seq2seq

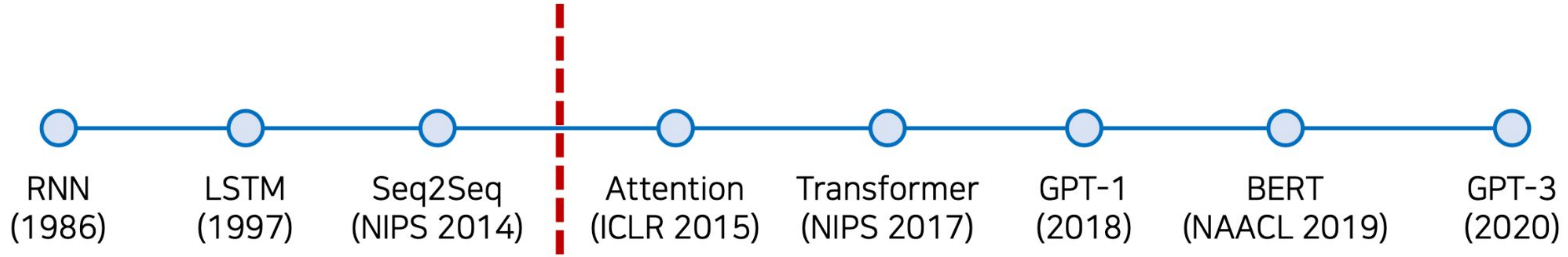
기계번역의 돌파구

SMARCLE

목차

01

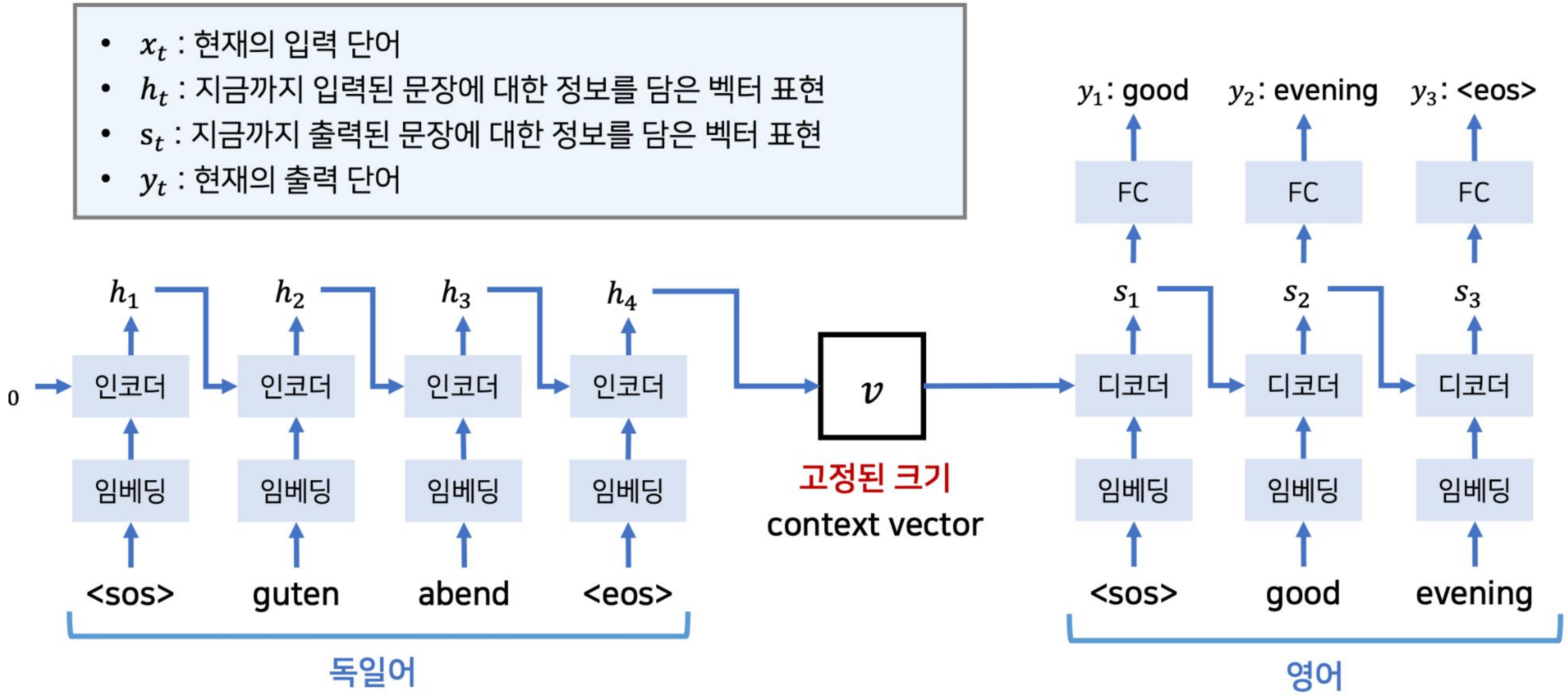
배경



출처 : https://github.com/ndb796/Deep-Learning-Paper-Review-and-Practice

02

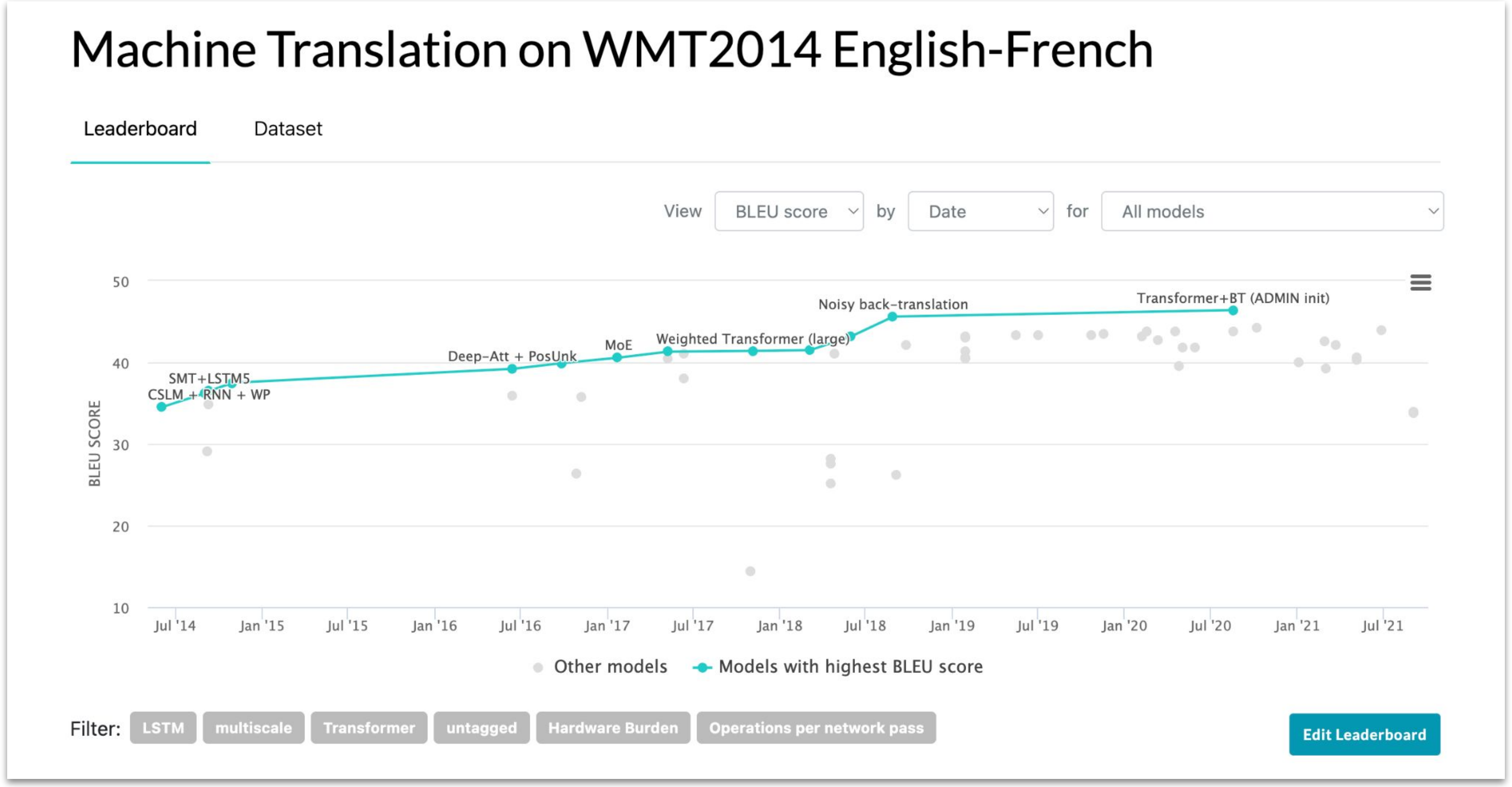
모델구조



출처 : https://github.com/ndb796/Deep-Learning-Paper-Review-and-Practice

03

성능



https://paperswithcode.com/sota/machine-translation-on-wmt2014-english-french

Method	test BLEU score (ntst14)
Bahdanau et al. [2]	28.45
Baseline System [29]	33.30
Single forward LSTM, beam size 12	26.17
Single reversed LSTM, beam size 12	30.59
Ensemble of 5 reversed LSTMs, beam size 1	33.00
