<t}^{u:v}, x^{\backslash u:v};\theta). \quad (1)$$

- X: source domain
- x: source sentence
- Y: target domain
- y: target sentence

- m: input sequence의 길이
- u: 마스킹의 시작점
- v: 마스킹의 끝점

- $x^{u:v}$: 문장 x에서 u부터 v까지의 부분
- $x^{\backslash u:v}$: u부터 v까지가 마스킹된 문장 x

MASS – method



- _ 로 표현된 부분이 마스킹을 나타냄
- 모델이 예측하는 것: input에서 마스킹된 x3, x4, x5, x6
- Input에서 마스킹되지 않았던 position 1–3, 7–8이 디코더에서 마스킹

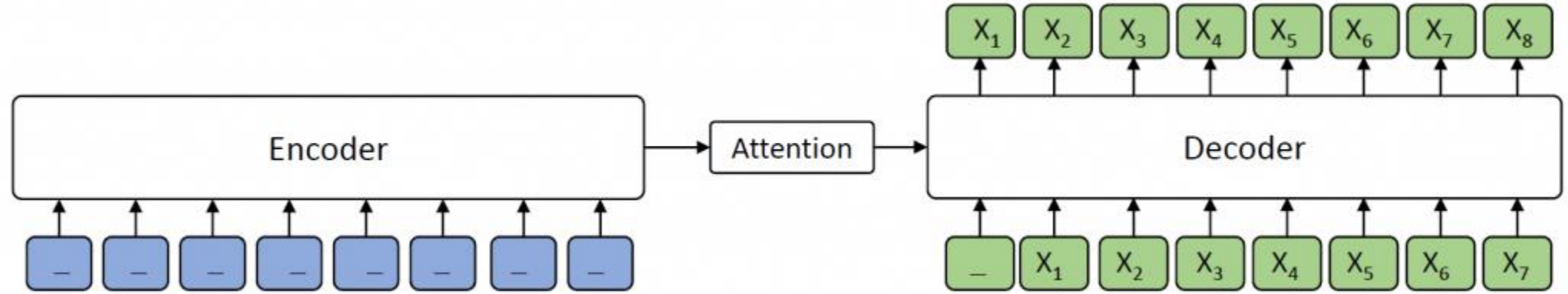
MASS – special cases

Length	Probability	Model
$k = 1$	$P(x^u \mid x^{\setminus u}; \theta)$	masked LM in BERT
$k = m$	$P(x^{1:m} \mid x^{\setminus 1:m}; \theta)$	standard LM in GPT
$k \in (1, m)$	$P(x^{u:v} \mid x^{\setminus u:v}; \theta)$	methods in between

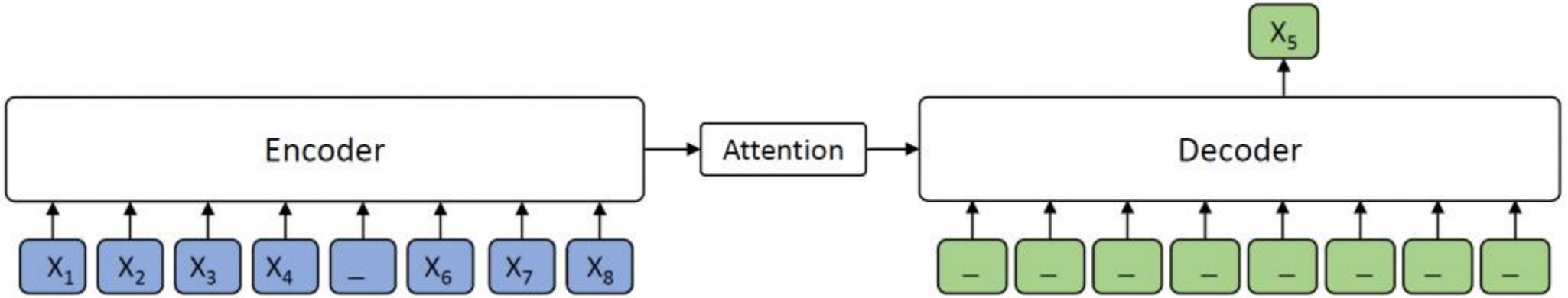
- Hyperparameter k에 따라 GPT와 BERT를 MASS의 한 사례로 여길 수 있음
- k = 1: 인코더 입력 중 한 토큰이 마스킹 → BERT
- k = m: 인코더 입력 전체가 마스킹 → GPT

→ MASS를 general pre-training framework로 확장할 수 있다

MASS – special cases



K=m, GPT



K=1, BERT

MASS – about BERT

- One may argue that:
  1. BERT와는 모델 구조가 조금 다르다
  2. BERT에서는 한번에 한 토큰만 마스킹하는 것이 아니라 여러 개의 토큰을 마스킹한다

