

# Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation

---



논문리뷰  
이도연

# 목차

---

**1 Introduction**

**2 SMT (Statistical Machine Translation)**

**3 Seq2Seq (RNN Encoder-Decoder)**

**4 GRU (Gated Recurrent Unit)**

**5 Experiments**

**6 Conclusion**

# 1. Introduction

---

**Applying RNN Encoder-Decoder with a novel hidden unit to SMT system**

- 두 개의 RNN 구조를 사용한 Encoder, Decoder → Seq2Seq
- 게이트가 추가된 RNN, LSTM → GRU 제안
- SMT에 적용해 성능을 높임
- 더 나아가 완전히 대체할 수 있는 가능성

## 2. SMT (Statistical Machine Translation)

### 통계적 기계 번역

Translation model - 번역모델 (Source, Target)

$$p(f | e) \propto p(e | f)p(f)$$

Language model - 언어모델 (얼마나 자연스러운 단어 순서인지, 다음에 출현하는 단어의 확률)

다양한 feature들의 확률을 선형 조합하여 제일 높은 확률을 갖는 문장으로 번역

$$\log p(f | e) = \sum_{n=1}^N w_n f_n(f, e) + \log Z(e)$$

$$\operatorname{argmax} \log p(\text{영|한}) = \sum_{n=1}^N w_n f_n(\text{영|한})$$

〈N-gram feature 예시〉

$$f_1 = P(i \text{ eat} | \text{나는 먹는})p(\text{like to} | \text{것을 좋아})$$

$$f_2 = P(i | \text{나는})p(\text{eat} | \text{먹는})p(\text{to} | \text{것을})p(\text{like} | \text{좋아})$$

·  
·

이 논문에서는 SMT에서 feature로 활용(문장의 확률을 도출)할 수 있는 아키텍처 RNN Encoder-Decoder 제시  
Statistical Machine Translation -> Neural Machine Translation

### 3. Seq2Seq (RNN Encoder-Decoder)

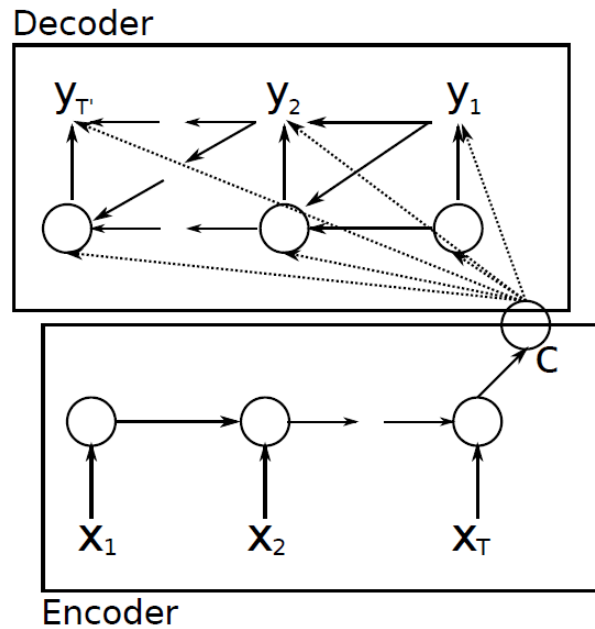


Figure 1: An illustration of the proposed RNN Encoder-Decoder.

#### Encoder 역할

- 가변적인 길이의 sequence를 고정된 크기의 벡터로 압축
- input sequence  $x$ 를 순차적으로 읽어
- 마지막에 input sequence 전체를 압축한  $c$  (context vector)

#### Decoder 역할

- 고정된 크기의 벡터를 다시 가변적인 길이의 sequence로 생성

$$h_{\langle t \rangle} = f(h_{\langle t-1 \rangle}, y_{t-1}, c)$$

$$P(y_t | y_{t-1}, y_{t-2}, \dots, y_1, c) = g(h_{\langle t \rangle}, y_{t-1}, c)$$

