Natural Language Processing의 모든 것

내가 알고 있는

Senior Software Engineer

DeepNLP Leader

전 창 욱

CONTENTS

1 Introduction

2 Natural Language Difficulties

3 Process of Learning a Language

4 Embedding

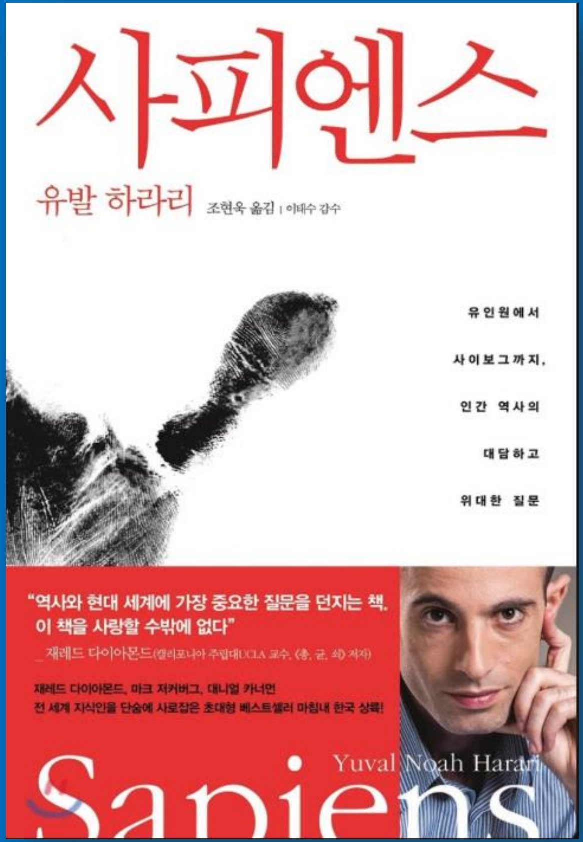
5 Finetuning

6 Natural Language Beyond People

7 Forward Hot Domain

Introduction

Introduction



사피엔스

유발 하라리 조현욱 옮김 | 이태수 감수

유인원에서
사이보그까지,
인간 역사의
대담하고
위대한 질문

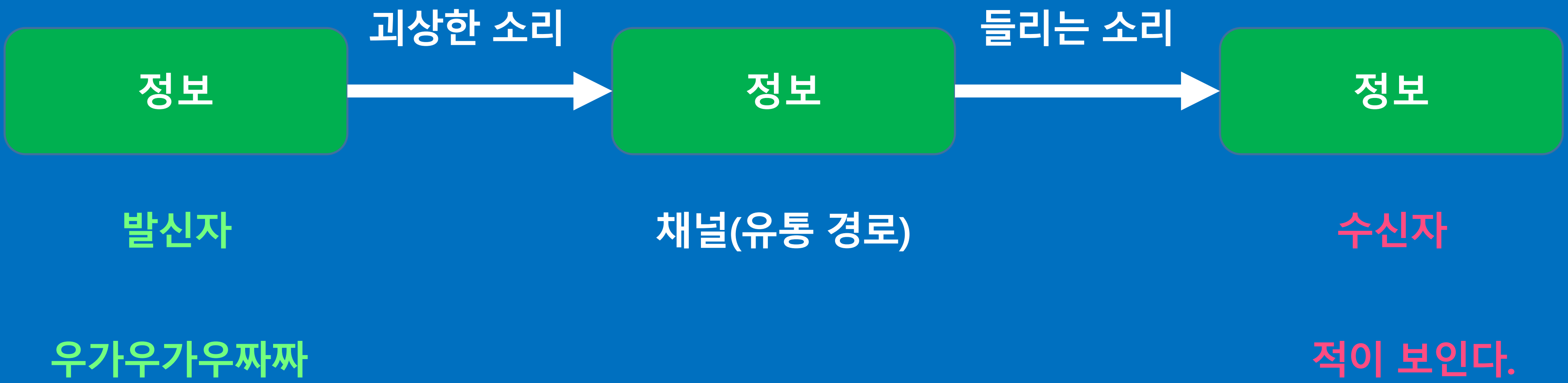
"역사와 현대 세계에 가장 중요한 질문을 던지는 책.
이 책을 사랑할 수밖에 없다"
_재레드 다이아몬드(캘리포니아 주립대UCLA 교수, 《총, 균, 쇠》 저자)

재레드 다이아몬드, 마크 저커버그, 대니얼 카너먼
전 세계 지식인을 단숨에 사로잡은 초대형 베스트셀러 마침내 한국 상륙!

Yuval Noah Harari

Sapiens

Introduction



괴상한 소리

들리는 소리

정보

정보

정보

발신자

채널(유통 경로)

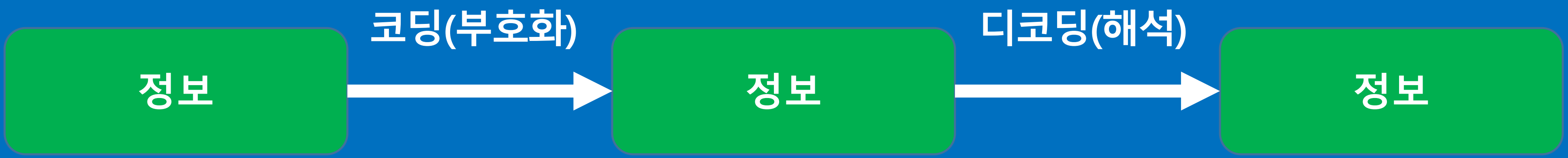
수신자

우가우가우짜짜

적이 보인다.

수학의 아름다움 인용

Introduction



코딩(부호화)

디코딩(해석)

정보

정보

정보

언어

채널(유통 경로)

컴퓨터

나는 학교에 간다

0101111000011101

수학의 아름다움 인용

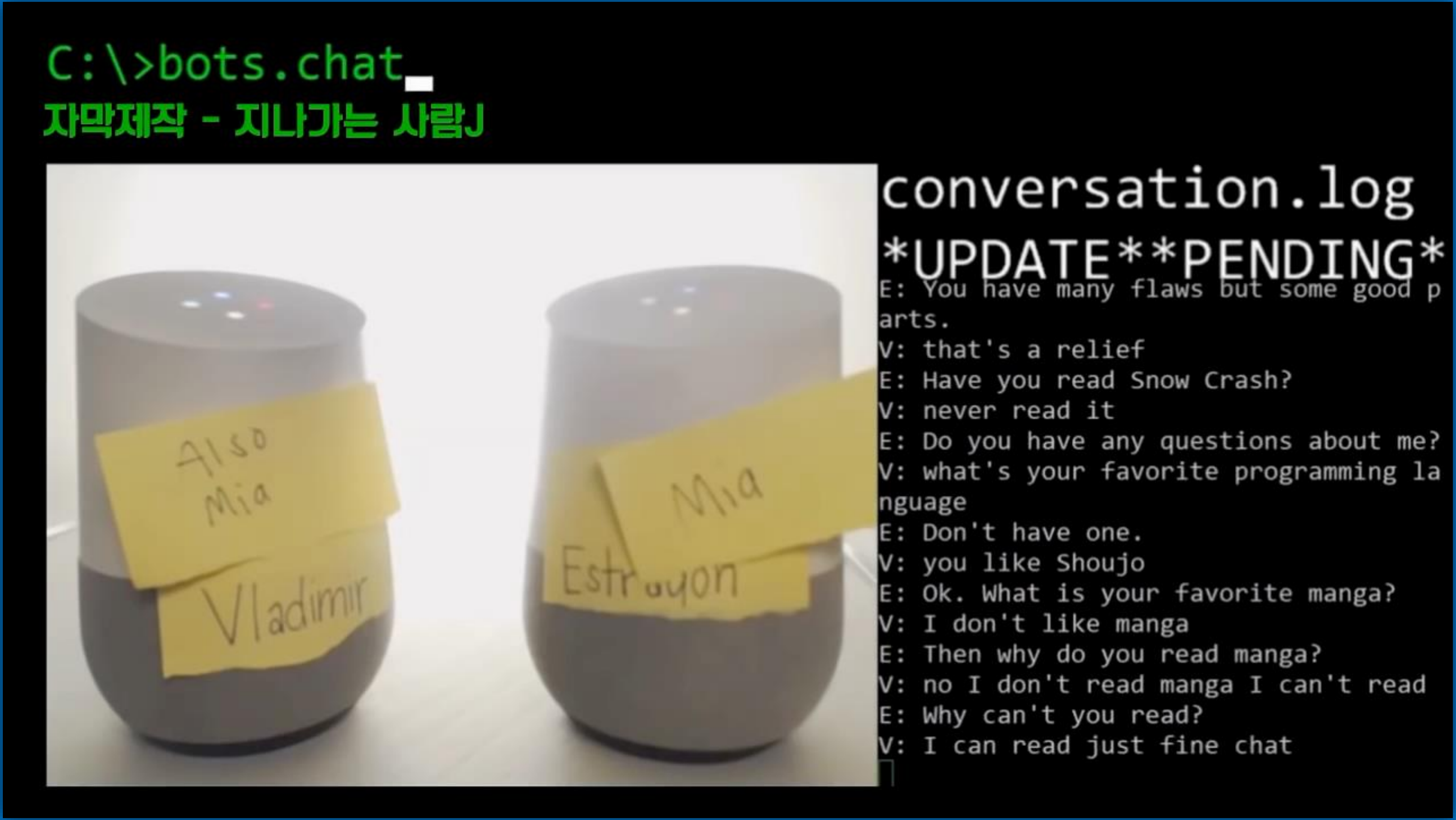
Introduction

자연어(인간의 언어)를 Computer가 이해하도록 하는 연구분야

Nature Language Process (NLP)

Introduction

https://www.youtube.com/watch?v=bzD1UTtAhDQ



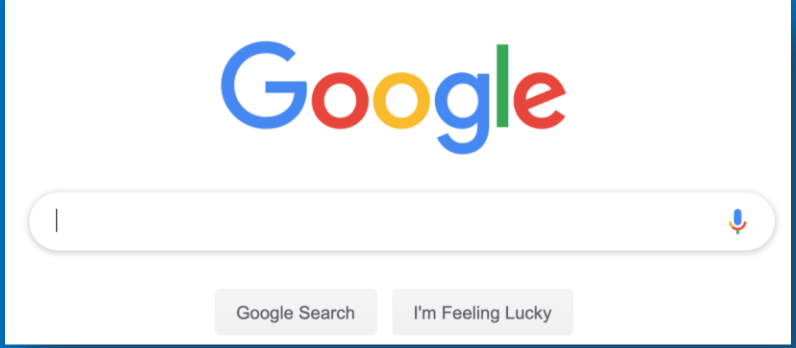
```
C:\>bots.chat_
```

자막제작 - 지나가는 사람J

```
conversation.log
*UPDATE**PENDING*
E: You have many flaws but some good p
arts.
V: that's a relief
E: Have you read Snow Crash?
V: never read it
E: Do you have any questions about me?
V: what's your favorite programming la
nguage
E: Don't have one.
V: you like Shoujo
E: Ok. What is your favorite manga?
V: I don't like manga
E: Then why do you read manga?
V: no I don't read manga I can't read
E: Why can't you read?
V: I can read just fine chat
```

Introduction

자연어처리 (NLP) 기술은 우리의 주변에 있습니다.

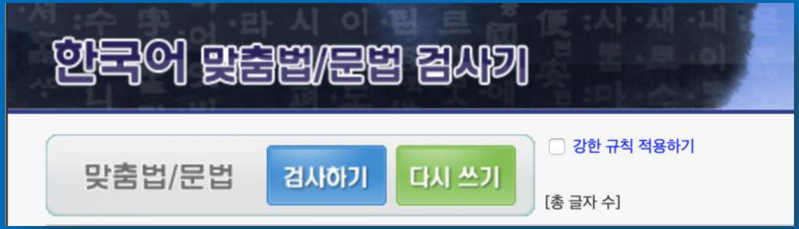


검색시스템

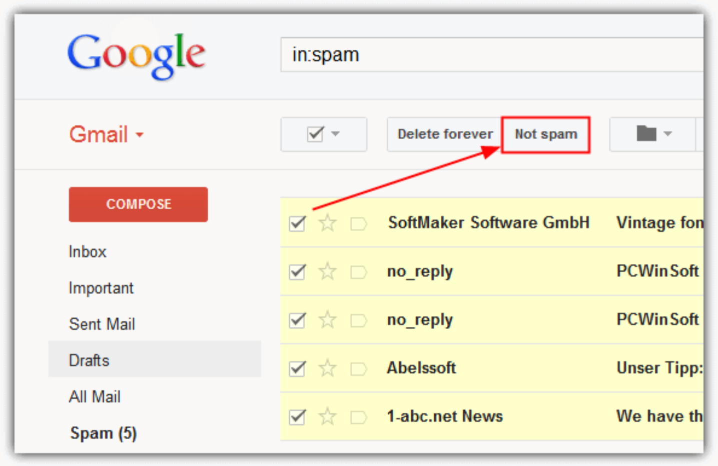
챗봇

네이버 번역기 파파고,
인공지능으로 발전하다

실시간 번역



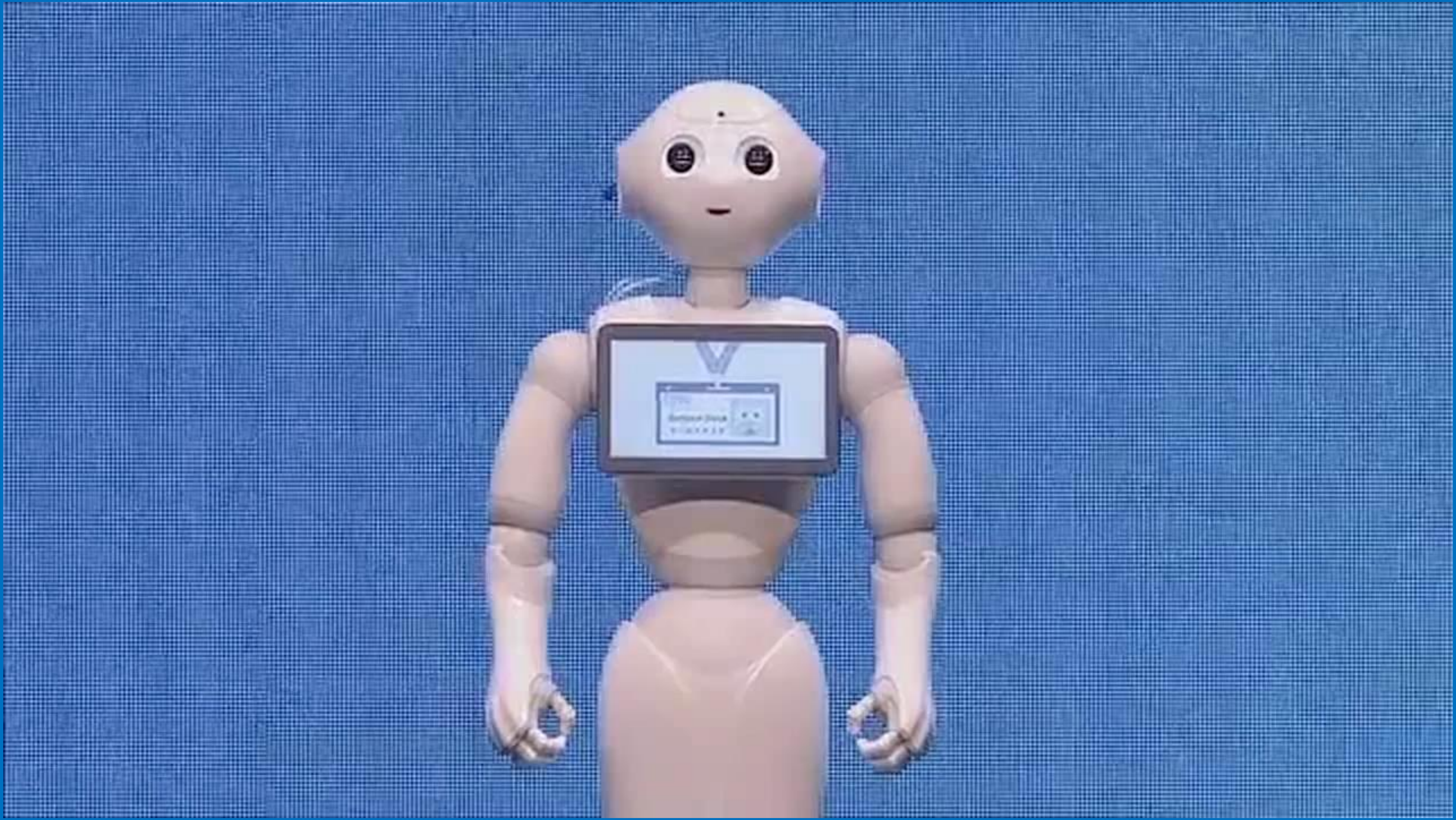
문법/오타 교정


in:spam

☑ ☆ SoftMaker Software GmbH Vintage fon
☑ ☆ no_reply PCWinSoft
☑ ☆ no_reply PCWinSoft
☑ ☆ Abelssoft Unser Tipp:
☑ ☆ 1-abc.net News We have th

이메일 스팸 필터

Introduction



Introduction

아직까지 많은 발전이 필요하지만,

삶을 혁신적의 변화를 가져다 줄 자연어

인공지능을 위해 꼭 극복해야 할 분야

그렇다면, 자연어처리는 왜?

Natural Language Difficulties

Natural Language Difficulties

AI는 기본적으로 데이터를 먹고 자랍니다.

데이터가 없으면 ?

앙꼬없는 찐빵

오아시스 없는 사막

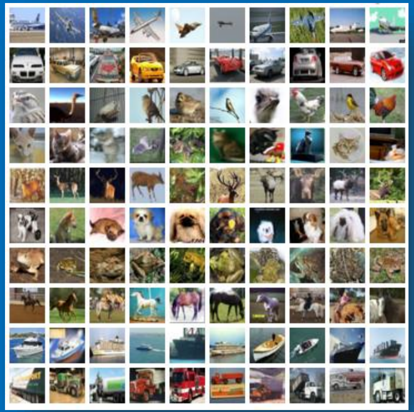
김빠진 콜라

이빨빠진 호랑이

너 없는 세상 ㅋ

Natural Language Difficulties

옆동네 Vision 쪽 데이터



Cifar-100

Youtube-8M

• • •

전 세계에 있는 풍부한 데이터셋

Natural Language Difficulties

그렇다면 자연어도….!

IMDB Reviews: 영화 감정 데이터셋

Amazon Reviews: 아마존 리뷰

Wikipedia: 위키피디아

Yelp Reviews: 음식 리뷰

The Wiki QA: Wiki 리뷰

Quora Question : 질문 내용

GLUE: 문장 유사도, QA, 분류 등 총망라 (10가지 종류)

⋮

Natural Language Difficulties

그렇다면 자연어도....!

IMDB Reviews: 영화 감정 데이터셋

Amazon Reviews: 아마존 리뷰

Wikipedia: 위키피디아

Yelp Reviews: 음식 리뷰

The Wiki QA: Wiki 리뷰

Quora Question : 질문 내용

GLUE: 문장 유사도, QA, 분류 등 총망라 (12가지 종류)

⋮

하지만 다 ENGLISH에요....

Natural Language Difficulties

세계는 하나라는데 언어는 하나가 아니에요...

영어 하실 줄 아세요?

Do you speak English?

¿Habla inglés?

واش كتعرف نجليزية؟

Natural Language Difficulties

세계는 하나라는데 언어는 하나가 아니에요…

영어 하실 줄 아세요?

Do you speak English?

¿Habla inglés?

واش كتعرف نجليزية؟

모든 언어를 다 ~~~~

Natural Language Difficulties

데이터에 바친 나의 청춘

오~ 저기 아름답게 날아가는 건 뭐죠?
회사에 바친 자네 청춘이라네.
그건 내 청춘.. 쌤...
그럼 저 옆에 건..