- However:
  1. 디코더의 모든 토큰이 마스킹되기 때문에 디코더 자체가 non-linear classifier로 기능함
     → BERT의 softmax matrix와 유사해지며, conditional probability가 BERT와 같음
  2. masking language modeling의 가장 중요한 아이디어는 마스킹을 통해 bidirectional information을 얻는 것이며, 여러 토큰을 마스킹하는 것은 거의 학습 속도 향상을 위함임

MASS – advantages

- GPT의 standard language modeling과 BERT의 masked language modeling 둘 다 인코더와 디코더를 따로따로 학습하고 있음

  → 일반적으로 encoder-decoder framework를 이용하는 language generation task들에 적합하지 않음

- MAS는 **language generation task**를 위해 인코더와 디코더를 **jointly pre-train**
  - 디코더는 previous 토큰에 의지하지 않고 인코더가 제공한 representation과 attention 정보만 참고해서 마스킹된 토큰을 예측해야 함
  - 인코더는 마스킹되지 않은 나머지 토큰들을 표현해야 함 → language understanding 능력 상승
  - 디코더는 연속적으로 마스킹된 토큰(sentence fragment, 즉 discrete하지 않음)을 예측하기 때문에 language modeling 능력 상승

Experiments and Results

Experiments and Results – configuration

- Basic structure: Transformer
  - Encoder: 6 layer
  - Decoder: 6 layer
  - Hidden size: 1024
  - Feed-forward filter size: 4096
- Task:
  - unsupervised machine translation (cross-lingual)
  - low-resource machine translation (cross-lingual)
  - text summarization (monolingual)
  - conversational response generation (monolingual)

Experiments and Results – Pretraining

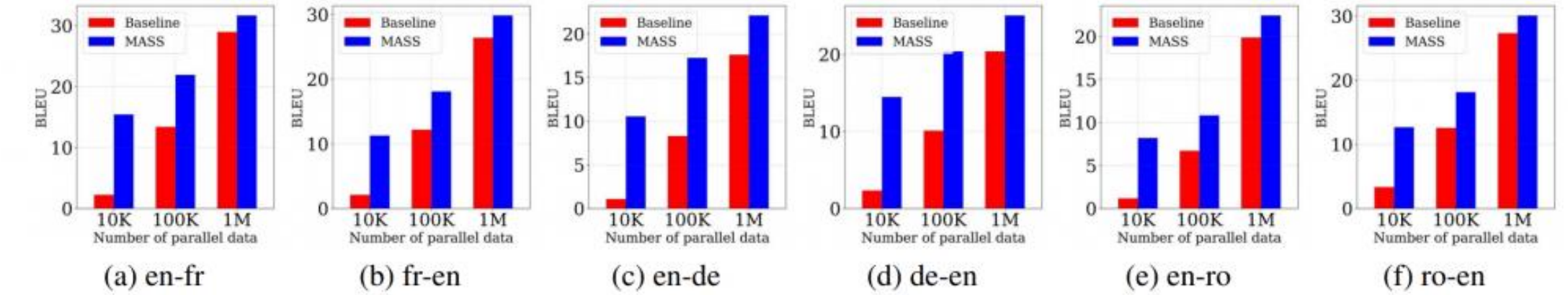
- For neural machine translation task:
  - Pair: English–French, English–German, and English–Romanian(low–resource)
  - Source와 target language를 구분하기 위해 각 토큰에 language embedding 추가 (XLM 기반)
  - Dataset: WMT News Crawl datasets, News Crawl dataset augmented with WMT16 data
  - Byte Pair Encoding: source, target language 사이의 subword unit 학습
- Masking: BERT를 따름
  - 마스킹의 80% – [M]
  - 마스킹의 10% – random token
  - 마스킹의 10% – unchanged
- 마스킹 토큰 개수 k: 문장 길이의 약 절반 (k에 관한 ablation 있음)
- Adam optimizer
  - Learning rate 10^–4
  - 8 NVIDIA V100 GPU
  - Batch: 3000 tokens

Experiments and Results – Unsupervised machine translation

Method	Setting	en - fr	fr - en	en - de	de - en	en - ro	ro - en
Artetxe et al. (2017)	2-layer RNN	15.13	15.56	6.89	10.16	-	-
Lample et al. (2017)	3-layer RNN	15.05	14.31	9.75	13.33	-	-
Yang et al. (2018)	4-layer Transformer	16.97	15.58	10.86	14.62	-	-
Lample et al. (2018)	4-layer Transformer	25.14	24.18	17.16	21.00	21.18	19.44
XLM (Lample & Conneau, 2019)	6-layer Transformer	33.40	33.30	27.00	34.30	33.30	31.80
MASS	6-layer Transformer	**37.50**	**34.90**	**28.30**	**35.20**	**35.20**	**33.10**