- $c$ 를 활용해 output(target) sequence

### 3. Seq2Seq (RNN Encoder-Decoder)

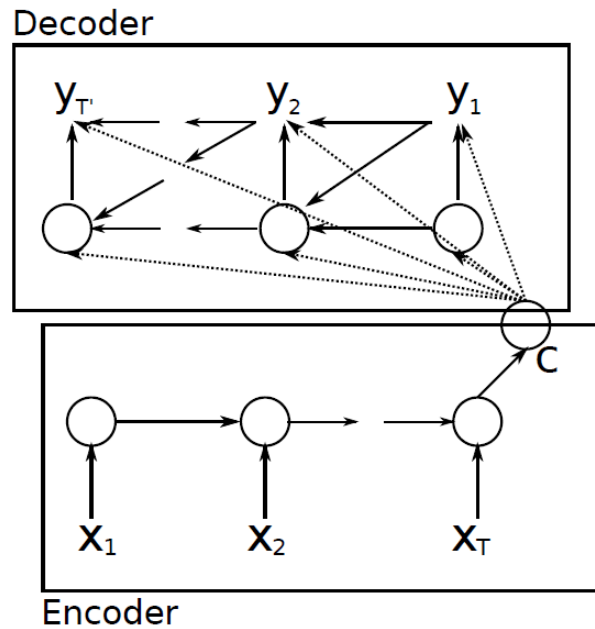


Figure 1: An illustration of the proposed RNN Encoder-Decoder.