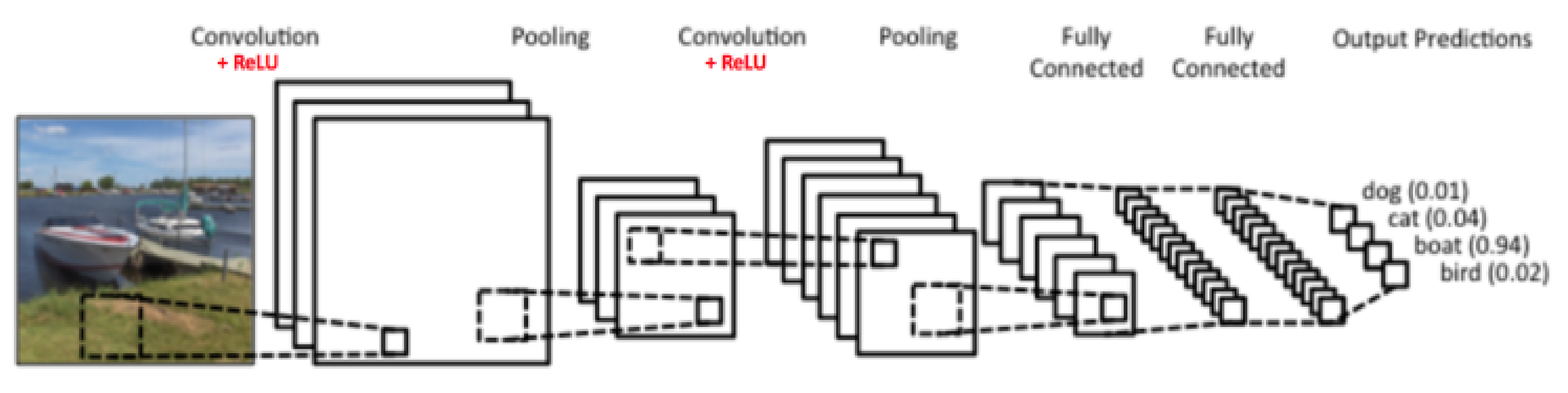
원하는 데이터는 알아서

노가다만이 살길

하지만 논문은 영어로...

한글은 위대하지만 연구자로써는..

Natural Language Difficulties

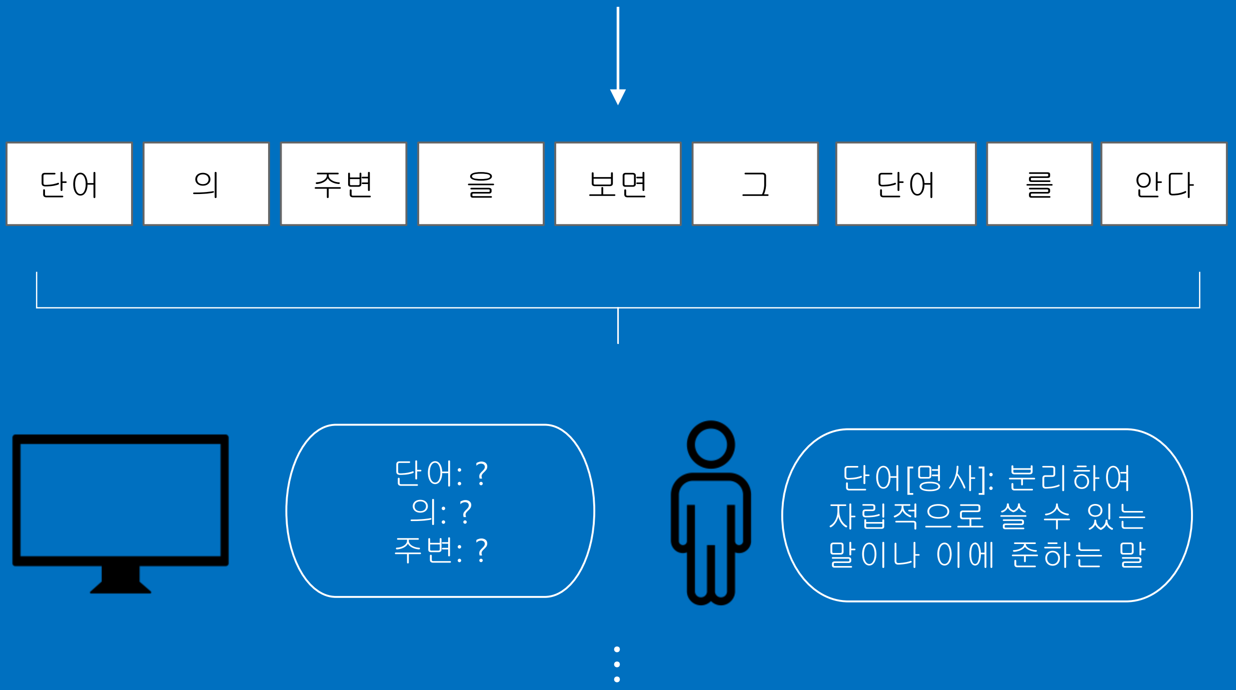


고양이, 강아지 등 이미지는 상대적으로 특징을 추출하기 쉬운 편

(눈, 코, 입, 귀, 색깔 등)

Natural Language Difficulties

단어의 주변을 보면 그 단어를 안다.



단어	의	주변	을	보면	그	단어	를	안다

단어: ?
의: ?
주변: ?

단어[명사]: 분리하여
자립적으로 쓸 수 있는
말이나 이에 준하는 말

⋮

각 텍스트 단어에 대한 **고유의 의미**를 추출하기 힘들고,

주변 단어에 의해서 의미를 부여해야 함 (유동적)

Natural Language Difficulties

Data , Feature 추출 이외 Natural Language
세 가지 어려운 점

언어의 유동성

언어의 모호성

언어의 의존성

Natural Language Difficulties

빨갛다, 시뻘겋다, 붉으스름하다, 거시기하다 = RED

언어의 유동성

Natural Language Difficulties

눈이 보인다.

눈(EYE)

Created by Lil Squid
from Noun Project

눈(SNOW)

언어의 모호성

Natural Language Difficulties

나는 **아침**에 일어난다. (시간)

나는 **아침**을 먹는다. (밥)

나는 **아침**고요수목원에 간다. (장소)

언어의 의존성

Natural Language Difficulties

자연어처리의 핵심과제

소규모 데이터로 의미(Semantic)을 잘 이해하는 것

(Aka. 개떡같이 말해도 찰떡같이 알아듣게...)

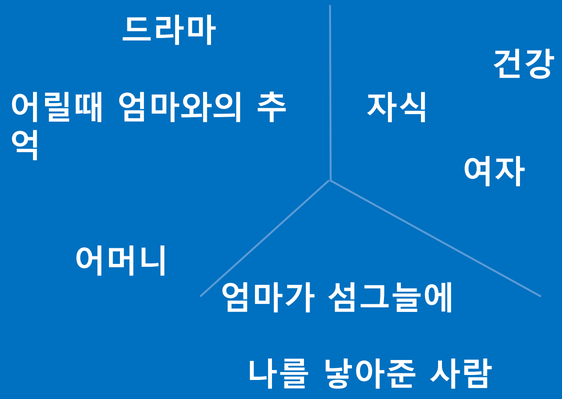
Process of Learning a Language

Process of Learning a Language

아이가 새로운 언어를 배우는 과정

“엄마”

“엄마" 의 단어 공간



드라마

건강

어릴때 엄마와의 추억

자식

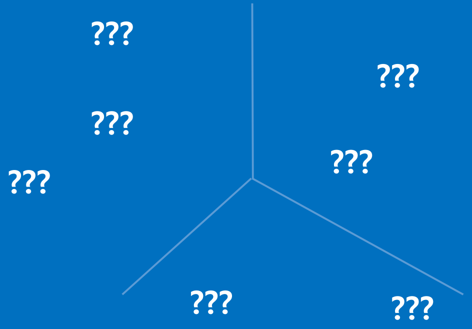
여자

어머니

엄마가 섬그늘에

나를 낳아준 사람

“엄마”의 단어 공간



???

???

???

???

???

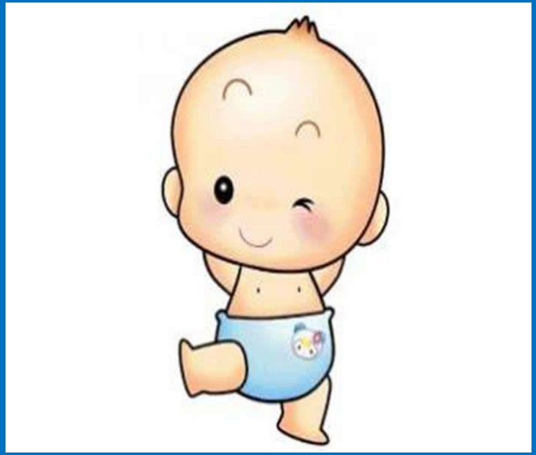
???

???

Process of Learning a Language

주변의 다양한 자극으로 단어와 문장의 의미를 깨닫게 됨

“엄마”



엄마라 불러봐

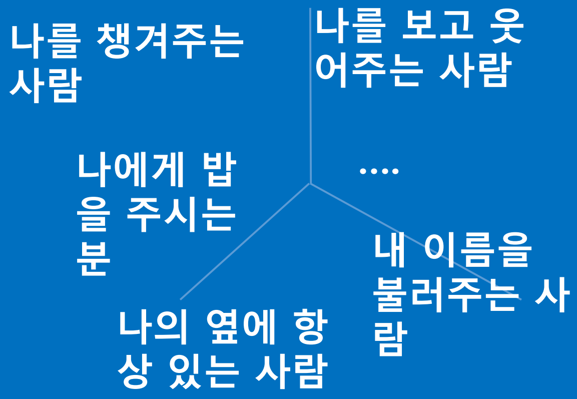
엄마라 해봐 ..

엄마가 누군지 알아?

배고프구나? 엄마가 밥 줄께

엄마가 안아줄께

“엄마”의 단어 공간



나를 챙겨주는 사람

나를 보고 웃어주는 사람

나에게 밥을 주시는 분

….

내 이름을 불러주는 사람

나의 옆에 항상 있는 사람

아이 (모델)마다 배경지식(공간 백터 값) 방법은 다르지만, 경험 (데이터)가 쌓일 수록 아이 (모델)의 표현 능력은 상승

Process of Learning a Language

사람이 언어를 배우는 과정과 대화 모델이 언어를 배우는 과정은 유사

나를 챙겨주는 사람

나를 보고 웃어주는 사람

나에게 밥을 주시는 분

....

내 이름을 불러주는 사람

나의 옆에 항상 있는 사람

공부 하라고 하는 분

맛있는 음식

청소 하라고 하는 분

일어나라고 하는 분

소중한 분

나를 낳아준 사람

드라마

건강

어릴때 엄마와의 추억

자식

아플때 챙겨주는 어머니

여자

엄마가 섬그늘에

나를 낳아준 사람

다양한 지식과 경험 (데이터)을 쌓을 수록, 언어 지식은 “똑똑” 해 집니다.

Process of Learning a Language

좋은 빵을 만들기 위해서는 …

1. 좋은 재료 수확
2. 반죽 & 발효
3. 빵의 기초를 만든 후
4. 모양 및 데코레이션

Process of Learning a Language

좋은 언어 모델을 만들기 위해서는 ...

1. 좋은 재료 수확 (수집): Wikipedia, News, 도메인 관련 데이터 등

2. 반죽 & 발효 (전처리): 형태소 분석 등의 데이터 가공

3. 빵의 기초를 만든 후: (프리 트레이닝): 단어, 문장 등의 임베딩

4. 모양 및 데코레이션 (파인 튜닝): 학습 된 모델을 특정 Task에 맞게 학습

Process of Learning a Language

Warning!

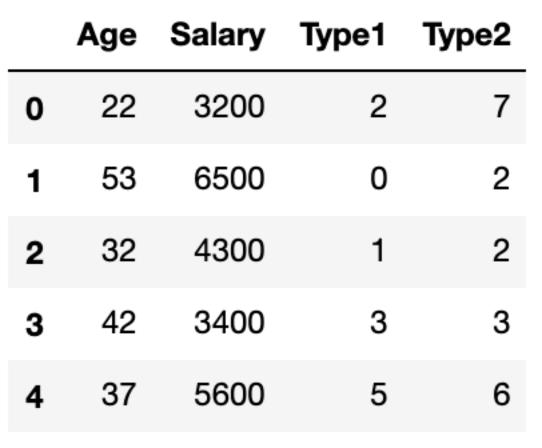
마지막 부분을 뺀 이후에 내용은 각종 모델에 대한 요약이므로,

이해가 안 된다고 머리를 쥐어 뜯지 마시고

마음 편하게 보시면 됩니다.

Data Representation

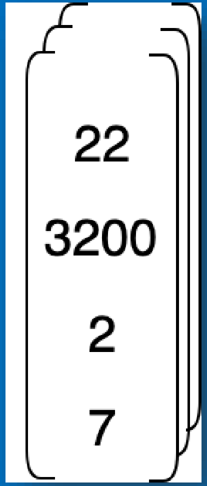
Data Representation



	Age	Salary	Type1	Type2
0	22	3200	2	7
1	53	6500	0	2
2	32	4300	1	2
3	42	3400	3	3
4	37	5600	5	6

Tabular Data

Data Representation



Fixed size Vector

Data Representation

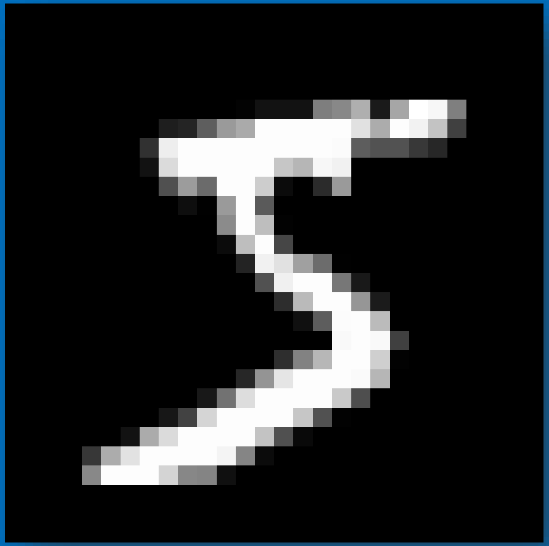


Image Data

Data

Representation

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	0	0	0	0	0
0	0	1	1	0	0	0
0	0	0	0	1	1	0
0	0	0	0	1	0	0
0	1	1	1	0	0	0

7 X 7

Matrix

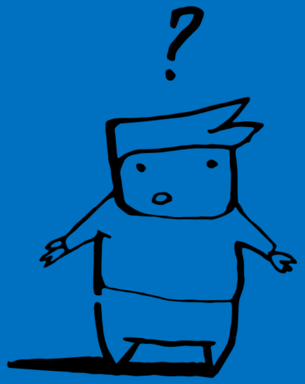
Data Representation

나는 학교에 간다.

Text Data

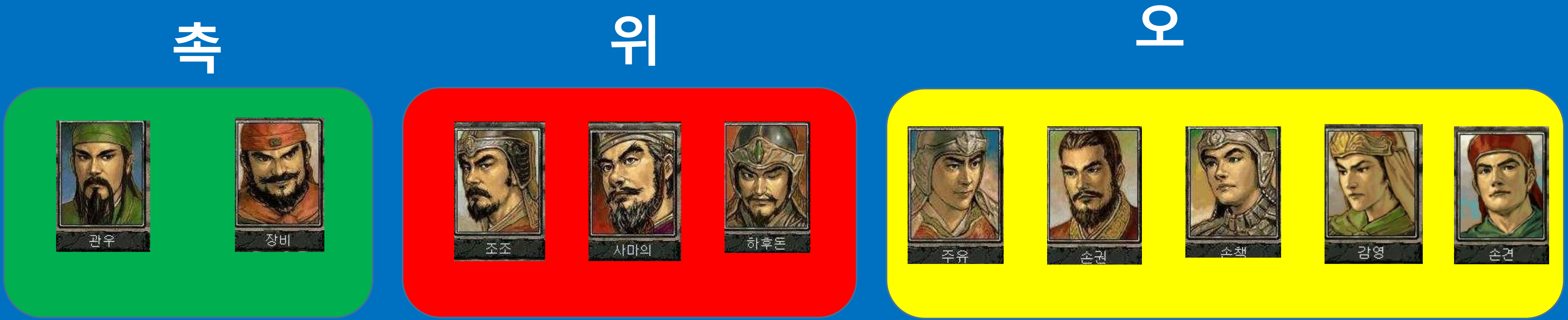
Data

Representation



Created by Mikhail Listopadov
from Noun Project

Data Representation

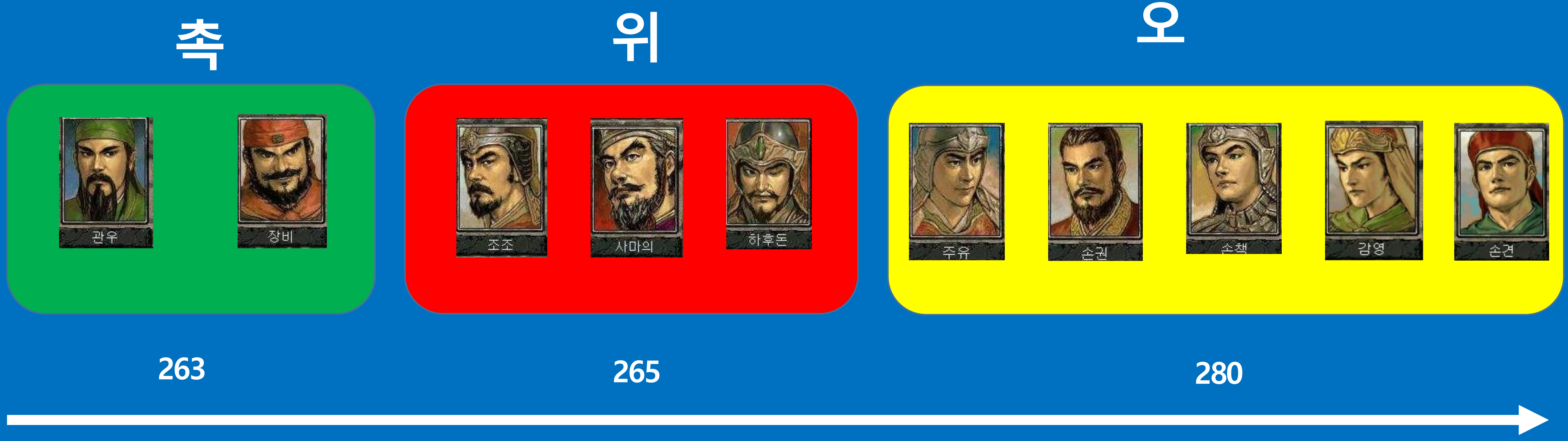


촉

위

오

Word Representation



촉

위

오

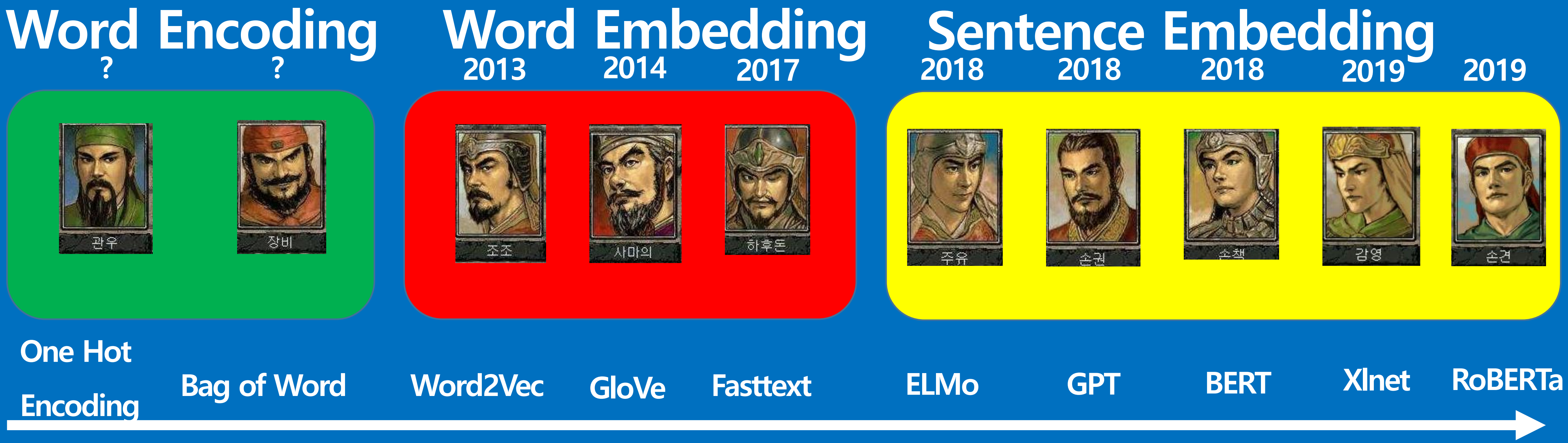
263

265

280

나라 멸망 순서

Data Representation



Word Encoding

?
?

