

# Back Translation

Ki Hyun Kim

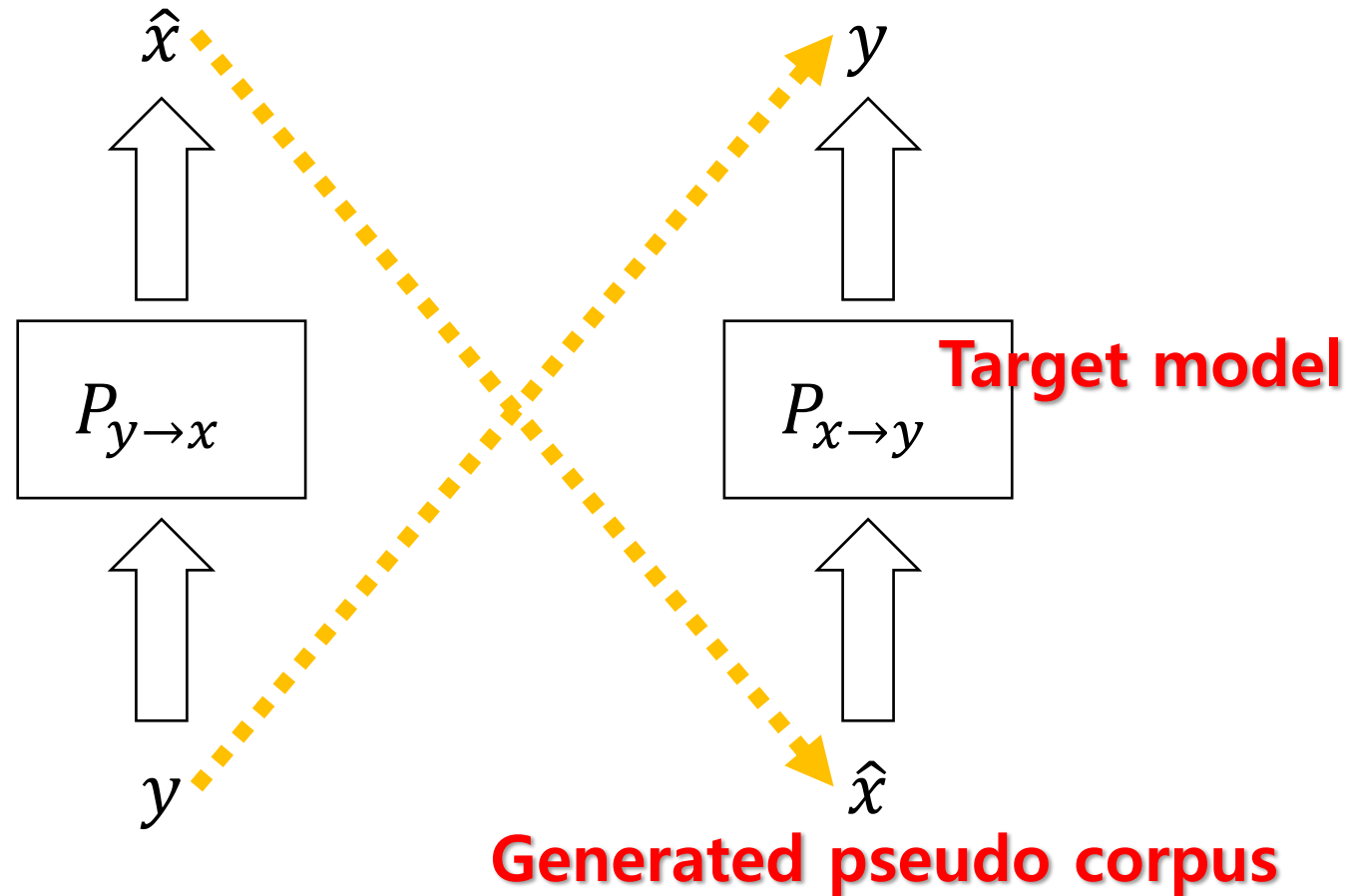
[nlp.with.deep.learning@gmail.com](mailto:nlp.with.deep.learning@gmail.com)

# Motivations

- 제한된 parallel corpus로 인해 decoder의 generation 성능이 떨어짐
  - 제한된 corpus로 인해 decoder의 언어모델 성능이 떨어짐
- 풍부한 monolingual corpus를 통해 추가로 decoder를 학습시켜 보자!