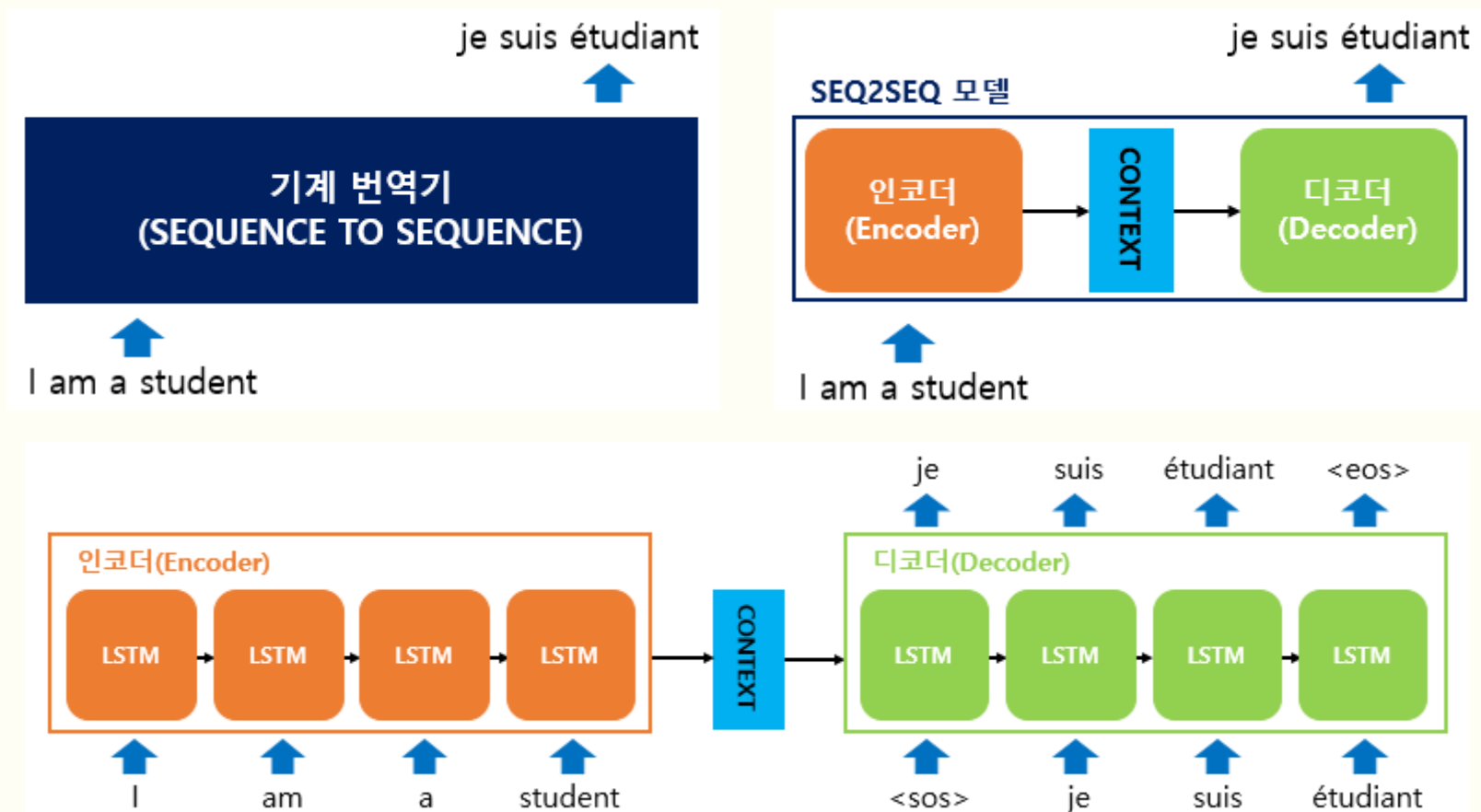
Maximize the conditional log-likelihood

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^N \log p_{\theta}(y_n | \mathbf{x}_n)$$

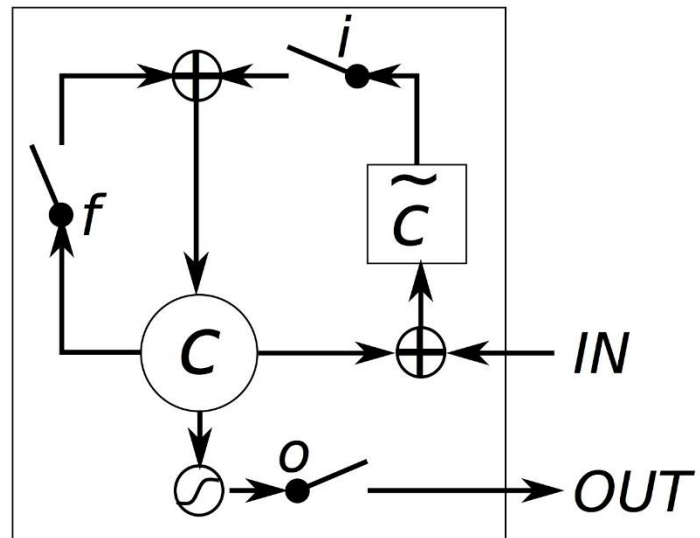
학습 후에는,

- input sequence가 주어졌을 때 target sequence 생성
- input, output sequence에 대한 score(probability)

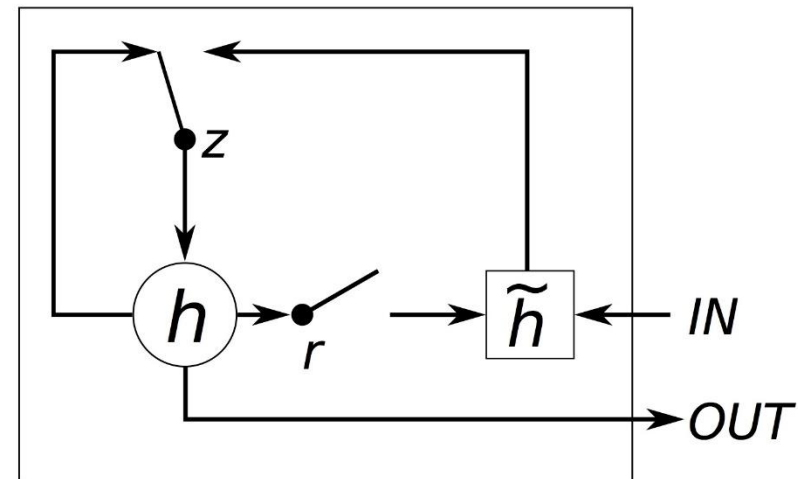
### 3. Seq2Seq (RNN Encoder-Decoder)