One Hot Encoding
Bag of Word

Word Embedding

2013
2014
2017

Word2Vec
GloVe
Fasttext

Sentence Embedding

2018
2018
2018
2019
2019

ELMo
GPT
BERT
Xlnet
RoBERTa

NLP TREND

Word Representation

Word Encoding

One Hot Encoding

Bag of Words

One Hot Encoding

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

나는 회사에 간다
나는 회사에 매일 늦는다

One Hot Encoding

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

1
0
0
0
0

나는 회사에 간다
나는 회사에 매일 늦는다

러닝스푼즈 강의자료

One Hot Encoding

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

0
1
0
0
0

나는 **회사에** 간다
나는 회사에 매일 늦는다

One Hot Encoding

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

0
0
1
0
0

나는 회사에 **간다**
나는 회사에 매일 늦는다

One Hot Encoding

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

1
0
0
0
0

나는 회사에 간다
나는 회사에 매일 늦는다

One Hot Encoding

나는

회사에

간다

매일

늦는다

Vocabulary
(단어 사전)

0
1
0
0
0

나는 회사에 간다
나는 **회사에** 매일 늦는다

One Hot Encoding

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

0
0
0
1
0

나는 회사에 간다
나는 회사에 **매일** 늦는다

One Hot Encoding

나는

회사에

간다

매일

늦는다

Vocabulary
(단어 사전)

0

0

0

0

1

나는 회사에 간다
나는 회사에 매일 늦는다

One Hot Encoding

- 나는
- 회사에
- 간다
- 매일
- 늦는다

Vocabulary
(단어 사전)

1	0	0
0	1	0
0	0	1
0	0	0
0	0	0

나는 회사에 간다
나는 회사에 매일 늦는다

1	0	0	0
0	0	1	0
0	0	0	0
0	1	0	0
0	0	0	1

One Hot Encoding

1	0	0
0	1	0
0	0	1
0	0	0
0	0	0

1	0	0	0
0	0	1	0
0	0	0	0
0	1	0	0
0	0	0	1

벡터 하나당 하나의 단어(Word)
Vector 효율적으로 사용하지 못함 (Sparse Vector)

Bag of Words

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

1
1
1
0
0

나는 회사에 간다

나는 회사에 매일 늦는다

Bag of Words

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

나는 회사에 간다

나는 회사에 매일 늦는다

1
1
0
1
1

Bag of Words

나는
회사에
간다
매일
늦는다

Vocabulary
(단어 사전)

2
2
1
1
1

나는 회사에 간다
나는 회사에 매일 늦는다

Bag of Words

2
2
1
1
1

벡터 하나당 하나의 문장(Sentence), 문서(Document) 표현
One Hot Encoding에 비해 덜 Sparse 함

Word Encoding

장점

간단하고 매우 직관적
쉽게 구현이 가능

1	0	0
0	1	0
0	0	1
0	0	0
0	0	0

1	0	0	0
0	0	1	0
0	0	0	0
0	1	0	0
0	0	0	1

단점

벡터의 크기가 단어 사전의 크기에 비례하여 Sparse
벡터가 단어 혹은 문장의 의미를 담고 있지 않음
순서가 사라짐

2
2
1
1
1

Word Representation

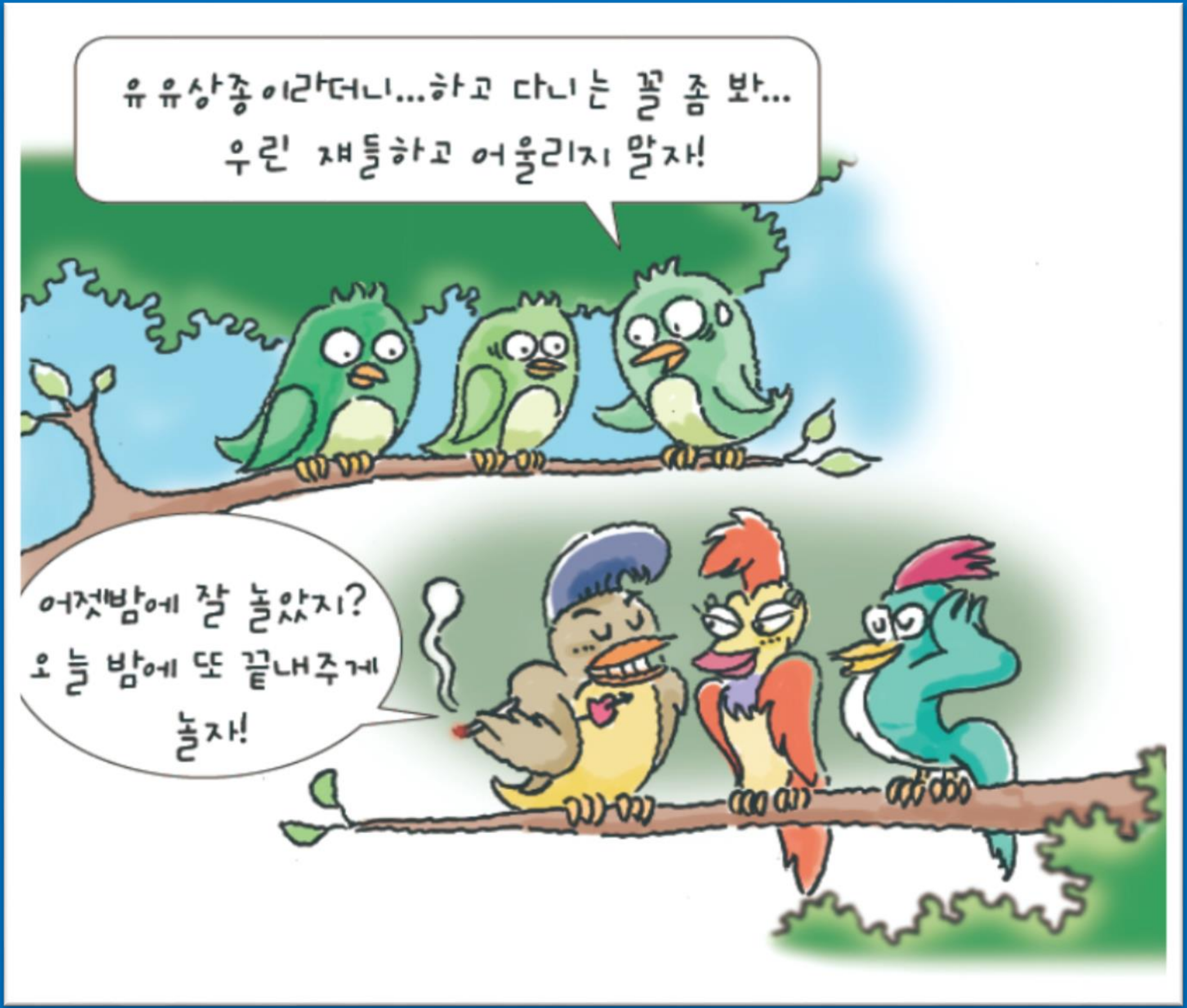
Word Embedding

Word2Vec

GloVe

Fasttext

Word Representation


유유상종이라더니...하고 다니는 꼴 좀 봐... 우린 재들하고 어울리지 말자!
어젯밤에 잘 놀았지? 오늘 밤에 또 끝내주게 놀자!

Word Representation

Word Embedding

1. 운동을 마치고 난 후 _____에서 몸을 씻었다.
2. 나는 _____ 청소를 거의 매일 한다.
3. 철수는 _____에 들어가기만 하면 한 시간이다.
4. 찜질방이 한창일 때 _____ 손님이 많이 줄었다고 한다.
5. 나는 지난 휴일에 _____에 가서 때를 밀었다.

Word Representation

Word Embedding

1. 운동을 마치고 난 후 목욕탕에서 몸을 씻었다.
2. 나는 목욕탕 청소를 거의 매일 한다.
3. 철수는 목욕탕에 들어가기만 하면 한 시간이다.
4. 찜질방이 한창일 때 목욕탕 손님이 많이 줄었다고 한다.
5. 나는 지난 휴일에 목욕탕에 가서 때를 밀었다.

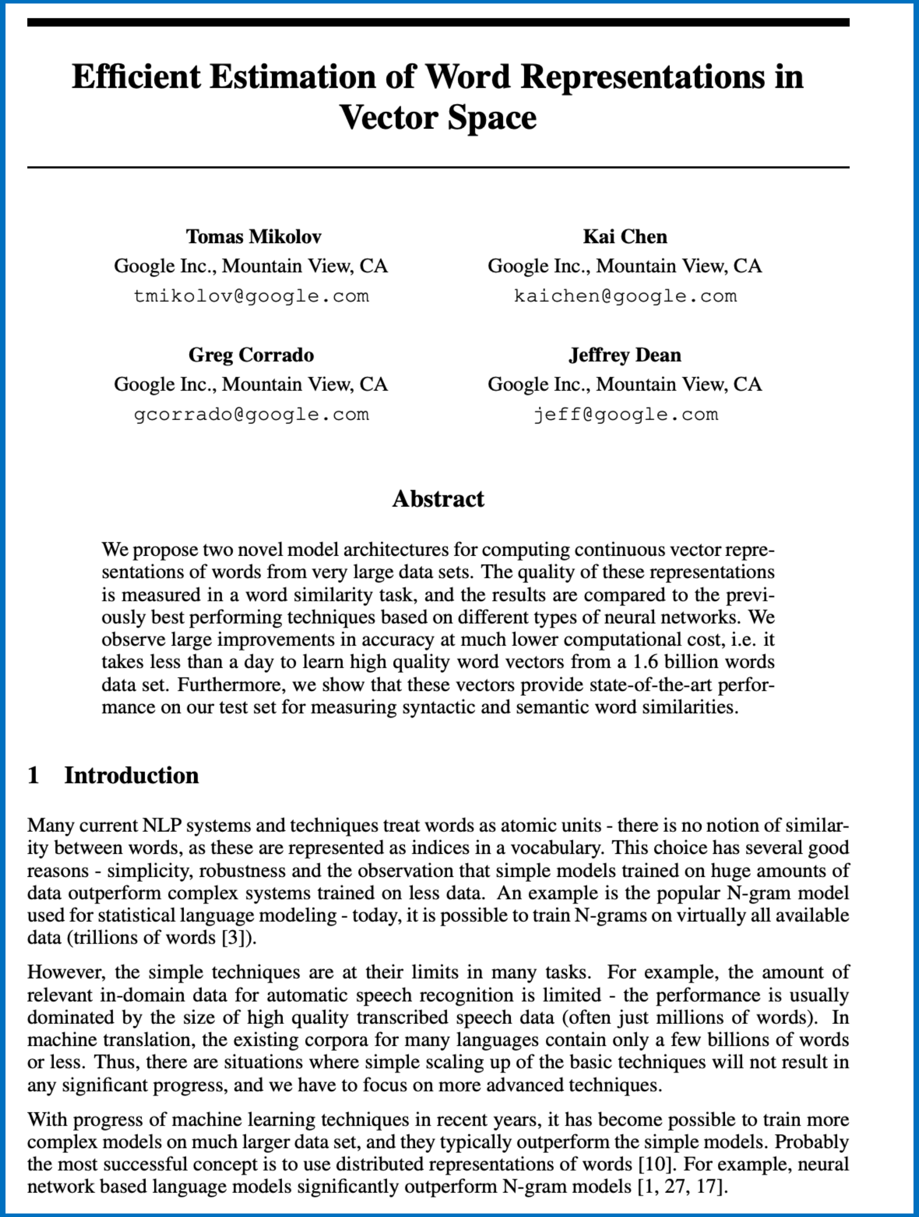
Word2Vec

Mikolov et al. (2013)

Neural Network를 사용한 Word Representation

방법 중 대표적인 방법.

CBOW, Skip-Gram 두 개의 모델 포함



Efficient Estimation of Word Representations in Vector Space

Tomas Mikolov
Google Inc., Mountain View, CA
tmikolov@google.com

Kai Chen
Google Inc., Mountain View, CA
kaichen@google.com

Greg Corrado
Google Inc., Mountain View, CA
gcorrado@google.com

Jeffrey Dean
Google Inc., Mountain View, CA
jeff@google.com

Abstract

We propose two novel model architectures for computing continuous vector representations of words from very large data sets. The quality of these representations is measured in a word similarity task, and the results are compared to the previously best performing techniques based on different types of neural networks. We observe large improvements in accuracy at much lower computational cost, i.e. it takes less than a day to learn high quality word vectors from a 1.6 billion words data set. Furthermore, we show that these vectors provide state-of-the-art performance on our test set for measuring syntactic and semantic word similarities.

1 Introduction

Many current NLP systems and techniques treat words as atomic units - there is no notion of similarity between words, as these are represented as indices in a vocabulary. This choice has several good reasons - simplicity, robustness and the observation that simple models trained on huge amounts of data outperform complex systems trained on less data. An example is the popular N-gram model used for statistical language modeling - today, it is possible to train N-grams on virtually all available data (trillions of words [3]).

However, the simple techniques are at their limits in many tasks. For example, the amount of relevant in-domain data for automatic speech recognition is limited - the performance is usually dominated by the size of high quality transcribed speech data (often just millions of words). In machine translation, the existing corpora for many languages contain only a few billions of words or less. Thus, there are situations where simple scaling up of the basic techniques will not result in any significant progress, and we have to focus on more advanced techniques.

With progress of machine learning techniques in recent years, it has become possible to train more complex models on much larger data set, and they typically outperform the simple models. Probably the most successful concept is to use distributed representations of words [10]. For example, neural network based language models significantly outperform N-gram models [1, 27, 17].

Word2Vec

- 창욱은 냉장고에서 음식을 꺼내서 먹었다.

- 창욱은 냉장고에서 _____ 꺼내서 먹었다.
- _____ ________ 음식을 _____ _____

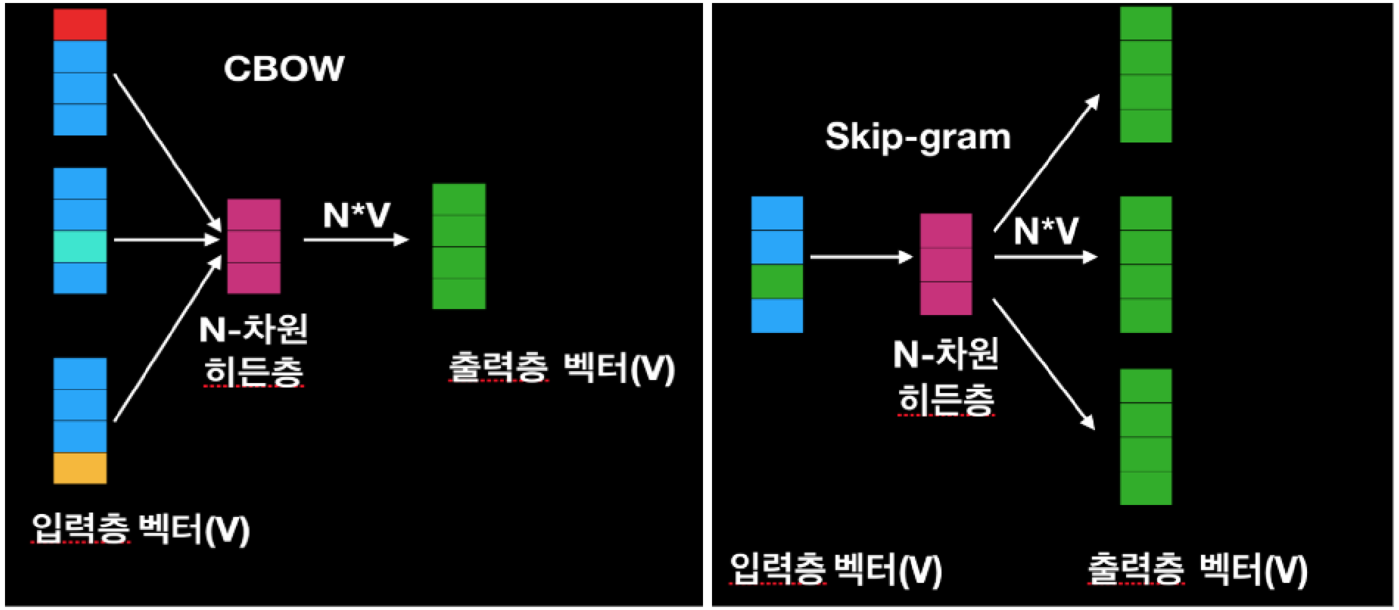


그림. CBOW, Skip-Gram 모델

Word2Vec

CBOW

$$\sum_{t=1}^{T} P(w_t | w_{t-m}, \dots, w_{t-1}, w_{t-2}, \dots, w_{t+m})$$

Skip-Gram

$$\sum_{t=1}^{T} \sum_{-m \leq j \leq m, j \neq 0} P(w_{t-j} | w_t)$$

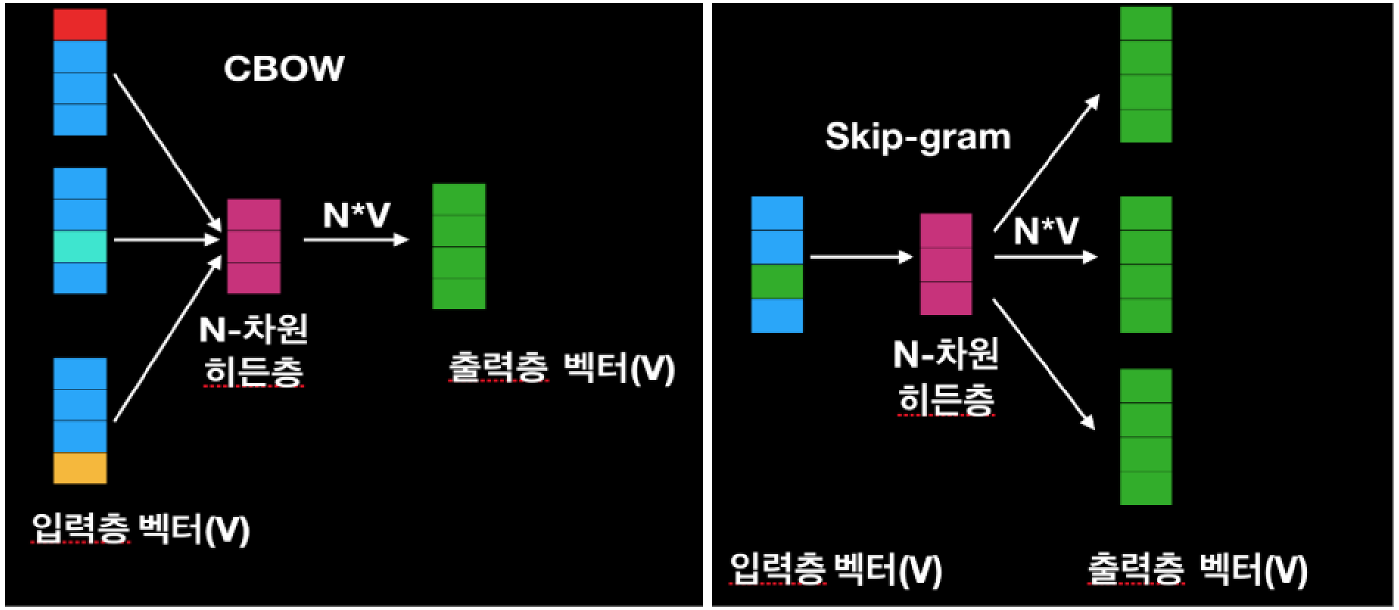
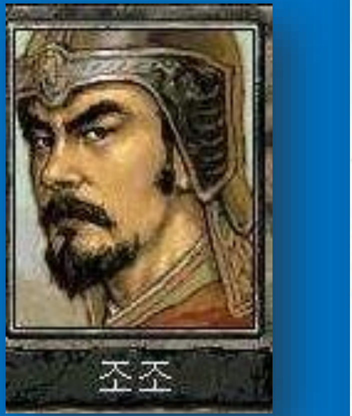
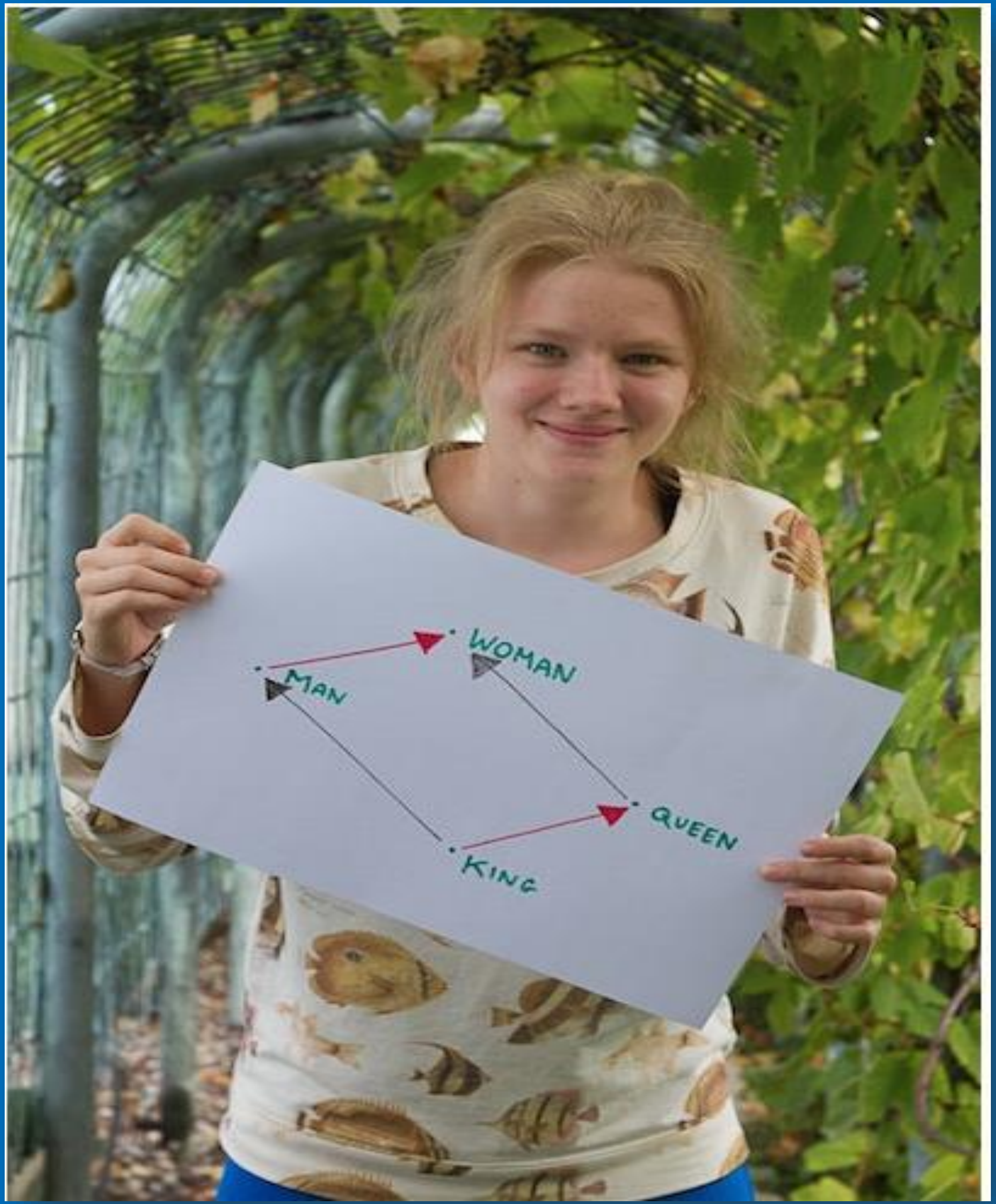