- Lample et al. (2018) : pre-training 없음
- XLM : pre-training 있음, 직전 SOTA 모델
- 한 모델을 English-English, French-French로 학습한 후 back translation 기법으로 pseudo-bilingual data를 생성하여 fine-tuning 진행
- 두 언어를 구별하기 위해 language embedding 추가
- Cross-lingual 정보는 Byte Pair Encoding 안에 들어 있음

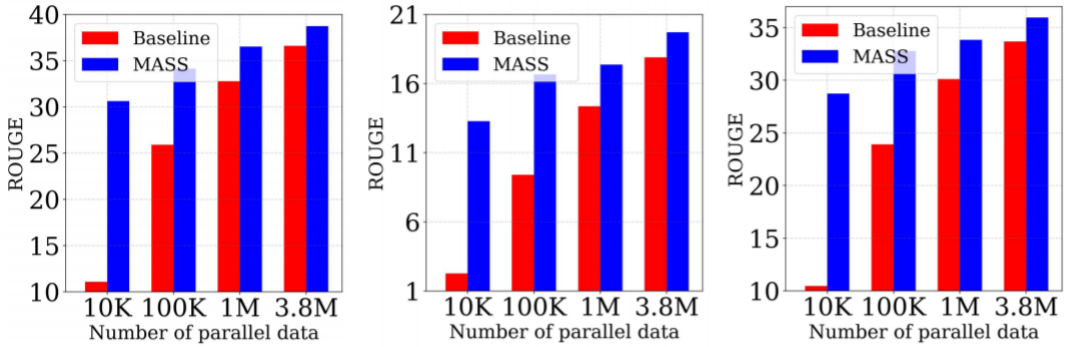
Experiments and Results - Low resource machine translation



- Low-resource machine translation: bilingual training data가 적은 machine translation task
- WMT14 English-French, WMT16 English-German, English-Romanian 데이터에서 parallel sentence 의 수를 10K, 100K, 1M로 늘려 가면서 low-resource 시나리오를 시뮬레이션

Experiments and Results – Text Summarization

Method	RG-1 (F)	RG-2 (F)	RG-L (F)
BERT+LM	37.75	18.45	34.85
DAE	35.97	17.17	33.14
MASS	**38.73**	**19.71**	**35.96**



(a) RG-1 (F) (b) RG-2 (F) (c) RG-L (F)

- Text Summarization: 긴 문서의 짧은 요약을 생성하는 task
- pre-trained 모델을 각각 다른 스케일((10K, 100K, 1M, 3.8M)의 Gigaword corpus로 fine-tune
- Encoder input: article
- Decoder input: title
- Baseline 1: BERT+LM (인코더: BERT / 디코더: language model로 pretrained)
- Baseline 2: DAE (Denoising Auto-Encoder)

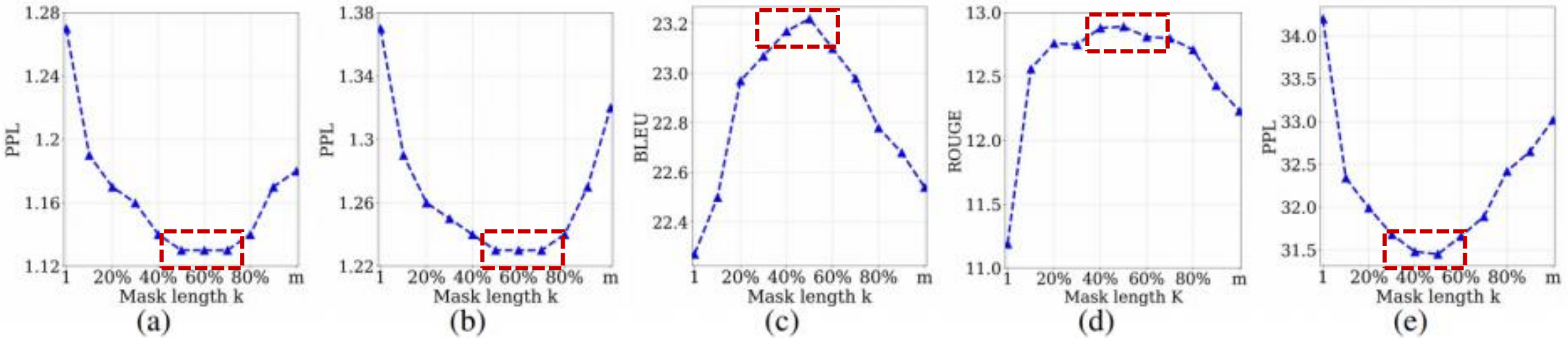
Experiments and Results – Conversational Response Generation