## 4. GRU (Gated Recurrent Unit)



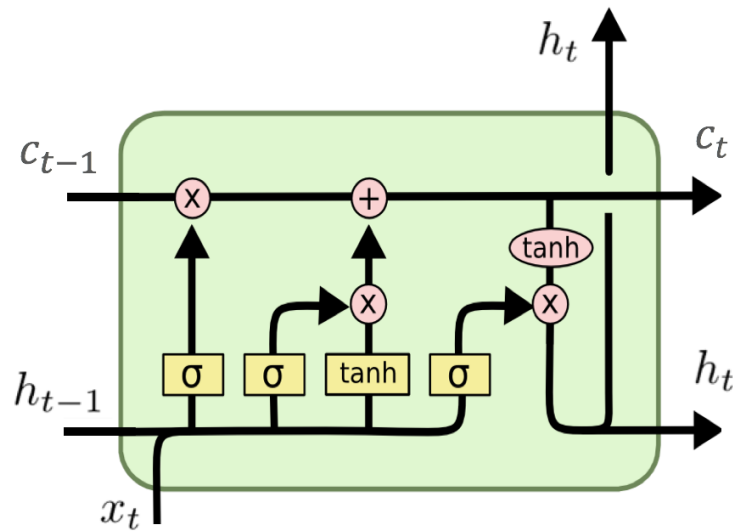
(a) Long Short-Term Memory



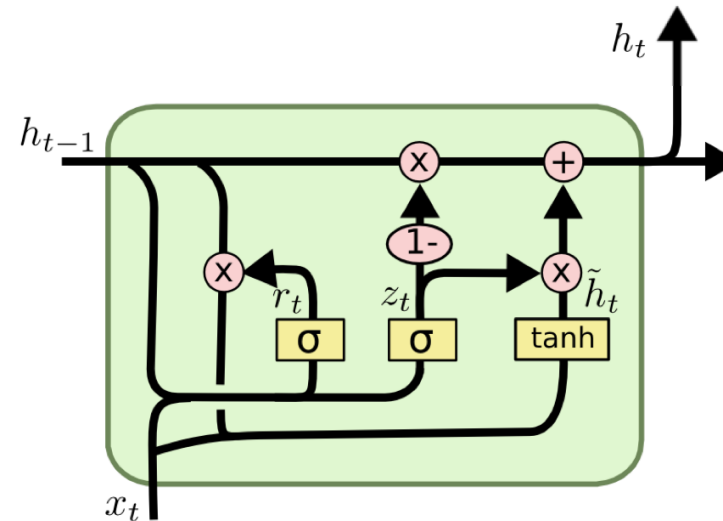
(b) Gated Recurrent Unit



## 4. GRU (Gated Recurrent Unit)



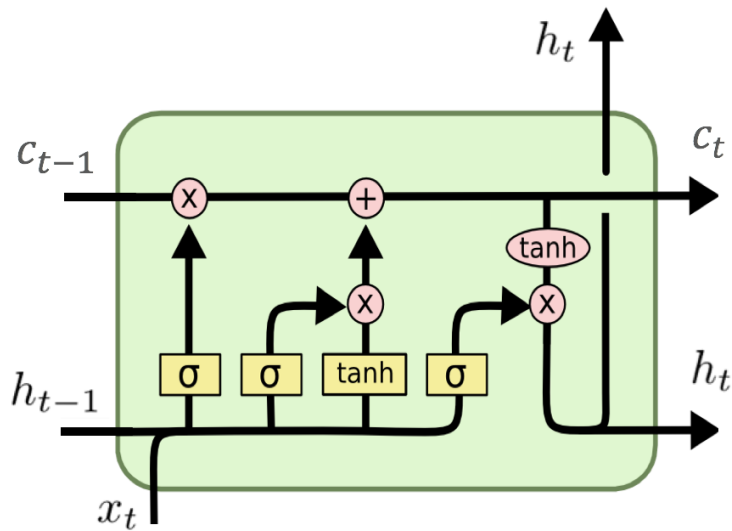
LSTM  
(Long-Short Term Memory)