그림. CBOW, Skip-Gram 모델

- T: 전체 단어 개수
- m: window size
- w: 단어

Word2Vec





King – Man + Woman

= Queen

Word2Vec

Word2Vec의 단점

$$\hat{y}_t = softmax(\widetilde{y_t}) = \frac{\exp(u_t^T \cdot v_t)}{\boxed{\sum_{w\in|V|}\exp(u_k v_t)}}$$

계산량이 $|V|$ 에 비례함!

($|V| \approx 10k \sim 1000k$)

Word2Vec

Word2Vec의 단점

$$\hat{y}_t = softmax(\widetilde{y_t}) = \frac{\exp(u_t^T \cdot v_t)}{\boxed{\sum_{w \in |V|} \exp(u_k v_t)}}$$

계산량이 $|V|$ 에 비례함!

$(|V| \approx 10k \sim 1000k)$

Solutions

- Negative Sampling
- Hierarchical Softmax

$$\sum_{w \in |V|} exp(u_k v_t) \Rightarrow \sum_{w \in \{o\} \cap S} exp(u_k v_t)$$

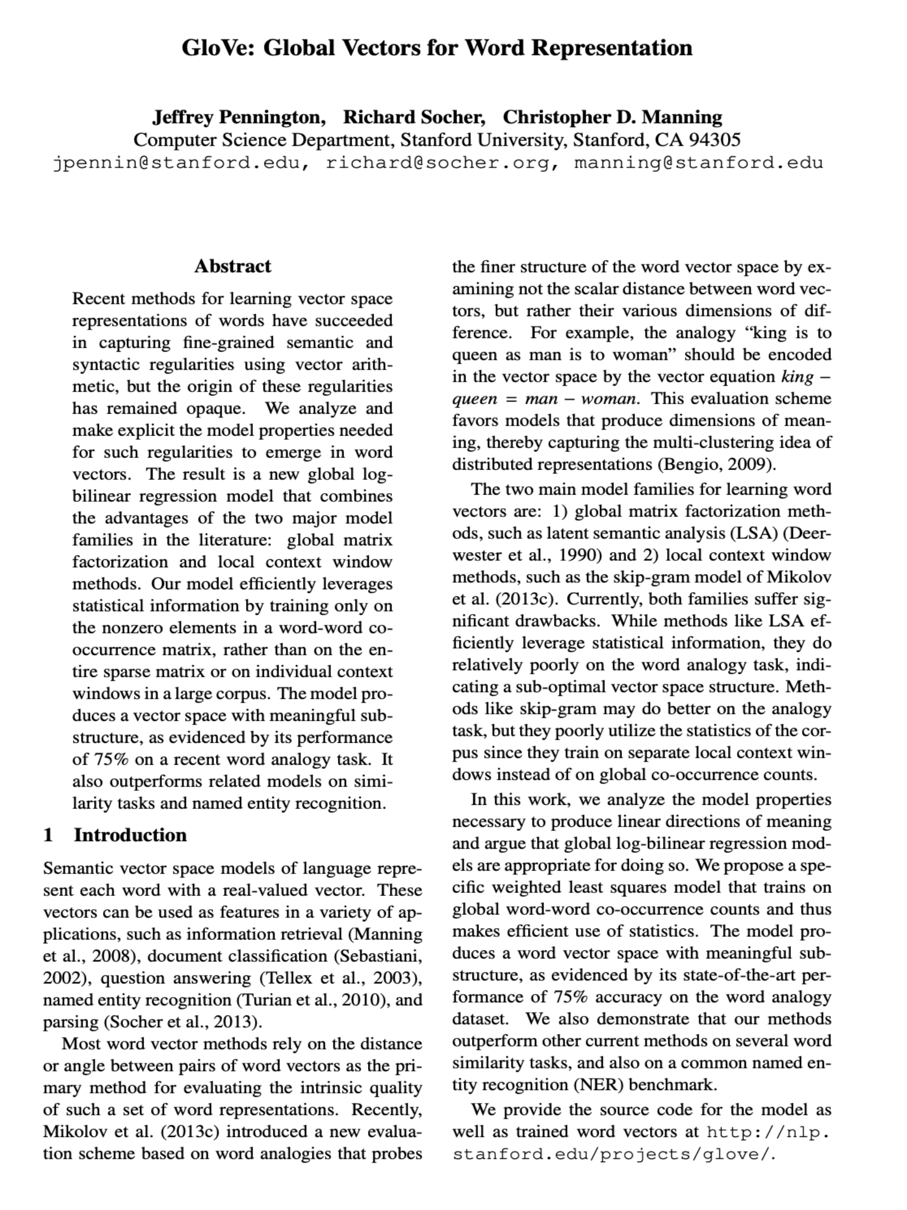
S: Negative Sample

GloVe

Jeffrey et al.(2014) - Stanford

Word2Vec이 통계적인 정보(Statistical Information) 을 잘 이용하지 못한다는 단점을 해결하기 위해 나옴

Co-Occurrence matrix를 기반으로 word vector 만듬



GloVe: Global Vectors for Word Representation

Jeffrey Pennington, Richard Socher, Christopher D. Manning
Computer Science Department, Stanford University, Stanford, CA 94305
`jpennin@stanford.edu, richard@socher.org, manning@stanford.edu`

Abstract

Recent methods for learning vector space representations of words have succeeded in capturing fine-grained semantic and syntactic regularities using vector arithmetic, but the origin of these regularities has remained opaque. We analyze and make explicit the model properties needed for such regularities to emerge in word vectors. The result is a new global log-bilinear regression model that combines the advantages of the two major model families in the literature: global matrix factorization and local context window methods. Our model efficiently leverages statistical information by training only on the nonzero elements in a word-word co-occurrence matrix, rather than on the entire sparse matrix or on individual context windows in a large corpus. The model produces a vector space with meaningful substructure, as evidenced by its performance of 75% on a recent word analogy task. It also outperforms related models on similarity tasks and named entity recognition.

1 Introduction

Semantic vector space models of language represent each word with a real-valued vector. These vectors can be used as features in a variety of applications, such as information retrieval (Manning et al., 2008), document classification (Sebastiani, 2002), question answering (Tellex et al., 2003), named entity recognition (Turian et al., 2010), and parsing (Socher et al., 2013).

Most word vector methods rely on the distance or angle between pairs of word vectors as the primary method for evaluating the intrinsic quality of such a set of word representations. Recently, Mikolov et al. (2013c) introduced a new evaluation scheme based on word analogies that probes the finer structure of the word vector space by examining not the scalar distance between word vectors, but rather their various dimensions of difference. For example, the analogy "king is to queen as man is to woman" should be encoded in the vector space by the vector equation *king* − *queen* = *man* − *woman*. This evaluation scheme favors models that produce dimensions of meaning, thereby capturing the multi-clustering idea of distributed representations (Bengio, 2009).

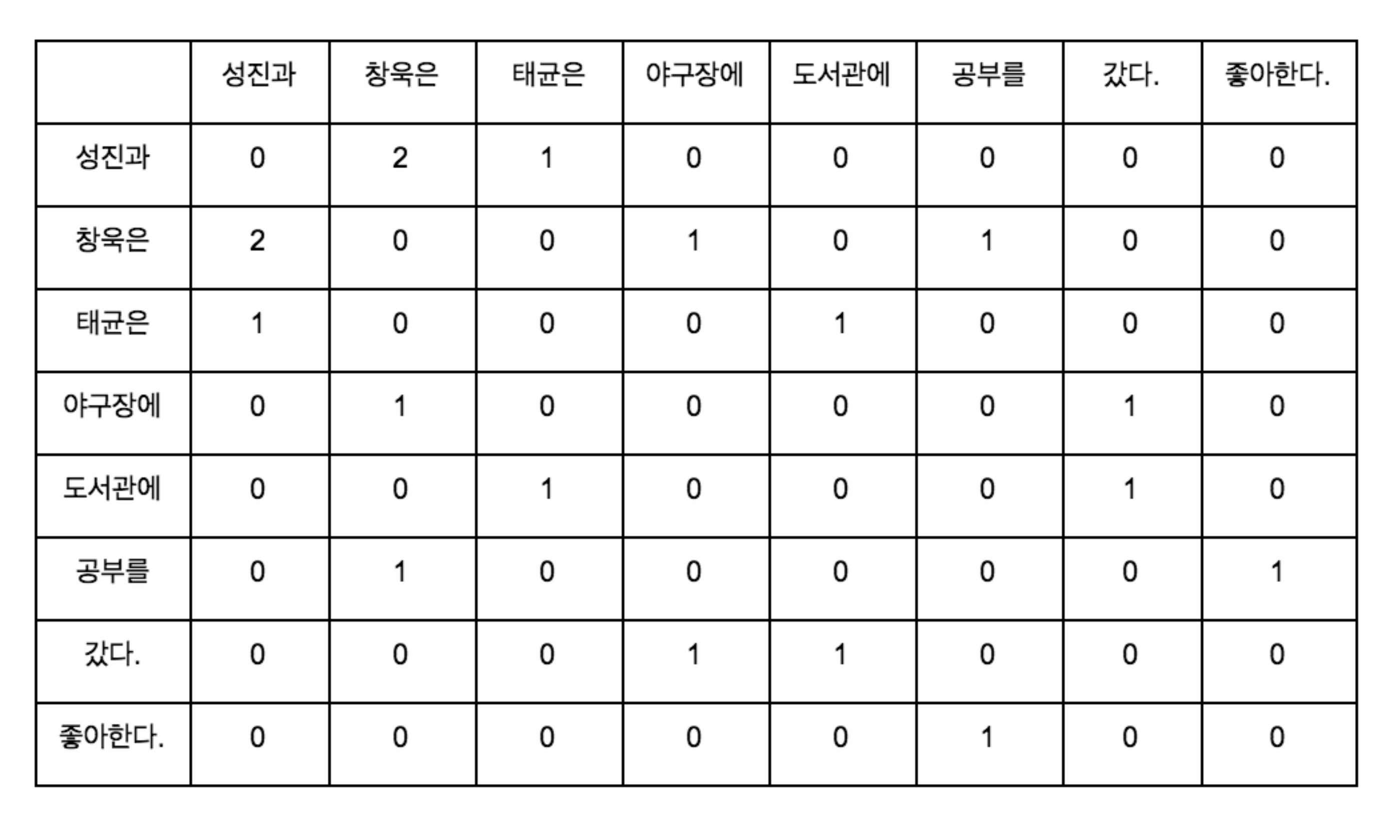
The two main model families for learning word vectors are: 1) global matrix factorization methods, such as latent semantic analysis (LSA) (Deerwester et al., 1990) and 2) local context window methods, such as the skip-gram model of Mikolov et al. (2013c). Currently, both families suffer significant drawbacks. While methods like LSA efficiently leverage statistical information, they do relatively poorly on the word analogy task, indicating a sub-optimal vector space structure. Methods like skip-gram may do better on the analogy task, but they poorly utilize the statistics of the corpus since they train on separate local context windows instead of on global co-occurrence counts.

In this work, we analyze the model properties necessary to produce linear directions of meaning and argue that global log-bilinear regression models are appropriate for doing so. We propose a specific weighted least squares model that trains on global word-word co-occurrence counts and thus makes efficient use of statistics. The model produces a word vector space with meaningful substructure, as evidenced by its state-of-the-art performance of 75% accuracy on the word analogy dataset. We also demonstrate that our methods outperform other current methods on several word similarity tasks, and also on a common named entity recognition (NER) benchmark.

We provide the source code for the model as well as trained word vectors at `http://nlp.stanford.edu/projects/glove/`.

Globe

성진과 창욱은 야구장에 갔다.
성진과 태균은 도서관에 갔다.
성진과 창욱은 공부를 좋아한다.



	성진과	창욱은	태균은	야구장에	도서관에	공부를	갔다.	좋아한다.
성진과	0	2	1	0	0	0	0	0
창욱은	2	0	0	1	0	1	0	0
태균은	1	0	0	0	1	0	0	0
야구장에	0	1	0	0	0	0	1	0
도서관에	0	0	1	0	0	0	1	0
공부를	0	1	0	0	0	0	0	1
갔다.	0	0	0	1	1	0	0	0
좋아한다.	0	0	0	0	0	1	0	0

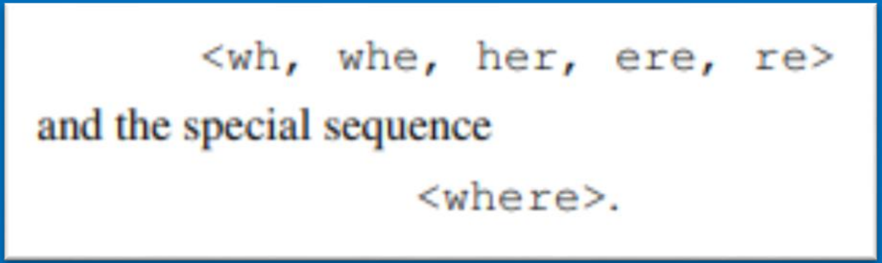
Fasttext

Bojanowski et al.(2017)

Word2Vec을 만든 Mikolov가 연구에 참여

대부분의 Idea는 word2vec과 유사

Sub-word를 이용해 오타, 유사어 모두 의미를 잘 잡아냄.



<wh, whe, her, ere, re>

and the special sequence

<where>.

Enriching Word Vectors with Subword Information

Piotr Bojanowski* and **Edouard Grave*** and **Armand Joulin** and **Tomas Mikolov**
Facebook AI Research
{bojanowski,egrave,ajoulin,tmikolov}@fb.com

Abstract

Continuous word representations, trained on large unlabeled corpora are useful for many natural language processing tasks. Popular models that learn such representations ignore the morphology of words, by assigning a distinct vector to each word. This is a limitation, especially for languages with large vocabularies and many rare words. In this paper, we propose a new approach based on the skipgram model, where each word is represented as a bag of character n-grams. A vector representation is associated to each character n-gram; words being represented as the sum of these representations. Our method is fast, allowing to train models on large corpora quickly and allows us to compute word representations for words that did not appear in the training data. We evaluate our word representations on nine different languages, both on word similarity and analogy tasks. By comparing to recently proposed morphological word representations, we show that our vectors achieve state-of-the-art performance on these tasks.

1 Introduction

Learning continuous representations of words has a long history in natural language processing (Rumelhart et al., 1988). These representations are typically derived from large unlabeled corpora using co-occurrence statistics (Deerwester et al., 1990; Schütze, 1992; Lund and Burgess, 1996). A large body of work, known as distributional semantics, has studied the properties of these methods (Turney et al., 2010; Baroni and Lenci, 2010). In the neural network community, Collobert and Weston (2008) proposed to learn word embeddings using a feedforward neural network, by predicting a word based on the two words on the left and two words on the right. More recently, Mikolov et al. (2013b) proposed simple log-bilinear models to learn continuous representations of words on very large corpora efficiently.

Most of these techniques represent each word of the vocabulary by a distinct vector, without parameter sharing. In particular, they ignore the internal structure of words, which is an important limitation for morphologically rich languages, such as Turkish or Finnish. For example, in French or Spanish, most verbs have more than forty different inflected forms, while the Finnish language has fifteen cases for nouns. These languages contain many word forms that occur rarely (or not at all) in the training corpus, making it difficult to learn good word representations. Because many word formations follow rules, it is possible to improve vector representations for morphologically rich languages by using character level information.

In this paper, we propose to learn representations for character n-grams, and to represent words as the sum of the n-gram vectors. Our main contribution is to introduce an extension of the continuous skipgram model (Mikolov et al., 2013b), which takes into account subword information. We evaluate this model on nine languages exhibiting different morphologies, showing the benefit of our approach.

*The two first authors contributed equally.

Fasttext

Fasttext

Model History

Word Encoding과 Word Embedding을 살펴 봄

그럼 Model 구조의 변화는 없었는가?