	Method	Data = 10K	Data = 110K
DAE	*Baseline*	82.39	26.38
	BERT+LM	80.11	24.84
	MASS	**74.32**	**23.52**

- Data: Cornell movie dialog corpus (140K sentence pairs)
- Hyperparameters: pretraining 때와 동일
- 기준: perplexity (낮을수록 좋음)

Ablation

Ablation – Study of Different K



- 마스킹되는 토큰의 개수 k를 바꾸어 실험함으로써 k의 영향을 확인
- (a): English-English pretrained 모델의 perplexity
- (b): French-French pretrained 모델의 perplexity
- (c): English-French translation task의 BLEU 스코어
- (d): text summarization task의 ROUGE 스코어
- (e): conversational response generation task의 perplexity

Ablation – Study of Different K

- 각 downstream task들에 대해, k가 문장 길이 m의 약 50%일 때 가장 좋은 성능
- m의 50% k는 인코더와 디코더 사이의 좋은 균형
- 인코더나 디코더 어느 한 쪽의 valid token이 너무 적다면 모델이 인코더나 디코더 한쪽에 의존하게 됨

→ joint training의 의미가 줄어들고, 인코더가 유의미한 representation을 생성하거나 디코더가 알맞은 문장을 생성하기 어려워짐

Ablation – discrete tokens instead of consecutive

Method	BLEU	Method	BLEU	Method	BLEU
Discrete	36.9	*Feed*	35.3	MASS	37.5

Table 6. The comparison between MASS and the ablation methods in terms of BLEU score on the unsupervised en-fr translation.

- MASS의 중요한 설계 중 하나는 **연속적**인 토큰을 마스킹하고 예측한다는 것
- Discrete: 연속적이지 않은 discrete token을 마스킹하여 학습한 것
- Discrete가 MASS보다 성능이 낮기 때문에 consecutive masking의 효과를 알 수 있음

Ablation – masking on decoder side

Method	BLEU	Method	BLEU	Method	BLEU
Discrete	36.9	*Feed*	35.3	MASS	37.5

Table 6. The comparison between MASS and the ablation methods in terms of BLEU score on the unsupervised en-fr translation.

- 인코더에서 마스킹되지 않은 토큰을 디코더에서 마스킹하는 것은 디코더가 이전에 등장한 토큰에 의존하는 대신 인코더에서 유용한 정보를 추출하게 하기 위함 (joint training)
- Feed: 디코더 입력에서 마스킹 없이 모든 토큰을 입력으로 제공하여 학습한 모델
- Feed가 MASS보다 성능이 낮기 때문에 디코더 부분에서의 마스킹이 중요함을 알 수 있음

Conclusion

doubt

- 실험에 사용된 English-French, English-German, English-Romanian 모두 같은 알파벳을 사용하며 관계가 가까운 유럽 언어
- MASS의 cross-lingual information은 Byte Pair Encoding (shared vocabulary) 안에 묵시적으로 포함되어 있음
- 영어-한국어, 영어-미얀마어, 영어-네팔어 언어 조합에서도 이런 방식이 통할까? (그럴 것 같지 않음)

Conclusion

- Language generation task에서는 인코더와 디코더를 함께 학습하는 것이 중요
- MASS는 encoder-decoder framework를 이용하여 문장의 일부분을 다른 부분으로부터 재구축하는 모델
- 모델 하나를 pre-train 후 여러 language generation task에 대해 fine-tuning할 수 있으며, 여러 task에서 좋은 성능을 기록
- 특히 unsupervised NMT에서 state-of-the-art를 경신했으며, low-resource NMT에서도 효과를 보임

Thank you!

Appendix - BLEU score

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

- BLEU: Bilingual Evaluation Understudy
  1. Precision : 실제 번역(reference)과 prediction이 얼마나 겹치는가
  2. Clipping : 예측된 문장에 중복된 단어가 있을 경우 보정
  3. Brevity Penalty : 예측된 문장 길이가 너무 짧을 경우 보정

Appendix - back translation

- Target sentence를 source sentence로 번역하고, 그 결과를 training data에 더하는 것
- 두 언어 간의 양방향 번역을 joint training
- Monolingual data로부터 pseudo-parallel data를 생성할 수 있음
- 인코더와 디코더가 **noisy translation으로부터 본래의 문장을 생성**하도록 학습

  → DAE 대신 사용할 수 있음