GRU  
(Gated Recurrent Unit)

## 4. GRU (Gated Recurrent Unit)

### LSTM



LSTM  
(Long-Short Term Memory)

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Forget 게이트

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

Input 게이트

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

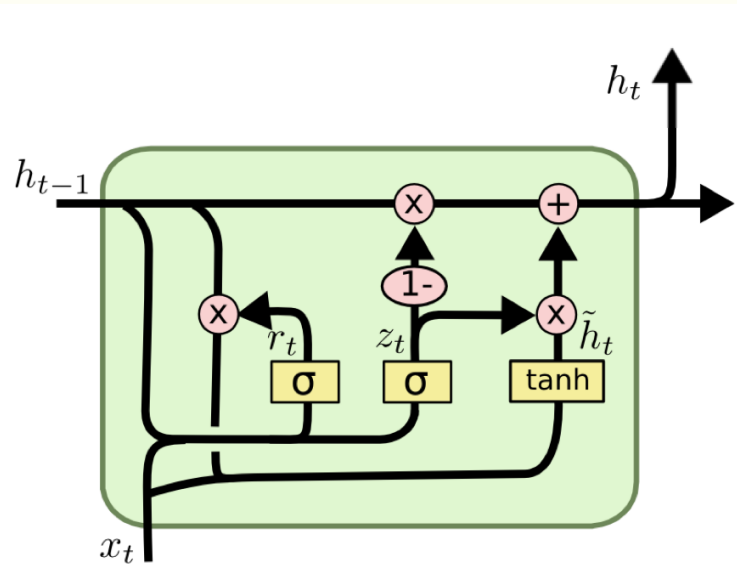
$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

Output 게이트

$$h_t = o_t * \tanh(C_t)$$

## 4. GRU (Gated Recurrent Unit)

GRU - LSTM보다 간단한 구조, 매개변수를 줄여 계산시간 단축



GRU  
(Gated Recurrent Unit)

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

Update 게이트

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

Reset 게이트

$$\tilde{h}_t = \tanh(W \cdot [r_t \cdot h_{t-1}, x_t])$$

$$h_t = \underbrace{(1 - z_t)}_{\text{Forget 게이트 역할}} \cdot h_{t-1} + \underbrace{z_t \cdot \tilde{h}_t}_{\text{Input 게이트 역할}}$$

Forget 게이트 역할

Input 게이트 역할

# 5. Experiments

## BLEU (Bilingual Evaluation Understudy) - 기계 번역의 성능 측정

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

- 1) 단어 개수 카운트로 측정하기
- 2) 중복을 제거하여 보정하기
- 3) 순서를 고려하기 위해서 n-gram으로 확장하기
- 4) 짧은 문장 길이에 대한 패널티

값이 클수록 성능이 더 좋고, (perplexity는 값이 작을수록 성능이 좋음)  
언어에 구애받지 않고 사용할 수 있으며,  
계산 속도가 빠르다.