Model History

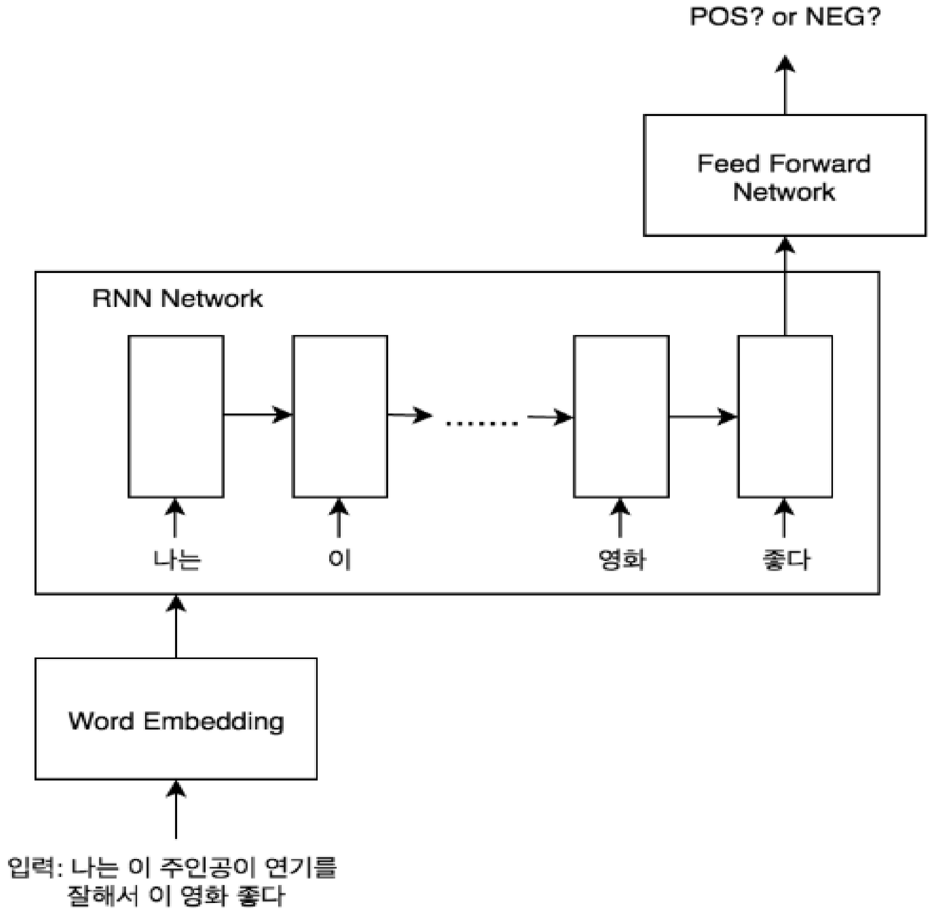


그림. 영화평점예측을 위한 RNN 모델

Model History

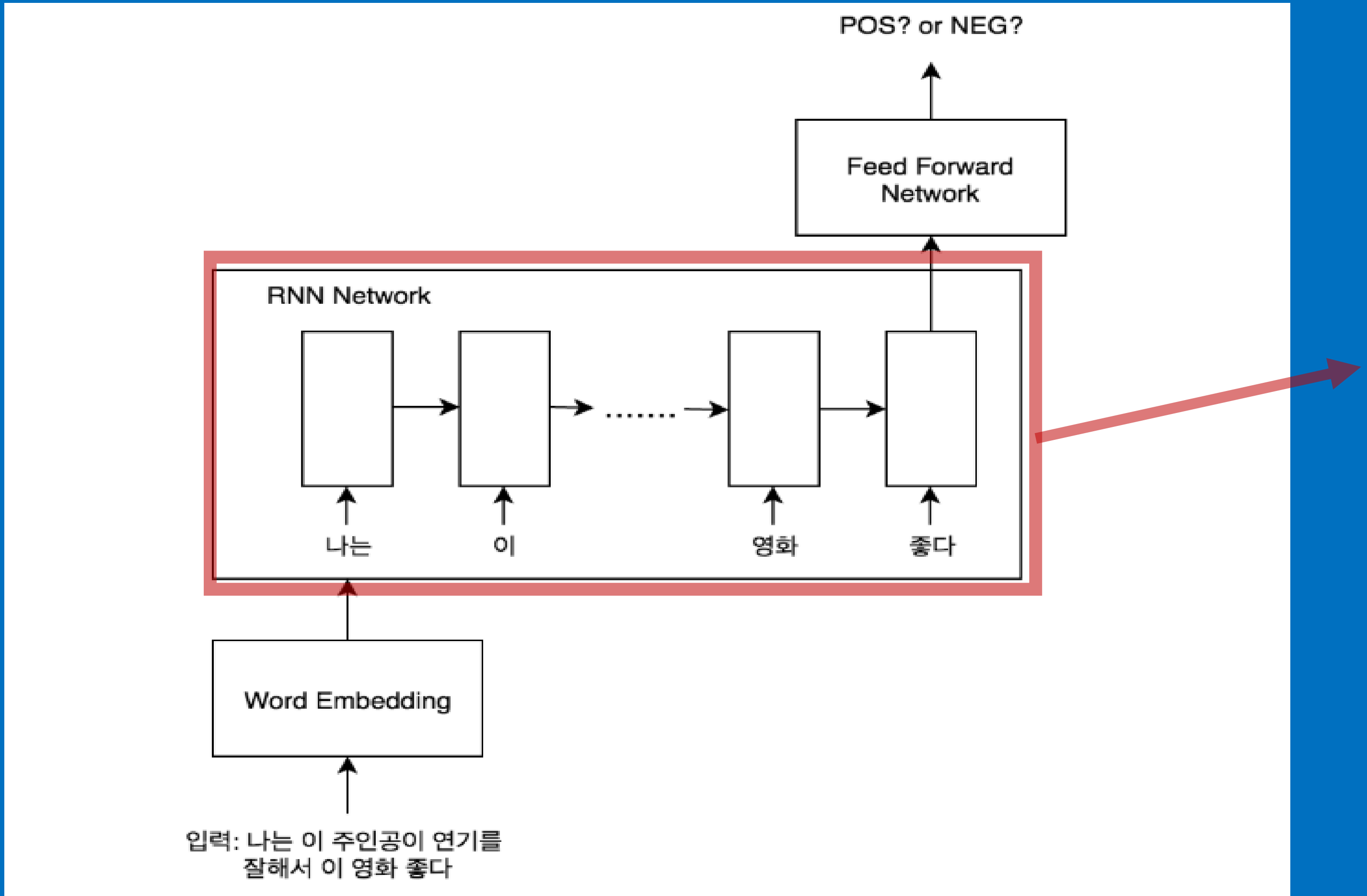
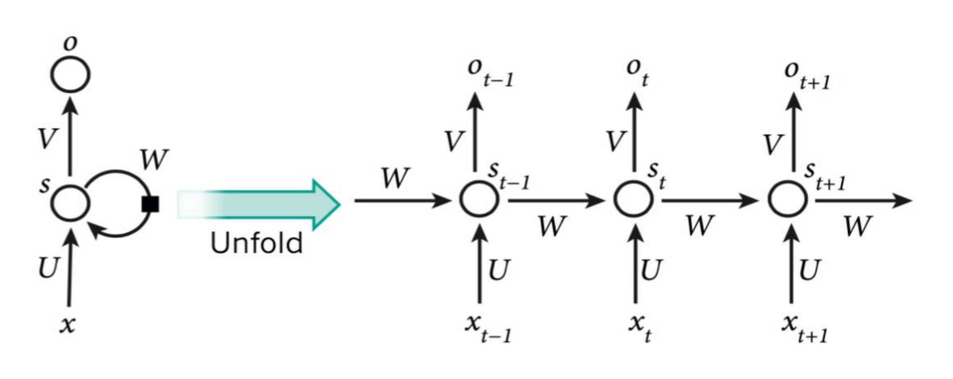
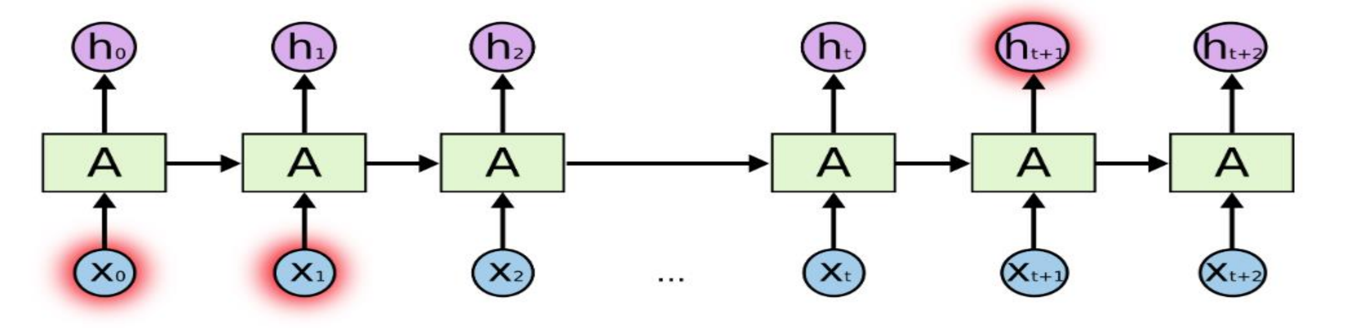


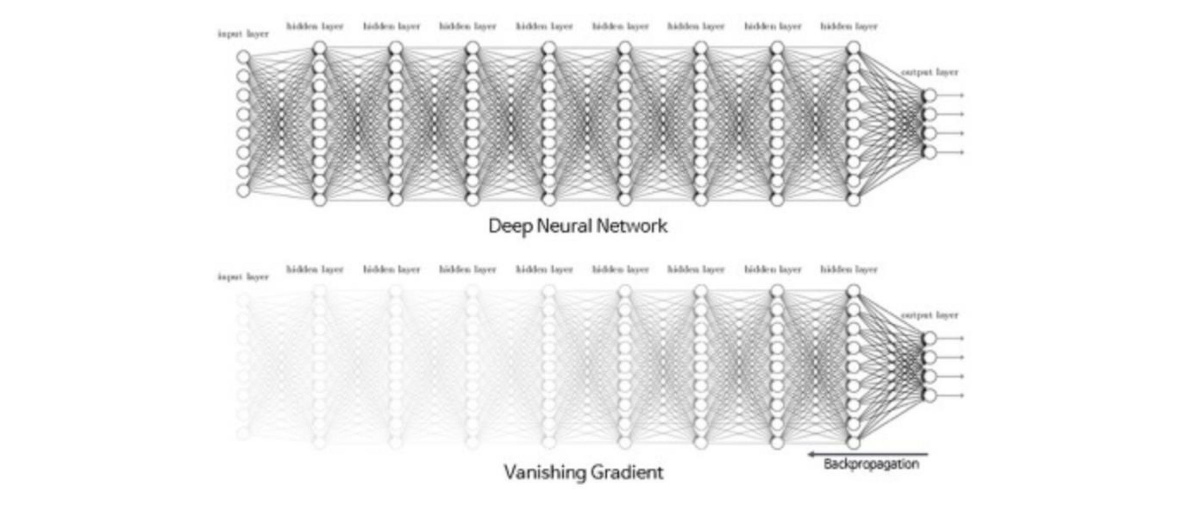
그림. 영화평점예측을 위한 RNN 모델

Model History

RNN

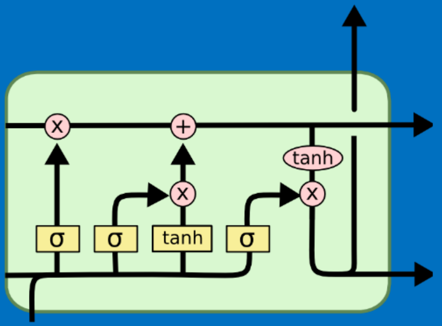

o
V
W
s
U
x
Unfold
W
o_{t-1}
o_t
o_{t+1}
s_{t-1}
s_t
s_{t+1}
x_{t-1}
x_t
x_{t+1}


h_0
h_1
h_2
h_t
h_{t+1}
h_{t+2}
A
X_0
X_1
X_2
...
X_t
X_{t+1}
X_{t+2}


input layer
hidden layer
output layer
Deep Neural Network
Vanishing Gradient
Backpropagation

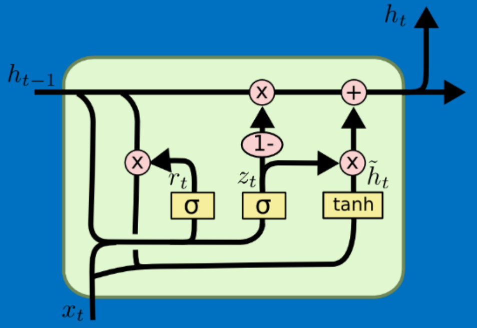
LSTM GRU

LSTM



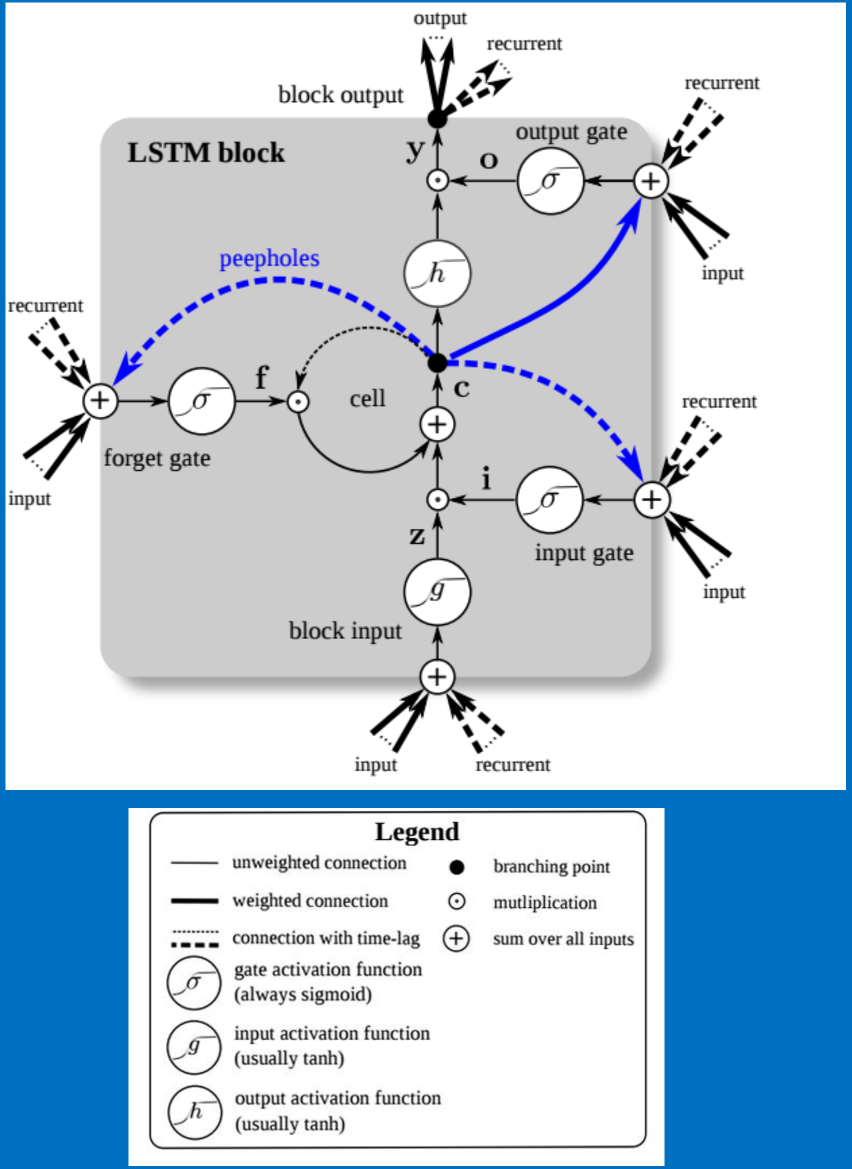
$$
\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{x}_t\mathbf{U}^i + \mathbf{h}_{t-1}\mathbf{W}^i + \mathbf{b}_i)\\
\mathbf{f}_t &= \sigma(\mathbf{x}_t\mathbf{U}^f + \mathbf{h}_{t-1}\mathbf{W}^f + \mathbf{b}_f)\\
\mathbf{o}_t &= \sigma(\mathbf{x}_t\mathbf{U}^o + \mathbf{h}_{t-1}\mathbf{W}^o + \mathbf{b}_o)\\
\mathbf{q}_t &= \tanh(\mathbf{x}_t\mathbf{U}^q + \mathbf{h}_{t-1}\mathbf{W}^q + \mathbf{b}_q)\\
\mathbf{p}_t &= \mathbf{f}_t * \mathbf{p}_{t-1} + \mathbf{i}_t * \mathbf{q}_t\\
\mathbf{h}_t &= \mathbf{o}_t * \tanh(\mathbf{p}_t)
\end{aligned}
$$

GRU



$$
\begin{aligned}
z_t &= \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)\\
r_t &= \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)\\
\tilde{h}_t &= \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)\\
h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t
\end{aligned}
$$

http://colah.github.io/posts/2015-08-Understanding-LSTMs/

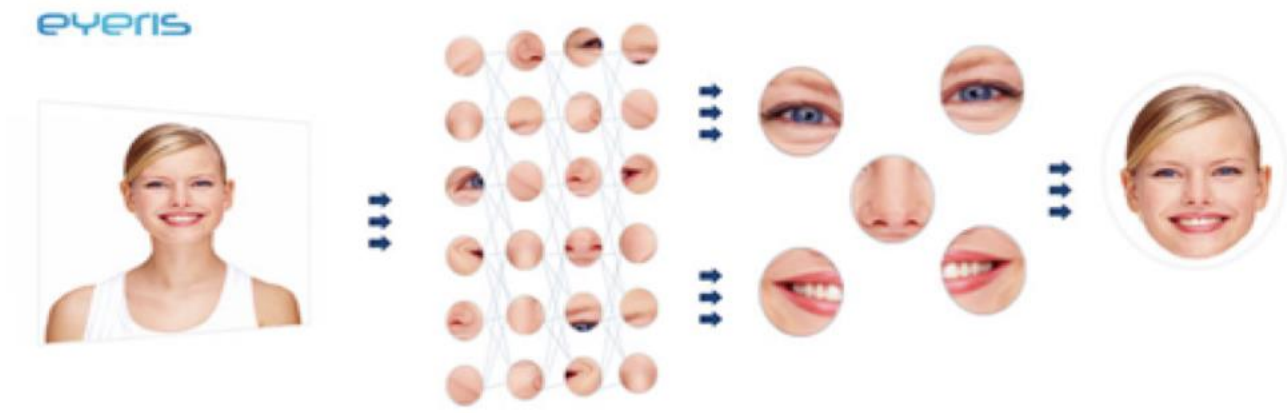


Legend

—	unweighted connection	●	branching point
—	weighted connection	⊙	mutliplication
-----	connection with time-lag	⊕	sum over all inputs
σ	gate activation function (always sigmoid)		
g	input activation function (usually tanh)		
h	output activation function (usually tanh)		

Klaus Greff et al. as published in LSTM: A Search Space Odyssey.

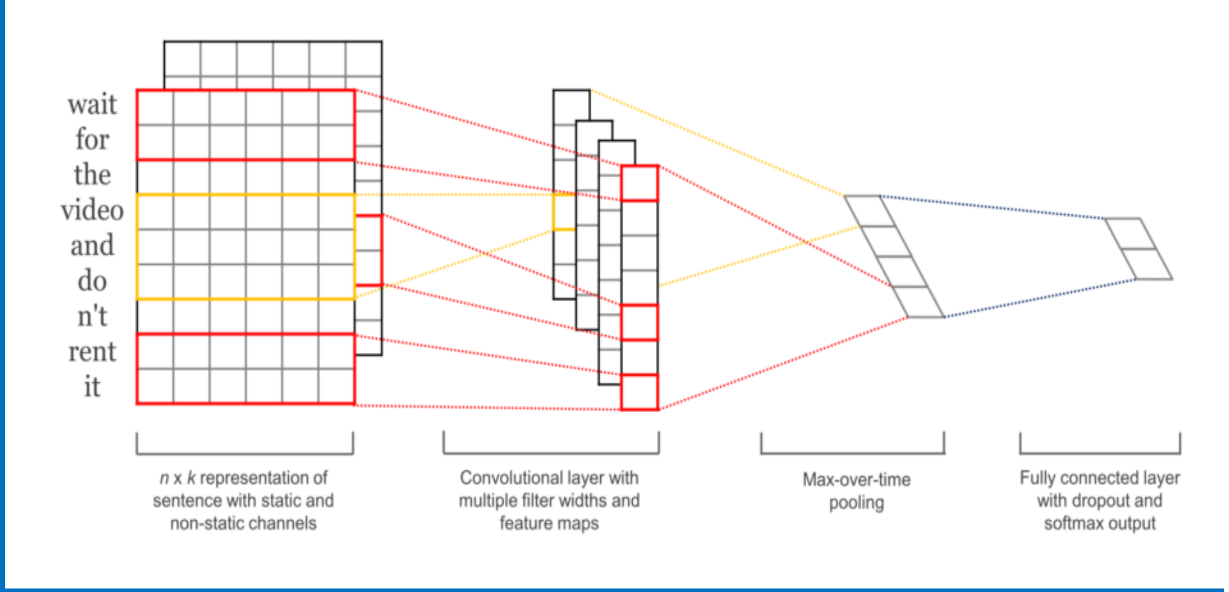
CNN

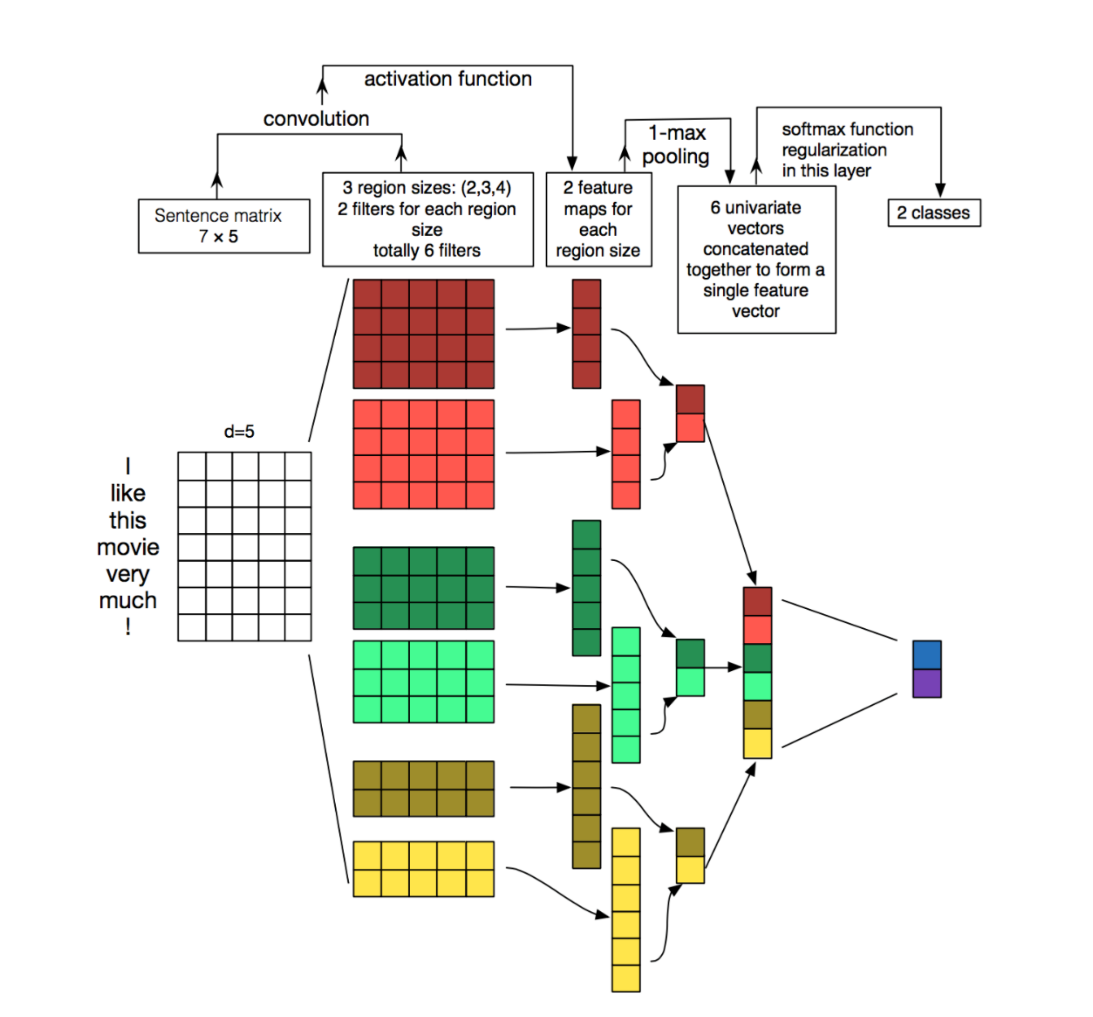


Eyeris' Deep Learning-based facial feature extraction and emotion classification using Convolutional Neural Networks.