Ensemble of 2 reversed LSTMs, beam size 12	33.27
Ensemble of 5 reversed LSTMs, beam size 2	34.50
Ensemble of 5 reversed LSTMs, beam size 12	**34.81**

$$\text{Unigram Precision} = \frac{\text{Ref들 중에서 존재하는 Ca의 단어의 수}}{\text{Ca의 총 단어 수}} = \frac{\text{the number of Ca words(unigrams) which occur in any Ref}}{\text{the total number of words in the Ca}}$$

04

코드 실습

Open in Colab

Sequence to Sequence Learning with Neural Networks (NIPS 2014) 실습

- 본 코드는 기본적으로 **Seq2Seq** 논문의 내용을 따릅니다.
  - 본 논문은 **딥러닝 기반의 자연어 처리** 기법의 기본적인 구성을 이해하고 공부하는 데에 도움을 줍니다.
  - 2020년 기준 가장 뛰어난 번역 모델은 Seq2Seq가 아닌 **Transformer 기반의 모델**입니다.
- 코드 실행 전에 **[런타임]** → **[런타임 유형 변경]** → 유형을 **GPU**로 설정합니다.

데이터 전처리(Preprocessing)

- **spaCy 라이브러리**: 문장의 토큰화(tokenization), 태깅(tagging) 등의 전처리 기능을 위한 라이브러리
  - 영어(Engilsh)와 독일어(Deutsch) 전처리 모듈 설치

```
%%capture
!python -m spacy download en
!python -m spacy download de
```

```
import spacy

spacy_en = spacy.load('en') # 영어 토큰화(tokenization)
spacy_de = spacy.load('de') # 독일어 토큰화(tokenization)
```

```
# 간단히 토큰화(tokenization) 기능 써보기
tokenized = spacy_en.tokenizer("I am a graduate student.")

for i, token in enumerate(tokenized):
    print(f"인덱스 {i}: {token.text}")
```

```
인덱스 0: I
인덱스 1: am
인덱스 2: a
인덱스 3: graduate
인덱스 4: student
```