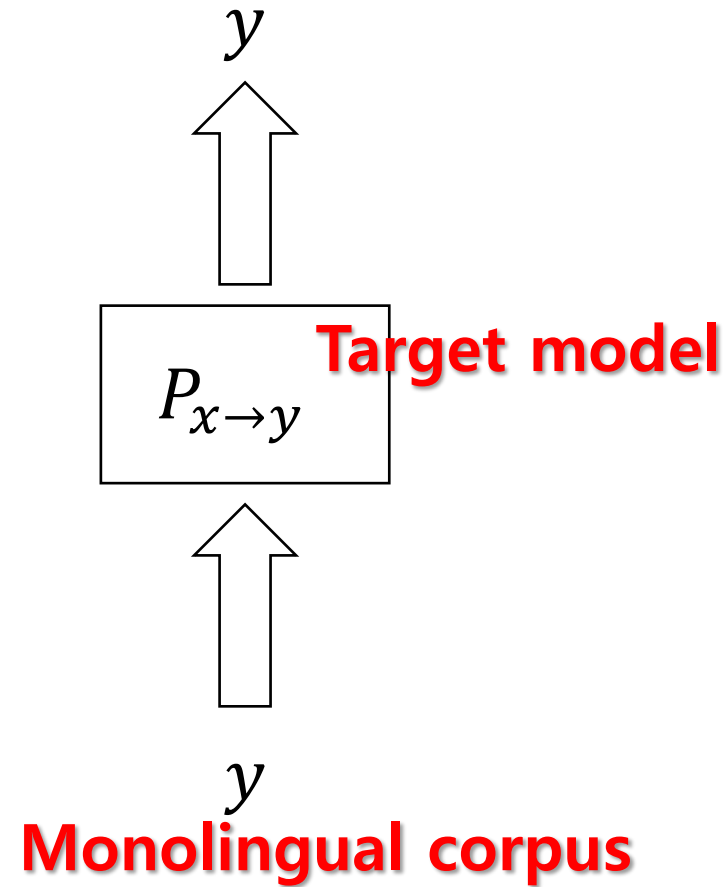
# Back Translation

- 보통 번역은 두 개의 모델이 동시에 나오기 마련
  - 반대쪽 모델을 활용하여 synthetic corpus를 만들 수 있음



# Copied Translation

- 그냥  $y$ 를 똑같이 입력에 넣어주자



# Equations

$$\mathcal{B} = \{x_n, y_n\}_{n=1}^N$$
$$\mathcal{M} = \{y_s\}_{s=1}^S$$

$$\hat{\mathcal{M}} = \{\hat{x}_s, y_s\}_{s=1}^S,$$

where  $\hat{x}_s = \operatorname{argmax}_{x \in \mathcal{X}} \log P(x|y_s; \theta_{y \rightarrow x})$ .

- Back Translation

$$\mathcal{L}(\theta_{x \rightarrow y}) = - \sum_{i=1}^N \log P(y_i | x_i; \theta_{x \rightarrow y}) - \sum_{s=1}^S \log P(y_s | \hat{x}_s; \theta_{x \rightarrow y})$$

- Copied Translation

$$\mathcal{L}(\theta_{x \rightarrow y}) = - \sum_{i=1}^N \log P(y_i | x_i; \theta_{x \rightarrow y}) - \sum_{s=1}^S \log P(y_s | y_s; \theta_{x \rightarrow y})$$

# Back and Copied Translation

## Back Translation

- Synthetic corpus를 만들어 반대로 넣어주어 학습
- Pros:
  - 매우 단순하고 직관적
- Cons:
  - 합성 데이터가 너무 많을 경우, 성능이 저하됨

## Copied Translation

- Encoder와 Decoder에 같은 데이터를 넣어 학습
- Pros:
  - Back translation 보다 더욱 단순함
- Cons:
  - 마찬가지로 너무 많이 넣을 경우 성능저하
  - <한/영>과 같이 character가 다를 경우, vocab이 너무 커짐

# LM Ensemble vs Back Translation

## LM Ensemble

- Pros:
  - 모든 monolingual corpus 활용 가능
- Cons:
  - 별도의 모델 학습 필요
  - 하이퍼 파라미터 다량 추가

## Back Translation

- Pros:
  - 매우 쉽고 직관적, 간단함
- Cons:
  - 제한적인 monolingual corpus 활용 가능

# Noised beam Back Translation [Edunov et al., 2019]

- Problems in previous method:
  - Model로부터 biased된 synthetic sentence가 생성됨
  - Synthetic source sentence로부터 잘못된 encoding을 학습
- Solution:
  - Noise를 섞어 (다양하게) 생성한 문장을 synthetic source sentence로 활용한다.
- 하지만 여전히 성능 개선의 여지가 남아있음



# Tagged Back Translation [Caswell et al., 2018]

- Conditional Sequence to Sequence를 응용
  - Encoder의 입력에 BT 여부를 표시
  - 따라서 encoder는 합성 코퍼스에 대해서는 다르게 동작하도록 학습
- 이를 통해, pseudo corpus의 비율에 상관없이 학습 가능

# Tagged Back Translation

a. Results on 24M BT Set

Model	AVG 13-18	2010	2011	2012	2013	2014	2015	2016	2017	2018
Bitext	32.05	24.8	22.6	23.2	26.8	28.5	31.1	34.7	29.1	42.1
BT	33.12	24.7	22.6	23.5	26.8	30.8	30.9	36.1	30.6	43.5
NoisedBT	34.70	26.2	<b>23.7</b>	<b>24.7</b>	<b>28.5</b>	31.3	33.1	37.7	<b>31.7</b>	<b>45.9</b>
TaggedBT	<b>34.83</b>	<b>26.4</b>	23.6	24.5	28.1	<b>32.1</b>	<b>33.4</b>	<b>37.8</b>	<b>31.7</b>	<b>45.9</b>

b. Results on 216M BT Set

Model	AVG 13-18	2010	2011	2012	2013	2014	2015	2016	2017	2018
<a href="#">Edunov et al. (2018)</a>	35.28			25.0	<b>29.0</b>	<b>33.8</b>	34.4	37.5	<b>32.4</b>	44.6
NoisedBT	35.17	<b>26.7</b>	24.0	<b>25.2</b>	28.6	32.6	33.9	38.0	32.2	45.7
TaggedBT	<b>35.42</b>	26.5	<b>24.2</b>	<b>25.2</b>	28.7	32.8	<b>34.5</b>	<b>38.1</b>	<b>32.4</b>	<b>46.0</b>

# Summary

- Monolingual corpus를 활용한 기계번역 성능 향상은 매우 흥미로운 주제
  - 각 source / target 언어 corpus를 활용하여 encoder / decoder를 학습시키고자함
- 다양한 알고리즘(from RL to Dual learning)들이 제시되었지만,  
Back Translation이 가장 직관적이고 간단하며 인기있는 방법
  - Pseudo corpus를 활용하여 디코더 언어모델의 성능을 향상시키는 것이 주 목적