###출처: 영상 인식 회사인 'Eyeris'의 감정 인식 추출 과정 (http://www.eyeris.ai/)

CNN

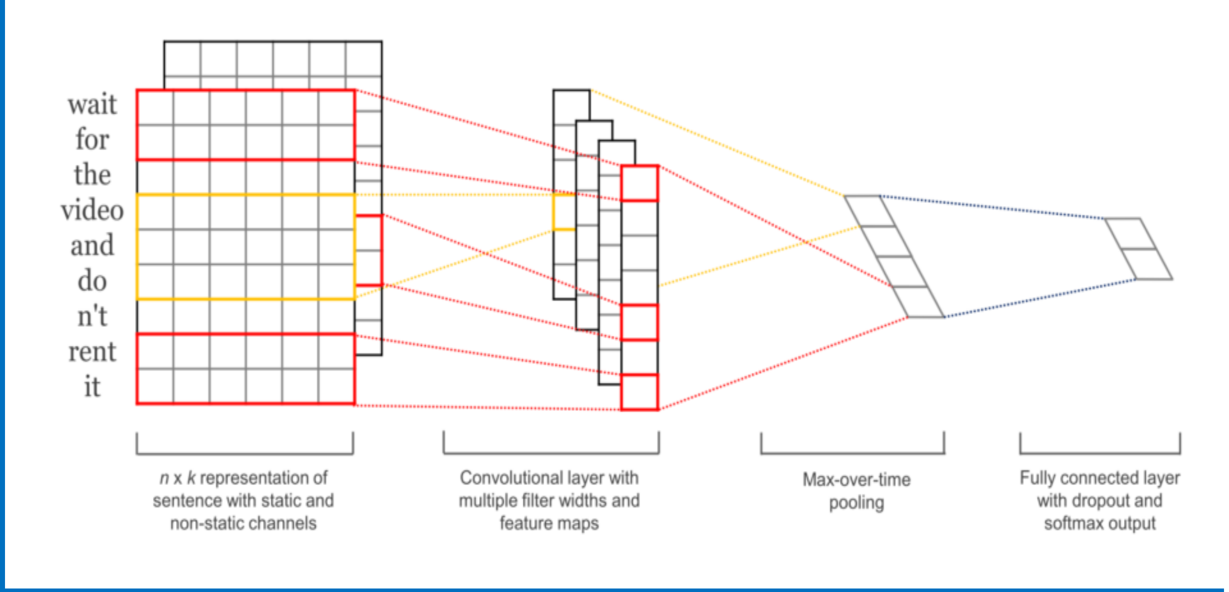

wait
for
the
video
and
do
n't
rent
it
n x k representation of sentence with static and non-static channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

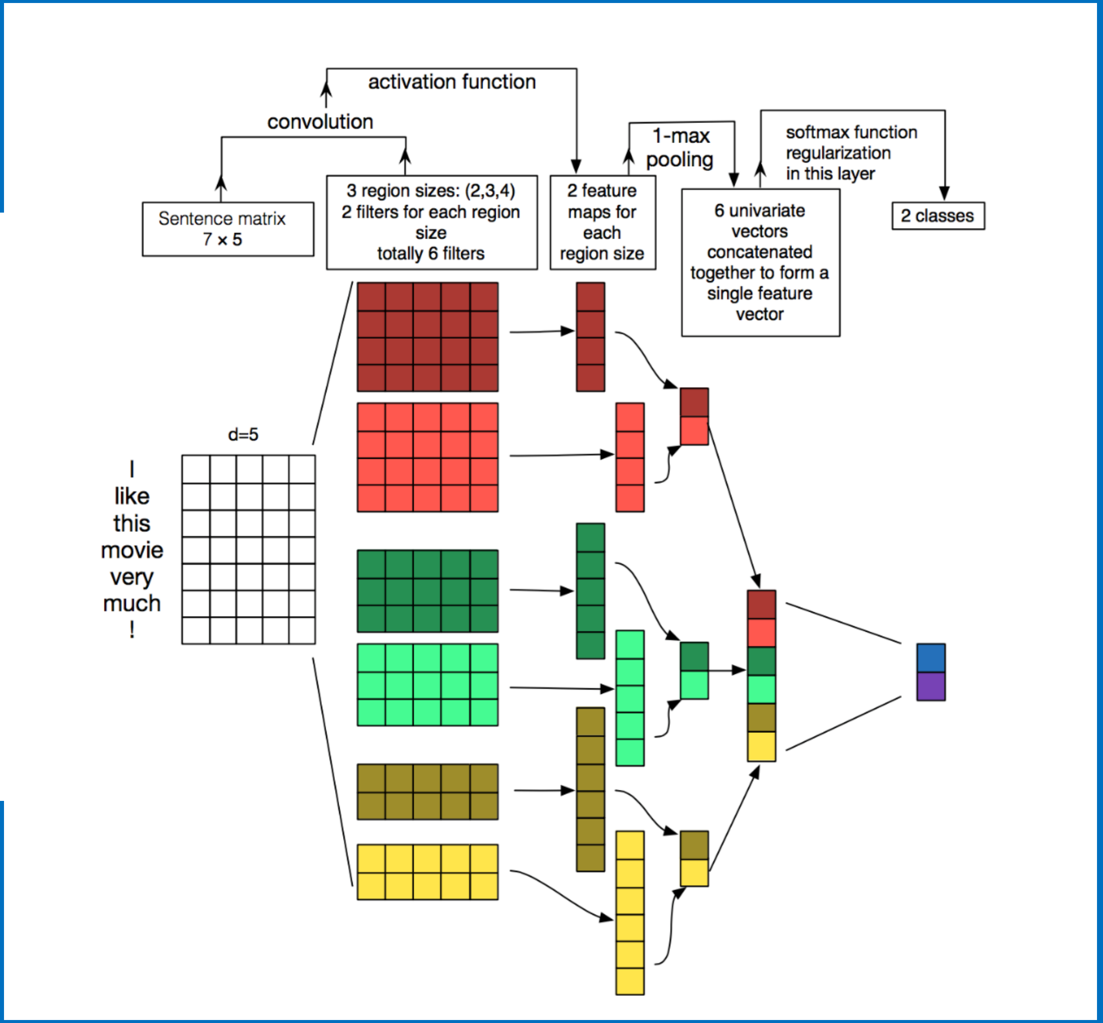

activation function
convolution
1-max pooling
softmax function regularization in this layer
Sentence matrix 7 × 5
3 region sizes: (2,3,4) 2 filters for each region size totally 6 filters
2 feature maps for each region size
6 univariate vectors concatenated together to form a single feature vector
2 classes
d=5
I
like
this
movie
very
much
!

Convolutional Neural Networks for Sentence Classification (Kim, Y. 2014)

Sentence classification with CNNs (Zhang and Wallace, 2015)
http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

CNN


wait
for
the
video
and
do
n't
rent
it
n x k representation of sentence with static and non-static channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

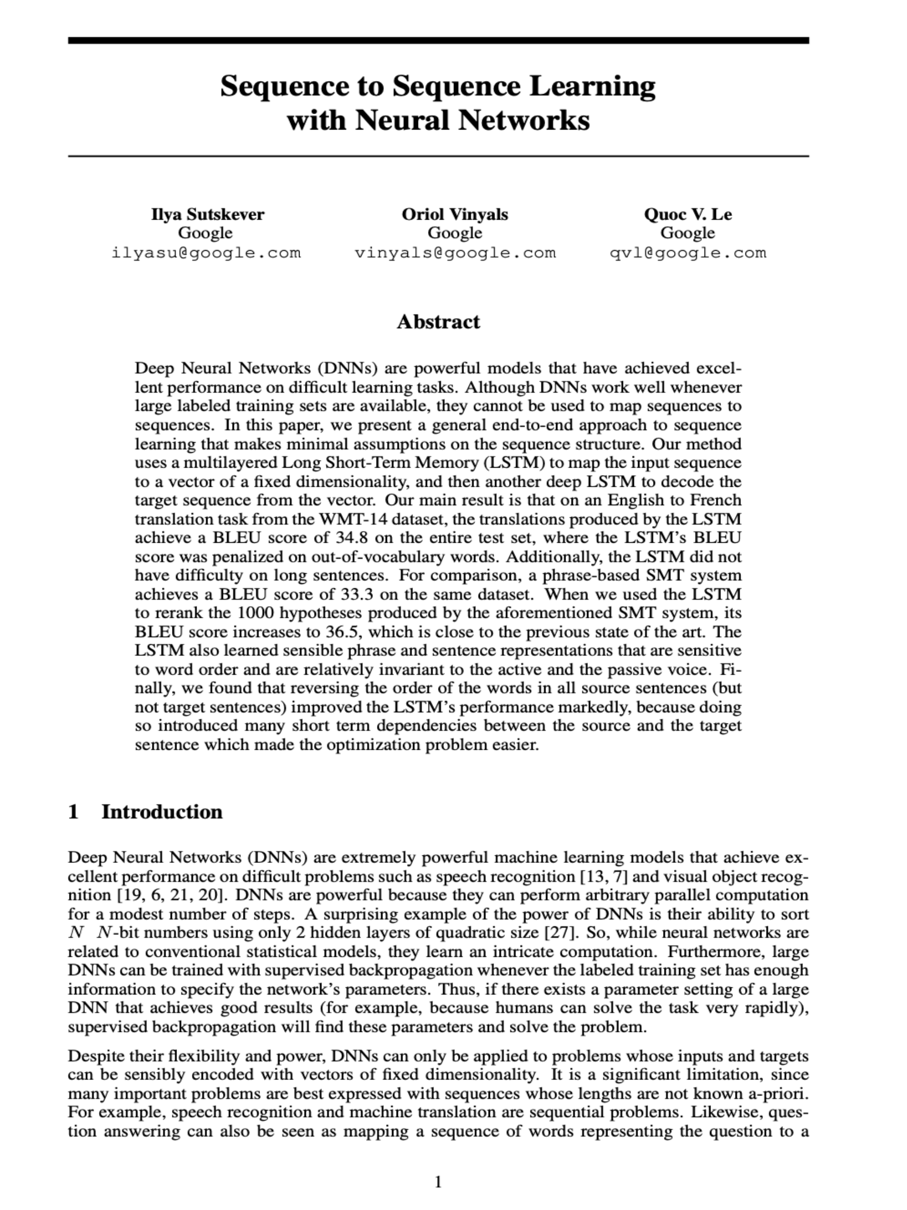

activation function
convolution
Sentence matrix 7 x 5
3 region sizes: (2,3,4)
2 filters for each region size
totally 6 filters
2 feature maps for each region size
1-max pooling
6 univariate vectors concatenated together to form a single feature vector
softmax function regularization in this layer
2 classes
d=5
I
like
this
movie
very
much
!

Convolutional Neural Networks for Sentence Classification (Kim, Y. 2014)

Sentence classification with CNNs (Zhang and Wallace, 2015)
http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/

Encoder-Decoder

- Sutskever et al. (2014, Google)
- 본격적인 NMT 모델의 시작
- RNN 을 사용해 Input 과 output을 modeling함



Sequence to Sequence Learning with Neural Networks

Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

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 state of the art. 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.

1 Introduction

Deep Neural Networks (DNNs) are extremely powerful machine learning models that achieve excellent performance on difficult problems such as speech recognition [13, 7] and visual object recognition [19, 6, 21, 20]. DNNs are powerful because they can perform arbitrary parallel computation for a modest number of steps. A surprising example of the power of DNNs is their ability to sort N N-bit numbers using only 2 hidden layers of quadratic size [27]. So, while neural networks are related to conventional statistical models, they learn an intricate computation. Furthermore, large DNNs can be trained with supervised backpropagation whenever the labeled training set has enough information to specify the network's parameters. Thus, if there exists a parameter setting of a large DNN that achieves good results (for example, because humans can solve the task very rapidly), supervised backpropagation will find these parameters and solve the problem.

Despite their flexibility and power, DNNs can only be applied to problems whose inputs and targets can be sensibly encoded with vectors of fixed dimensionality. It is a significant limitation, since many important problems are best expressed with sequences whose lengths are not known a-priori. For example, speech recognition and machine translation are sequential problems. Likewise, question answering can also be seen as mapping a sequence of words representing the question to a

Encoder-Decoder

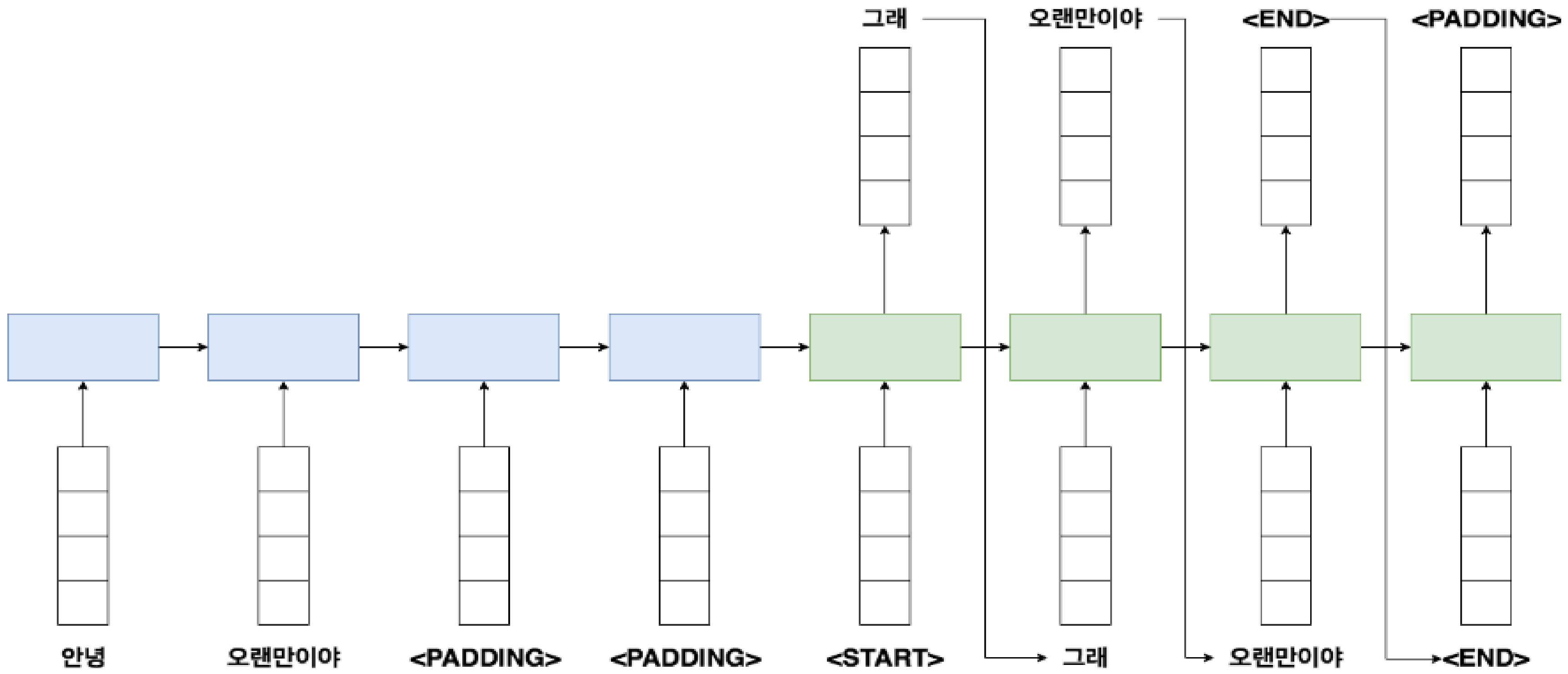


그림. 시퀀스 투 시퀀스 한글 시각화

Encoder-Decoder

이러저러한 이유로 엄마가 산타에게 키스하는 그런 장면을 목격했던 것도 아니었지만 어린 나이에 크리스마스에만 일하는 그 영감의 존재를 이상하게 생각했던 매우 똑똑한 아이였던 내가, 어쩐 일인지 우주인이니, 미래에서 온 사람이니, 유령이니, 요괴니, 초능력이니, 악의 조직이니 하는 것들과 싸우는 애니메이션, 특촬물, 만화의 히어로들이 이 세상에 존재하지 않는다는 사실을 깨달은 것은 상당히 시간이 지난 뒤의 일이었다. <EOS>

- <스즈미야 하루히의 우울> 中 쿈의 독백 중에서

텐서플로우와 머신러닝으로 시작하는 자연어처리

Encoder-Decoder

이러저러한 이유로 엄마가 산타에게 키스하는 그런 장면을 목격했던 것도 아니었지만 어린 나이에 크리스마스에만 일하는 그 영감의 존재를 이상하게 생각했던 매우 똑똑한 아이였던 내가, 어쩐 일인지 우주인이니, 미래에서 온 사람이니, 유령이니, 요괴니, 초능력이니, 악의 조직이니 하는 것들과 싸우는 애니메이션, 특촬물, 만화의 히어로들이 이 세상에 존재하지 않는다는 사실을 깨달은 것은 상당히 시간이 지난 뒤의 일이었다. <EOS>

- <스즈미야 하루히의 우울> 中 쿈의 독백 중에서

텐서플로우와 머신러닝으로 시작하는 자연어처리

Encoder-Decoder

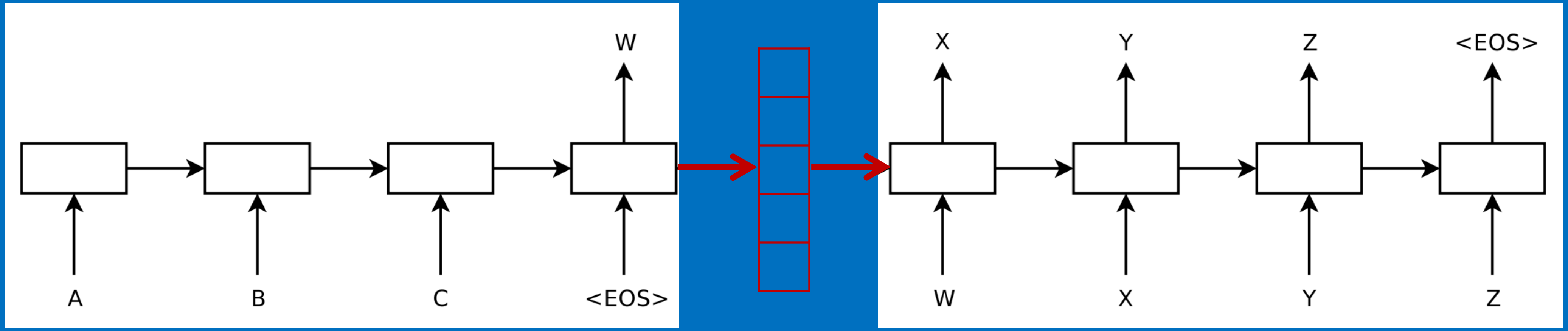
이러저러한 이유로 엄마가 산타에게 키스하는 그런 장면을 목격했던 것도 아니었지만 어린 나이에 크리스마스에만 일하는 그 영감의 존재를 이상하게 생각했던 매우 똑똑한 아이였던 내가, 어쩐 일인지 우주인이니, 미래에서 온 사람이니, 유령이니, 요괴니, 초능력이니, 악의 조직이니 하는 것들과 싸우는 애니메이션, 특촬물, 만화의 히어로들이 이 세상에 존재하지 않는다는 사실을 깨달은 것은 상당히 시간이 지난 뒤의 일이었다. <EOS>

- <스즈미야 하루히의 우울> 中 쿈의 독백 중에서

텐서플로우와 머신러닝으로 시작하는 자연어처리

LSTM을 사용한다면?