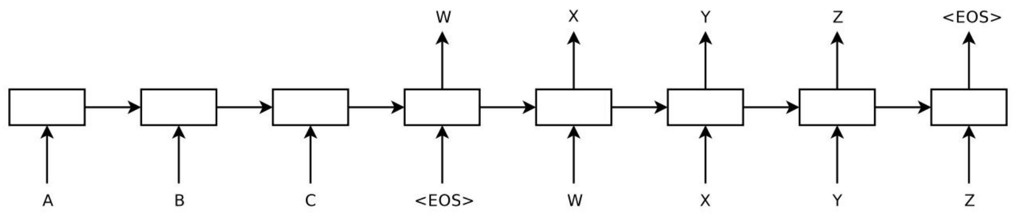
https://github.com/ndb796/Deep-Learning-Paper-Review-and-Practice/blob/master/code_practices/Sequence_to_Sequence_with_LSTM_Tutorial.ipynb

05

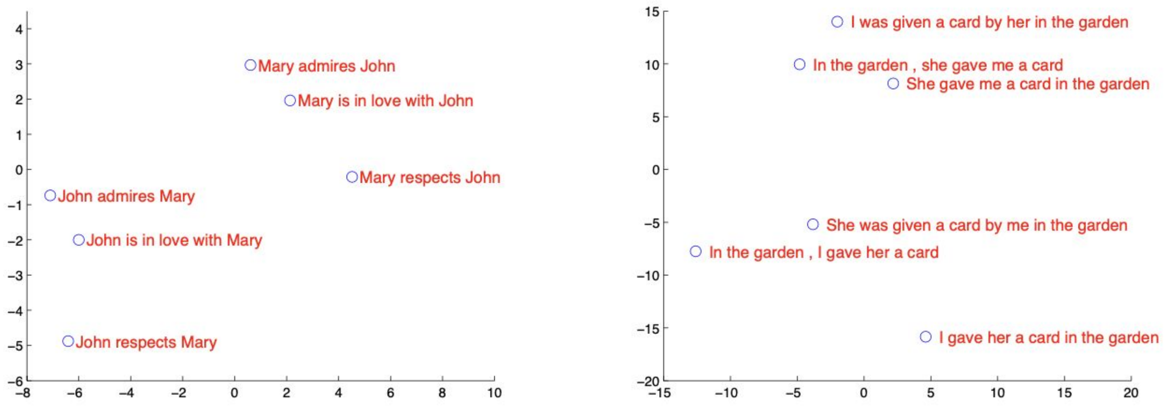
Sequence to Sequence Learning with Neural Network