# 5. Experiments

---

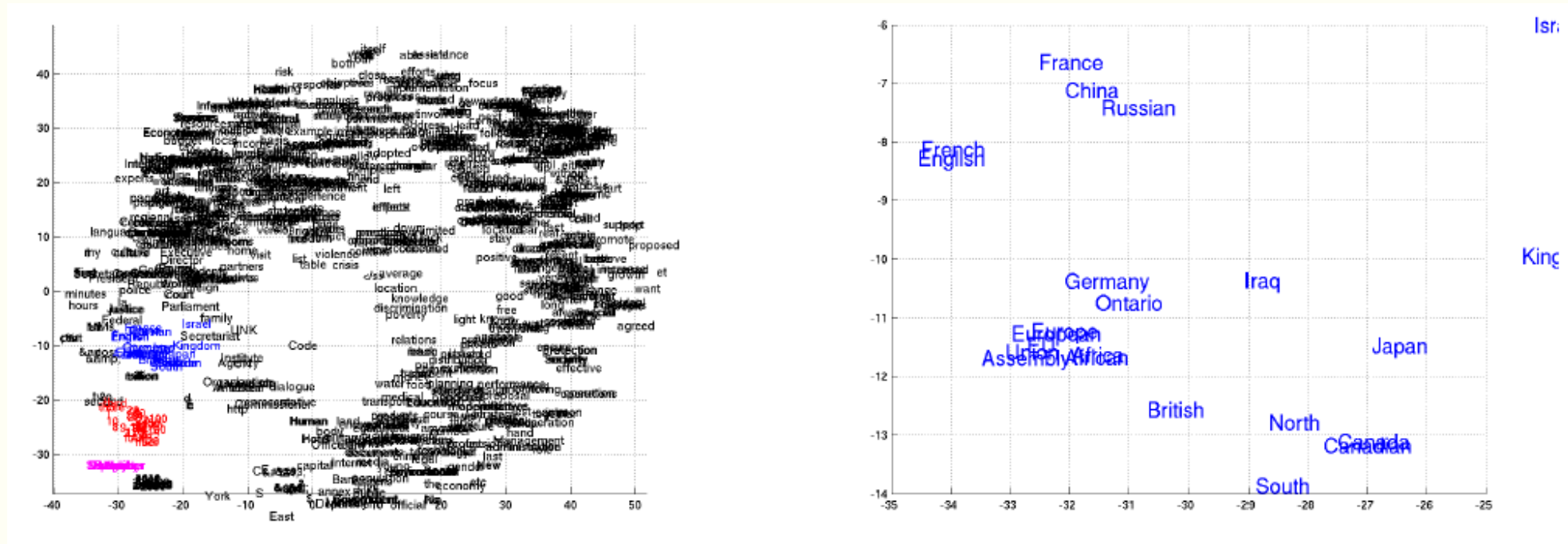
Task(Dataset) - WMT'14 영어 -> 프랑스어 번역

Models	BLEU	
	dev	test
Baseline	30.64	33.30
RNN	31.20	33.87
CSLM + RNN	31.48	34.64
CSLM + RNN + WP	31.50	34.54

RNN(Seq2Seq)을 추가한 모델의 성능이 향상되었음

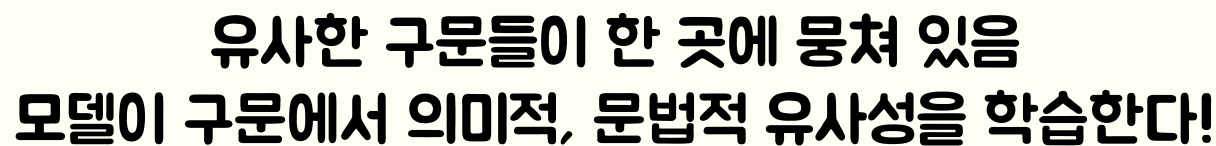
# 5. Experiments

## Word Representation



유사한 단어가 비슷한 위치에 뭉쳐 있음  
모델이 단어 내의 의미적 유사성을 학습한다!

## Phrase Representation



## 6. Conclusion

---

- 임의의 길이를 가진 Sequence에 대한 Neural Network Architecture 제시
  - > RNN Encoder-Decoder, Seq2Seq의 초기 형태를 제안
    - 1) score a pair of sequences
    - 2) generate a target sequence given a source sequence
- 새로운 hidden unit (reset gate, update gate) GRU 제안
- SMT에서 각 phrase pair의 score를 구하는데 RNN Encoder-Decoder 사용해 성능을 높임
- 더 나아가 완전히 대체할 수 있는 가능성 -> Neural Machine Translation



# Q&A

---

# 참고

---

[https://www.stechstar.com/user/zbxe/study\\_SQL/54663](https://www.stechstar.com/user/zbxe/study_SQL/54663) LSTM, GRU 이미지

<https://wikidocs.net/24996> Seq2Seq 이미지

<https://wikidocs.net/31695> BLEU Score

<https://www.youtube.com/watch?v=T-w0XOROpPY> 논문 리뷰