Encoder-Decoder

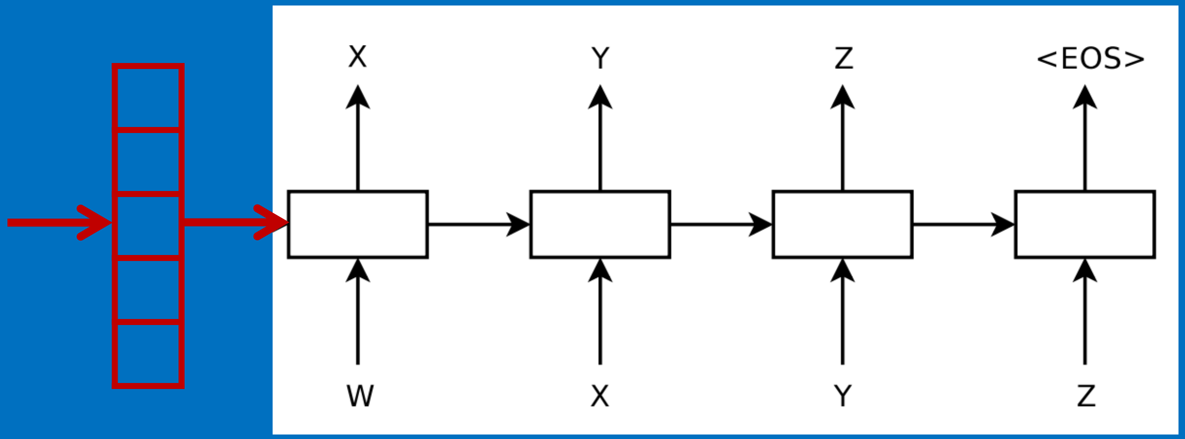


모든 Input Sequence의 정보를 하나의 fixed-size vector에 압축해서 담고 있음

Encoder-Decoder

Input

The Sequence to Sequence model is good

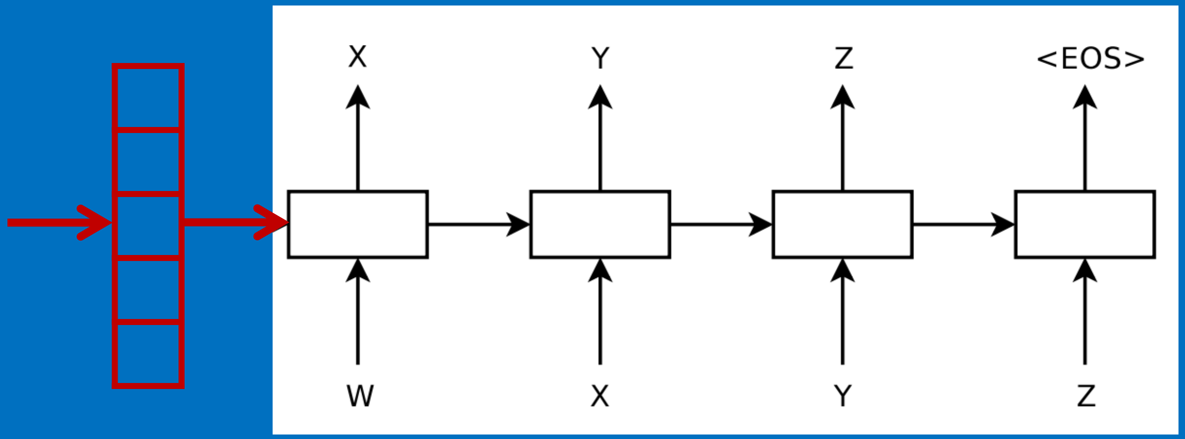


모든 Input Sequence의 정보를 하나의 fixed-size vector에 압축해서 담고 있음

Encoder-Decoder

Input

The Sequence to Sequence which is made by Cho et al. at NYU University model is good for Machine Translation



모든 Input Sequence의 정보를 하나의 fixed-size vector에 압축해서 담고 있음

Encoder-Decoder

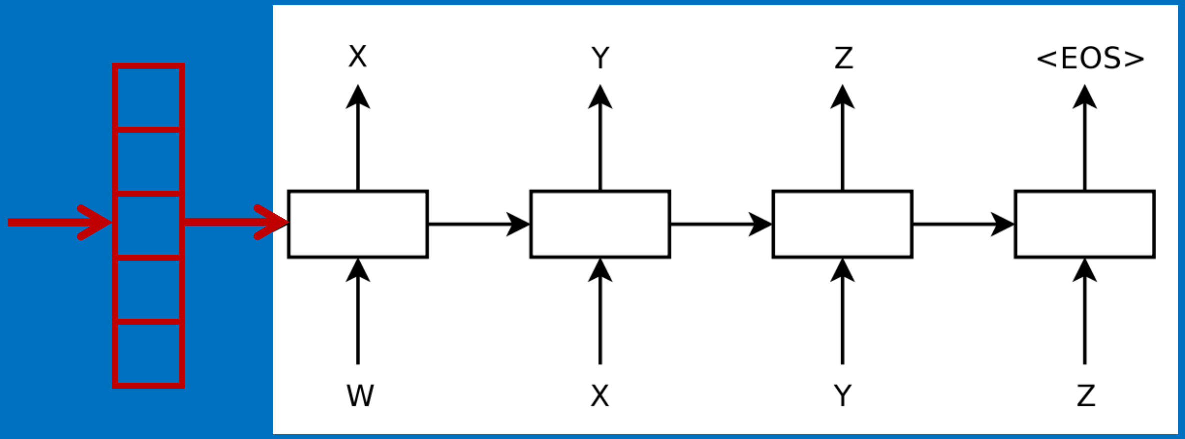
Input

The Sequence to Sequence which is

…

…

model is good for Machine Translation



모든 Input Sequence의 정보를 하나의 fixed-size vector에 압축해서 담고 있음

Encoder-Decoder

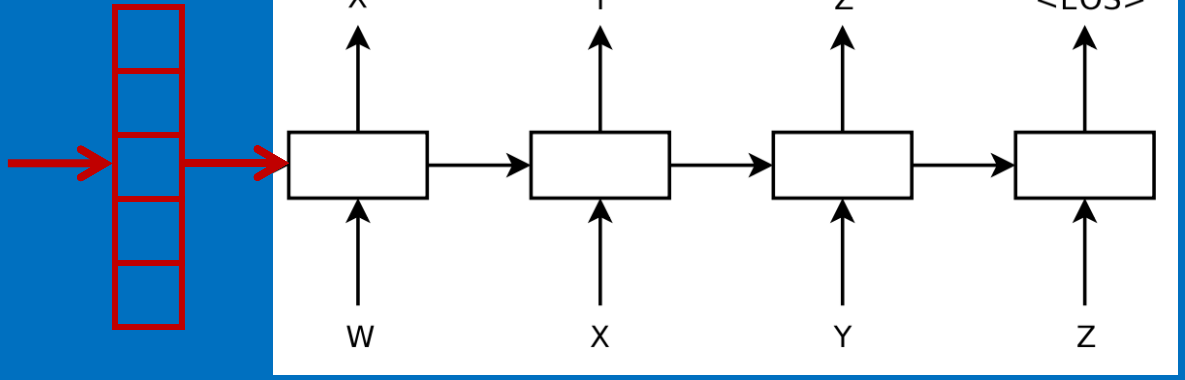
Input

The Sequence to Sequence which is

…

…

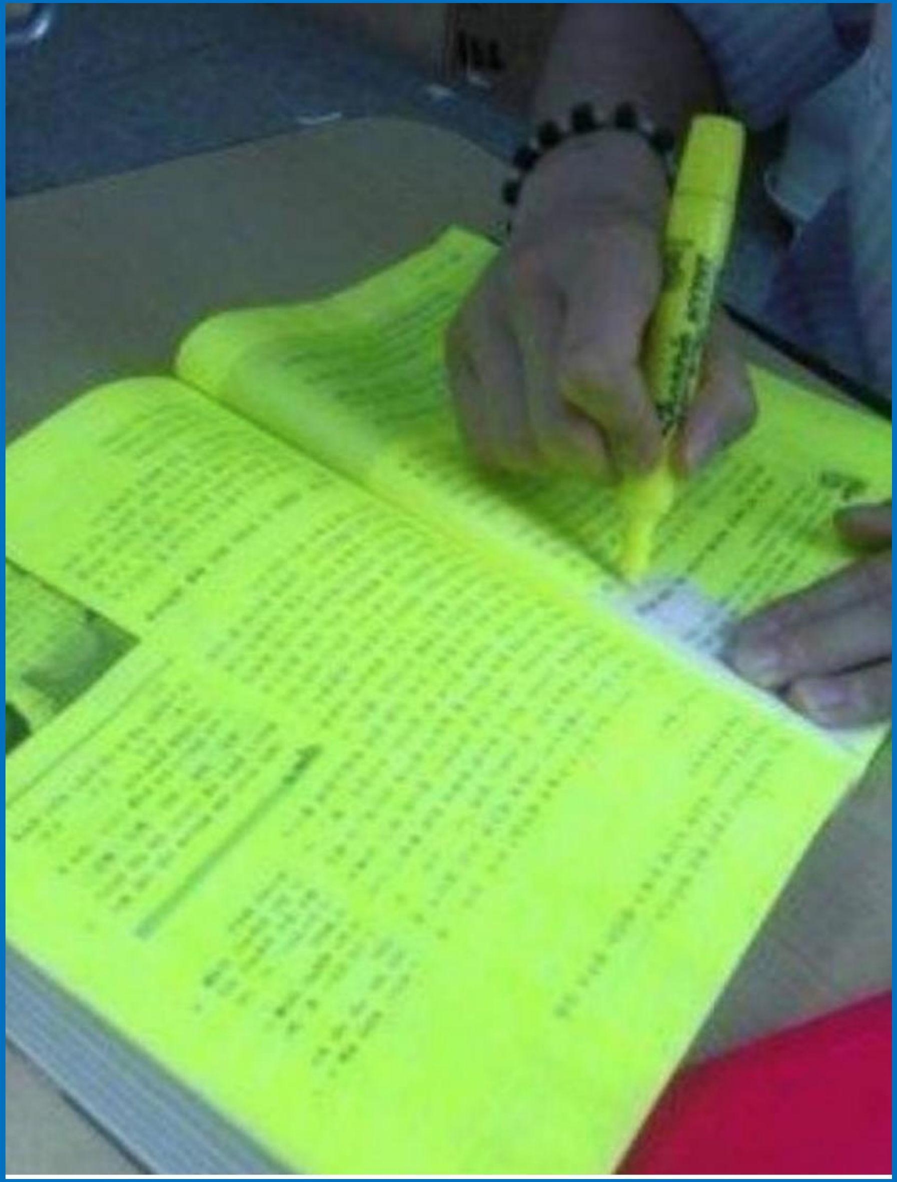
model is good for Machine Translation



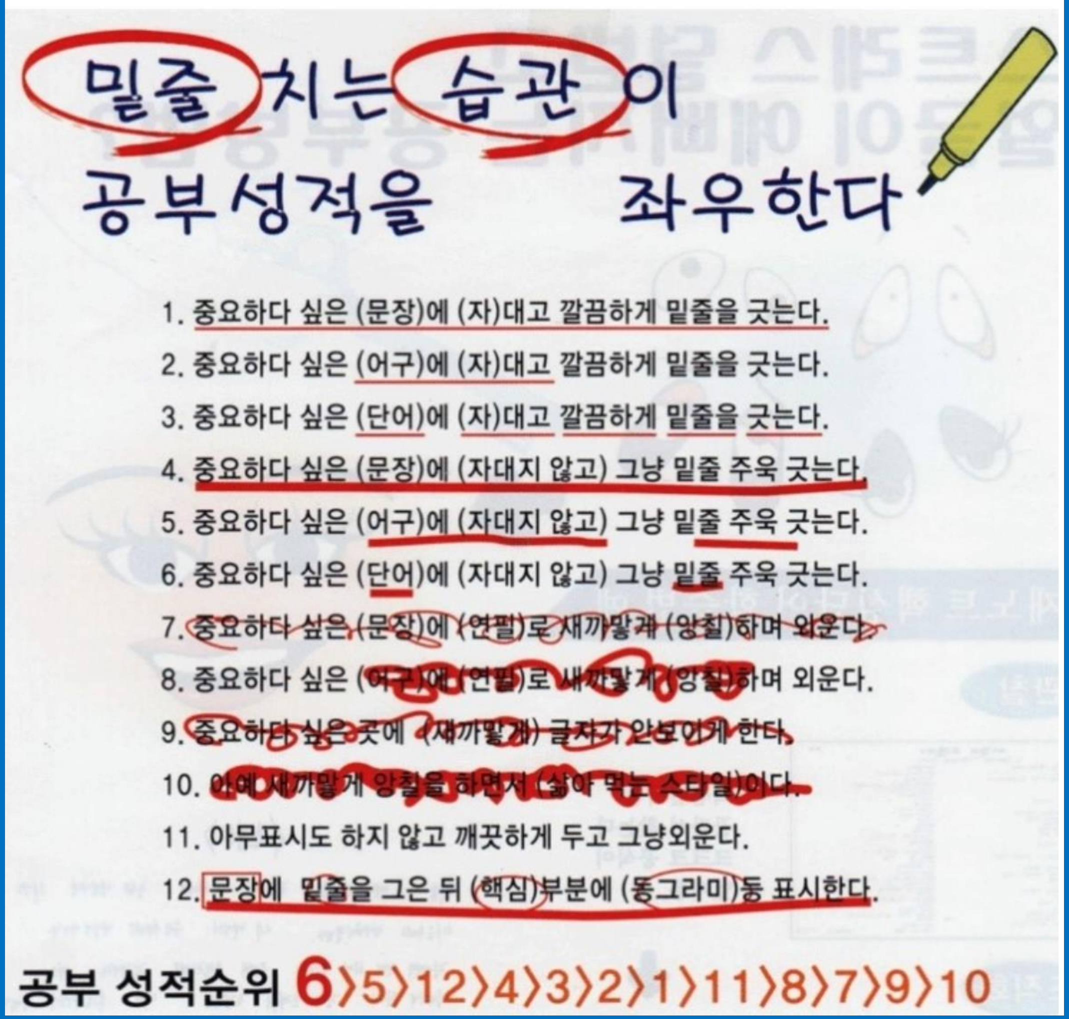
모든 Input Sequence의 정보를 하나의 fixed-size vector에 압축해서 담고 있음

문장이 길어질수록 전체 정보를 하나의 벡터로 표현하기 어려워 짐

Attention



전국 1등의 공부 방법



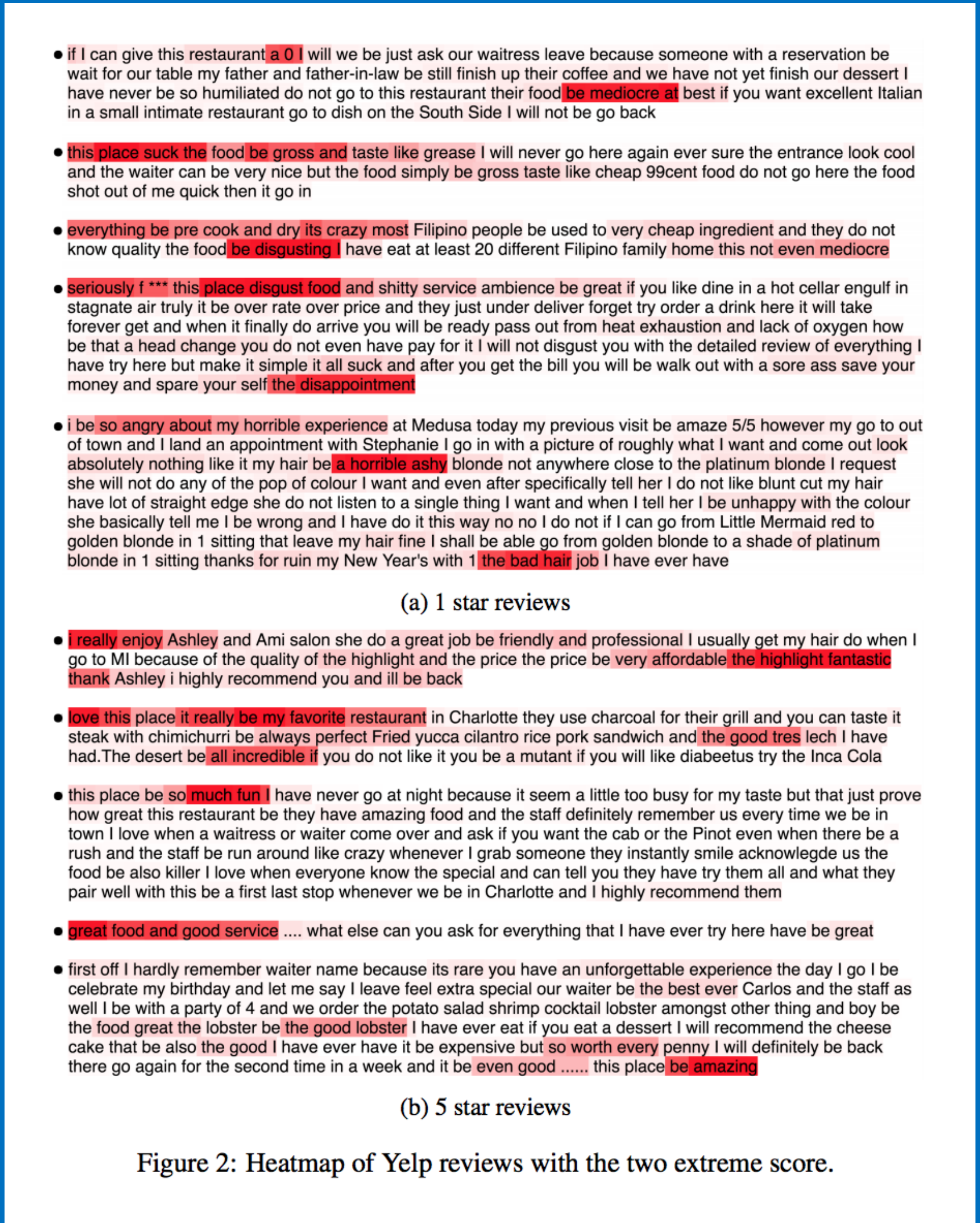
밑줄 치는 습관이
공부성적을 좌우한다

1. 중요하다 싶은 (문장)에 (자)대고 깔끔하게 밑줄을 긋는다.
2. 중요하다 싶은 (어구)에 (자)대고 깔끔하게 밑줄을 긋는다.
3. 중요하다 싶은 (단어)에 (자)대고 깔끔하게 밑줄을 긋는다.
4. 중요하다 싶은 (문장)에 (자대지 않고) 그냥 밑줄 주욱 긋는다.
5. 중요하다 싶은 (어구)에 (자대지 않고) 그냥 밑줄 주욱 긋는다.
6. 중요하다 싶은 (단어)에 (자대지 않고) 그냥 밑줄 주욱 긋는다.
7. 중요하다 싶은 (문장)에 (연필)로 새까맣게 (앙칠)하며 외운다.
8. 중요하다 싶은 (어구)에 (연필)로 새까맣게 (앙칠)하며 외운다.
9. 중요하다 싶은 곳에 (새까맣게) 글자가 안보이게 한다.
10. 아예 새까맣게 앙칠을 하면서 (삶아 먹는 스타일)이다.
11. 아무표시도 하지 않고 깨끗하게 두고 그냥외운다.
12. 문장에 밑줄을 그은 뒤 (핵심)부분에 (동그라미)등 표시한다.