Abstract

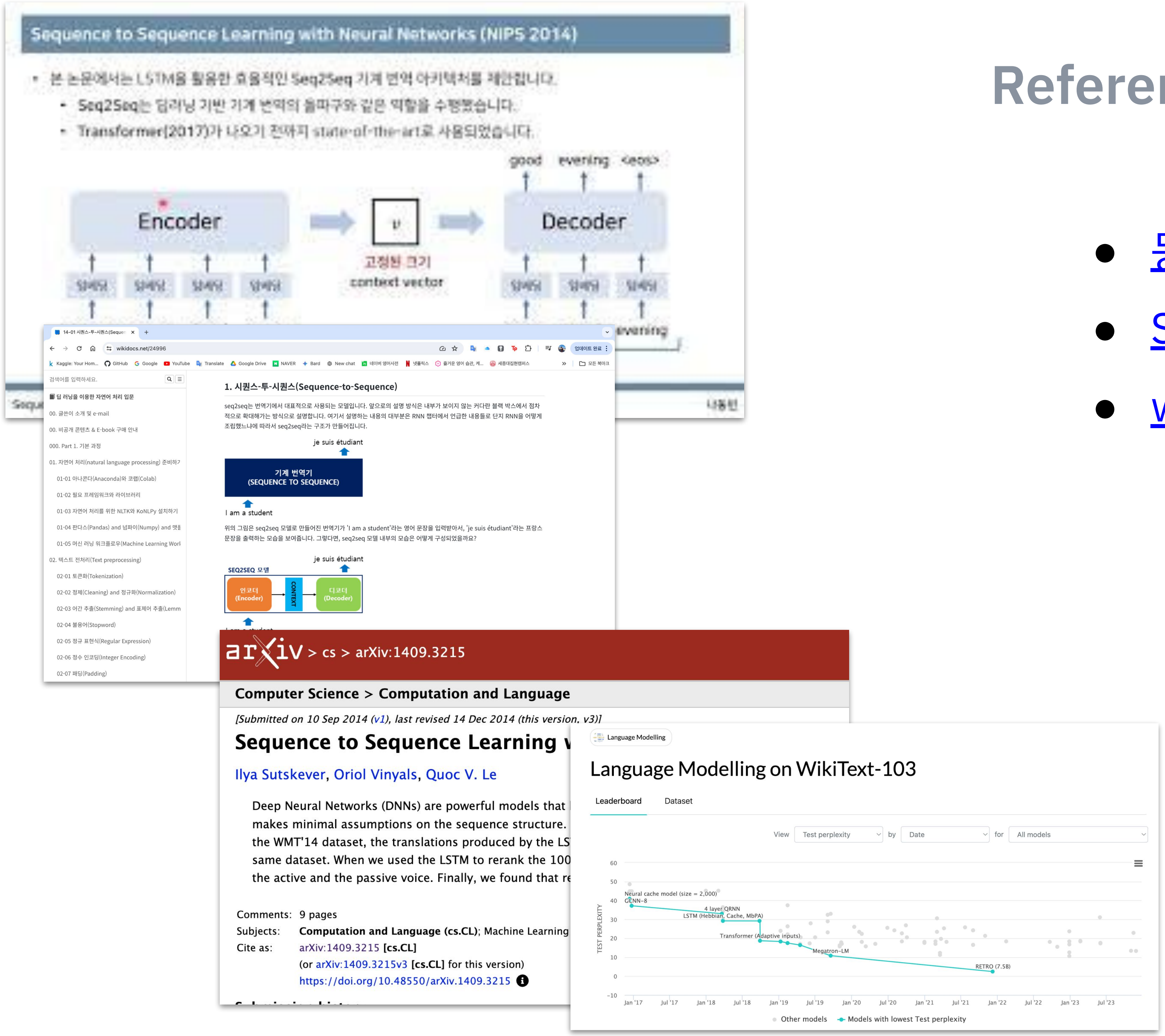
Deep Neural Networks (DNNs) are powerful models that have achieved excellent performance on difficult learning tasks. Although DNNs work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. In this paper, we present a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimensionality, and then another deep LSTM to decode the target sequence from the vector. Our main result is that on an English to French translation task from the WMT'14 dataset, the translations produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where the LSTM's BLEU score was penalized on out-of-vocabulary words. Additionally, the LSTM did not have difficulty on long sentences. For comparison, a phrase-based SMT system achieves a BLEU score of 33.3 on the same dataset. When we used the LSTM to rerank the 1000 hypotheses produced by the aforementioned SMT system, its BLEU score increases to 36.5, which is close to the previous best result on this task. The LSTM also learned sensible phrase and sentence representations that are sensitive to word order and are relatively invariant to the active and the passive voice. Finally, we found that reversing the order of the words in all source sentences (but not target sentences) improved the LSTM's performance markedly, because doing so introduced many short term dependencies between the source and the target sentence which made the optimization problem easier.



$$p(y_1, \ldots, y_{T'} | x_1, \ldots, x_T) = \prod_{t=1}^{T'} p(y_t | v, y_1, \ldots, y_{t-1})$$



출처 : https://arxiv.org/abs/1409.3215



06

Reference

- [동빈나 꼼꼼한 리뷰논문]()
- [Sequence to Sequence Learning with Neural Network]()
- [wikidocs 딥러닝을 이용한 자연어 처리 입문]()

Thank you