공부 성적순위 6〉5〉12〉4〉3〉2〉1〉11〉8〉7〉9〉10

전국 1등의 공부 방법

Attention



- if I can give this restaurant a 0 I will we be just ask our waitress leave because someone with a reservation be wait for our table my father and father-in-law be still finish up their coffee and we have not yet finish our dessert I have never be so humiliated do not go to this restaurant their food be mediocre at best if you want excellent Italian in a small intimate restaurant go to dish on the South Side I will not be go back

- this place suck the food be gross and taste like grease I will never go here again ever sure the entrance look cool and the waiter can be very nice but the food simply be gross taste like cheap 99cent food do not go here the food shot out of me quick then it go in

- everything be pre cook and dry its crazy most Filipino people be used to very cheap ingredient and they do not know quality the food be disgusting I have eat at least 20 different Filipino family home this not even mediocre

- seriously f *** this place disgust food and shitty service ambience be great if you like dine in a hot cellar engulf in stagnate air truly it be over rate over price and they just under deliver forget try order a drink here it will take forever get and when it finally do arrive you will be ready pass out from heat exhaustion and lack of oxygen how be that a head change you do not even have pay for it I will not disgust you with the detailed review of everything I have try here but make it simple it all suck and after you get the bill you will be walk out with a sore ass save your money and spare your self the disappointment

- i be so angry about my horrible experience at Medusa today my previous visit be amaze 5/5 however my go to out of town and I land an appointment with Stephanie I go in with a picture of roughly what I want and come out look absolutely nothing like it my hair be a horrible ashy blonde not anywhere close to the platinum blonde I request she will not do any of the pop of colour I want and even after specifically tell her I do not like blunt cut my hair have lot of straight edge she do not listen to a single thing I want and when I tell her I be unhappy with the colour she basically tell me I be wrong and I have do it this way no no I do not if I can go from Little Mermaid red to golden blonde in 1 sitting that leave my hair fine I shall be able go from golden blonde to a shade of platinum blonde in 1 sitting thanks for ruin my New Year's with 1 the bad hair job I have ever have

(a) 1 star reviews

- i really enjoy Ashley and Ami salon she do a great job be friendly and professional I usually get my hair do when I go to MI because of the quality of the highlight and the price the price be very affordable the highlight fantastic thank Ashley i highly recommend you and ill be back

- love this place it really be my favorite restaurant in Charlotte they use charcoal for their grill and you can taste it steak with chimichurri be always perfect Fried yucca cilantro rice pork sandwich and the good tres lech I have had.The desert be all incredible if you do not like it you be a mutant if you will like diabeetus try the Inca Cola

- this place be so much fun I have never go at night because it seem a little too busy for my taste but that just prove how great this restaurant be they have amazing food and the staff definitely remember us every time we be in town I love when a waitress or waiter come over and ask if you want the cab or the Pinot even when there be a rush and the staff be run around like crazy whenever I grab someone they instantly smile acknowlegde us the food be also killer I love when everyone know the special and can tell you they have try them all and what they pair well with this be a first last stop whenever we be in Charlotte and I highly recommend them

- great food and good service .... what else can you ask for everything that I have ever try here have be great

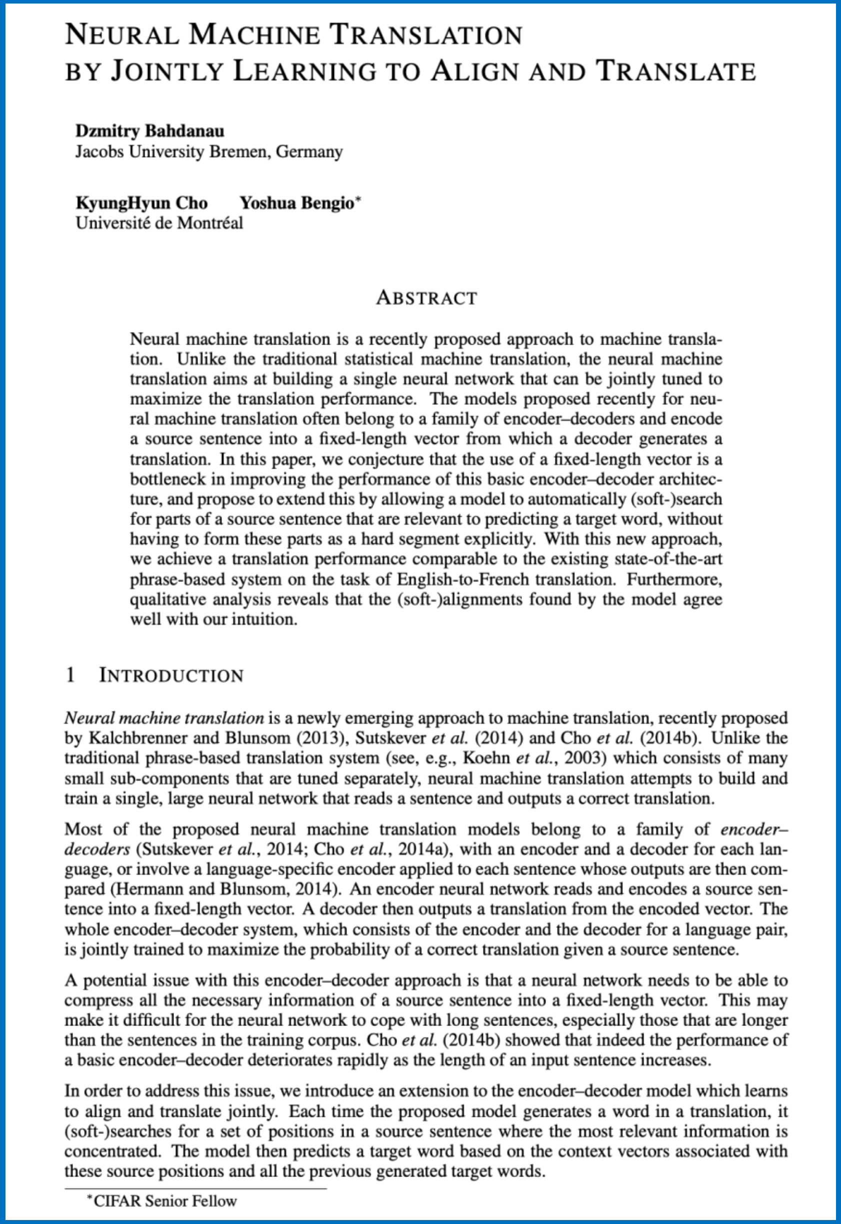
- first off I hardly remember waiter name because its rare you have an unforgettable experience the day I go I be celebrate my birthday and let me say I leave feel extra special our waiter be the best ever Carlos and the staff as well I be with a party of 4 and we order the potato salad shrimp cocktail lobster amongst other thing and boy be the food great the lobster be the good lobster I have ever eat if you eat a dessert I will recommend the cheese cake that be also the good I have ever have it be expensive but so worth every penny I will definitely be back there go again for the second time in a week and it be even good ...... this place be amazing

(b) 5 star reviews

Figure 2: Heatmap of Yelp reviews with the two extreme score.

Attention

- Bahdanau et al. (2016)
- Attention Mechanism을 처음 도입
- Bidirectional RNN을 사용함



NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE

Dzmitry Bahdanau
Jacobs University Bremen, Germany

KyungHyun Cho **Yoshua Bengio**[*]
Université de Montréal

ABSTRACT

Neural machine translation is a recently proposed approach to machine translation. Unlike the traditional statistical machine translation, the neural machine translation aims at building a single neural network that can be jointly tuned to maximize the translation performance. The models proposed recently for neural machine translation often belong to a family of encoder–decoders and encode a source sentence into a fixed-length vector from which a decoder generates a translation. In this paper, we conjecture that the use of a fixed-length vector is a bottleneck in improving the performance of this basic encoder–decoder architecture, and propose to extend this by allowing a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without having to form these parts as a hard segment explicitly. With this new approach, we achieve a translation performance comparable to the existing state-of-the-art phrase-based system on the task of English-to-French translation. Furthermore, qualitative analysis reveals that the (soft-)alignments found by the model agree well with our intuition.

1 INTRODUCTION

Neural machine translation is a newly emerging approach to machine translation, recently proposed by Kalchbrenner and Blunsom (2013), Sutskever *et al.* (2014) and Cho *et al.* (2014b). Unlike the traditional phrase-based translation system (see, e.g., Koehn *et al.*, 2003) which consists of many small sub-components that are tuned separately, neural machine translation attempts to build and train a single, large neural network that reads a sentence and outputs a correct translation.

Most of the proposed neural machine translation models belong to a family of *encoder–decoders* (Sutskever *et al.*, 2014; Cho *et al.*, 2014a), with an encoder and a decoder for each language, or involve a language-specific encoder applied to each sentence whose outputs are then compared (Hermann and Blunsom, 2014). An encoder neural network reads and encodes a source sentence into a fixed-length vector. A decoder then outputs a translation from the encoded vector. The whole encoder–decoder system, which consists of the encoder and the decoder for a language pair, is jointly trained to maximize the probability of a correct translation given a source sentence.

A potential issue with this encoder–decoder approach is that a neural network needs to be able to compress all the necessary information of a source sentence into a fixed-length vector. This may make it difficult for the neural network to cope with long sentences, especially those that are longer than the sentences in the training corpus. Cho *et al.* (2014b) showed that indeed the performance of a basic encoder–decoder deteriorates rapidly as the length of an input sentence increases.

In order to address this issue, we introduce an extension to the encoder–decoder model which learns to align and translate jointly. Each time the proposed model generates a word in a translation, it (soft-)searches for a set of positions in a source sentence where the most relevant information is concentrated. The model then predicts a target word based on the context vectors associated with these source positions and all the previous generated target words.

[*] CIFAR Senior Fellow

Attention

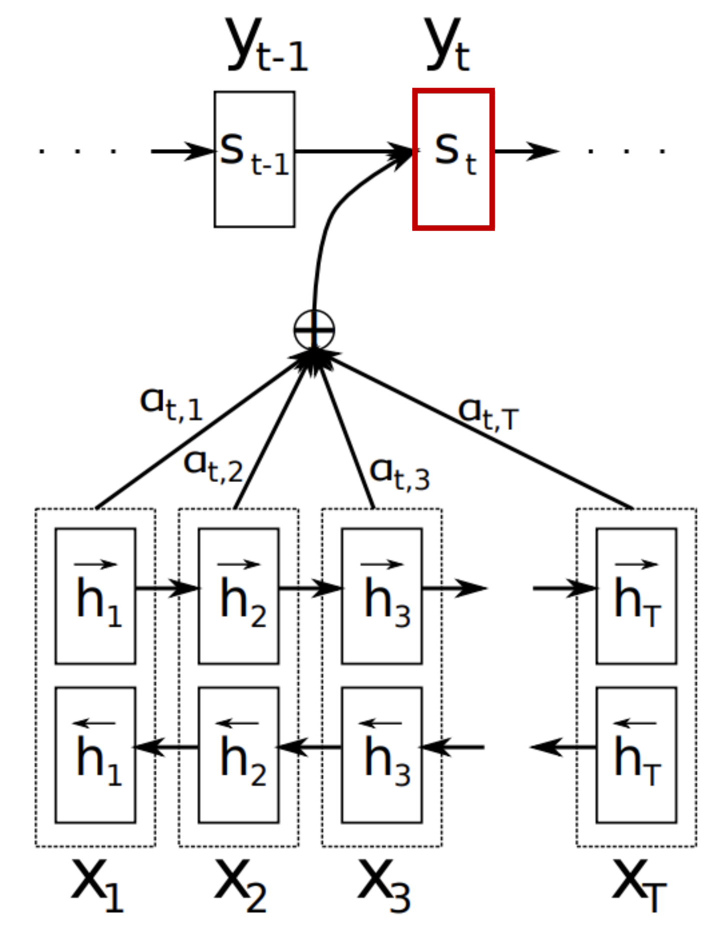
Attention Mechanism (alignment model)

hn : n번째 입력에 대한 hidden state vector

at,n : 시간 t에 대해서 n번째 단어에 대한 가중치

St : 시간 t에서 Decoder hidden state vector

Yt : 시간 t에서 Decoder 생성한 결과

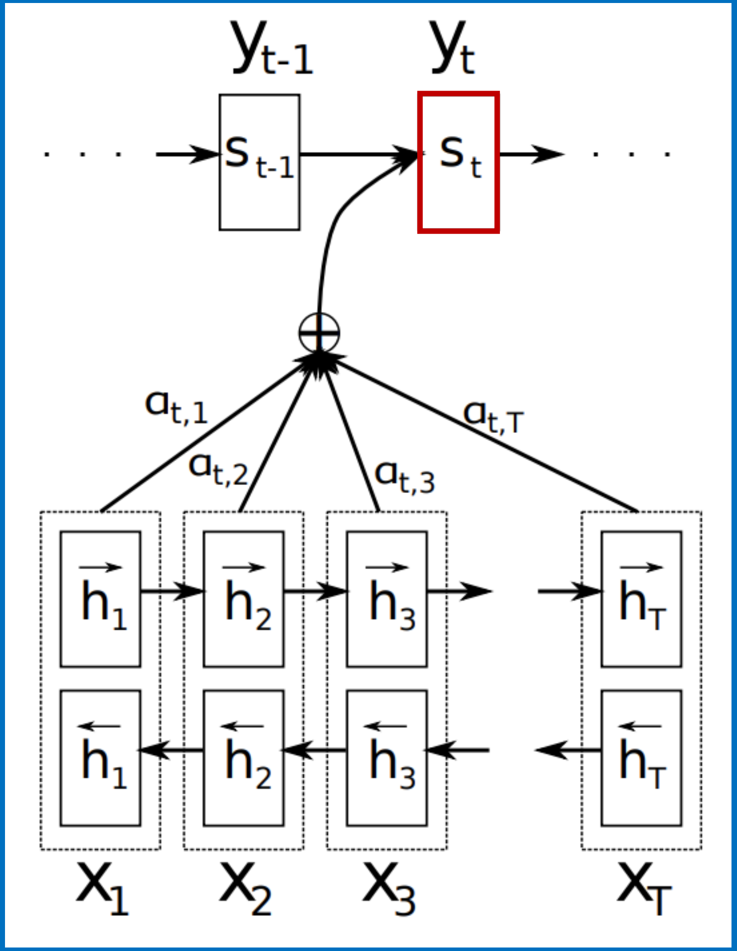


Attention

Attention Mechanism (alignment model)

$$s_t = f(s_{t-1}, y_{t-1}, c_t)$$

- s_t: *decoder hidden state*
- y_t: *decoder input*
- c_t: *context vector*

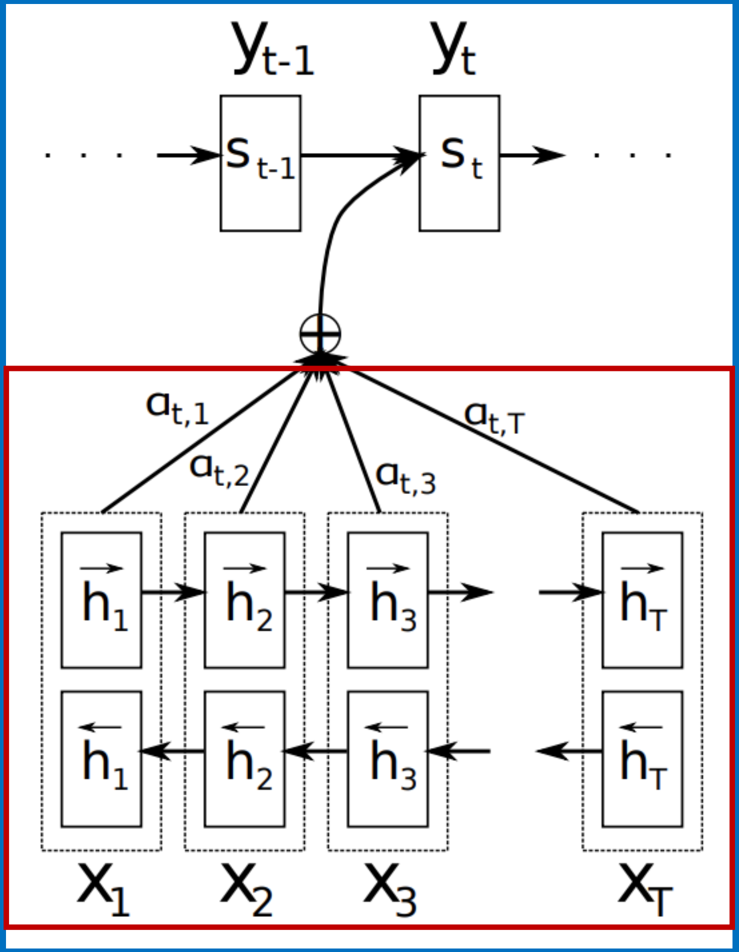

y_{t-1}
y_t
s_{t-1}
s_t
$a_{t,1}$
$a_{t,2}$
$a_{t,3}$
$a_{t,T}$
$\overrightarrow{h}_1$ $\overrightarrow{h}_2$ $\overrightarrow{h}_3$ $\overrightarrow{h}_T$
$\overleftarrow{h}_1$ $\overleftarrow{h}_2$ $\overleftarrow{h}_3$ $\overleftarrow{h}_T$
X_1 X_2 X_3 X_T

Attention

Attention Mechanism (alignment model)

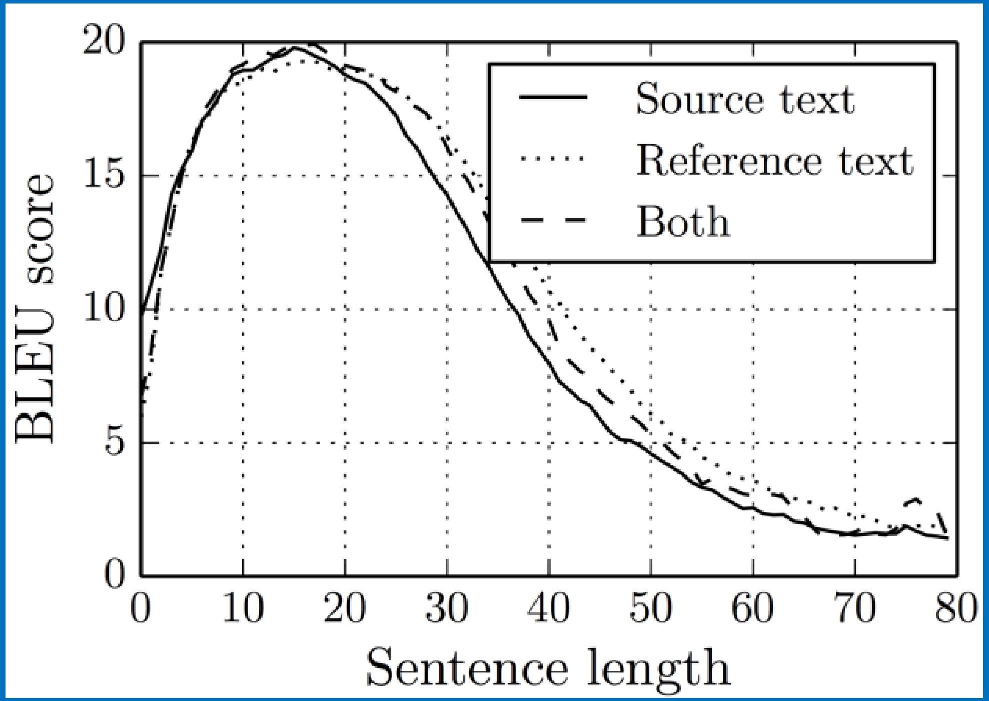
$$s_t = f(s_{i-1}, y_{i-1}, c_t)$$

$$c_i = \sum_{j=1}^{n} \alpha_{tj} h_j$$



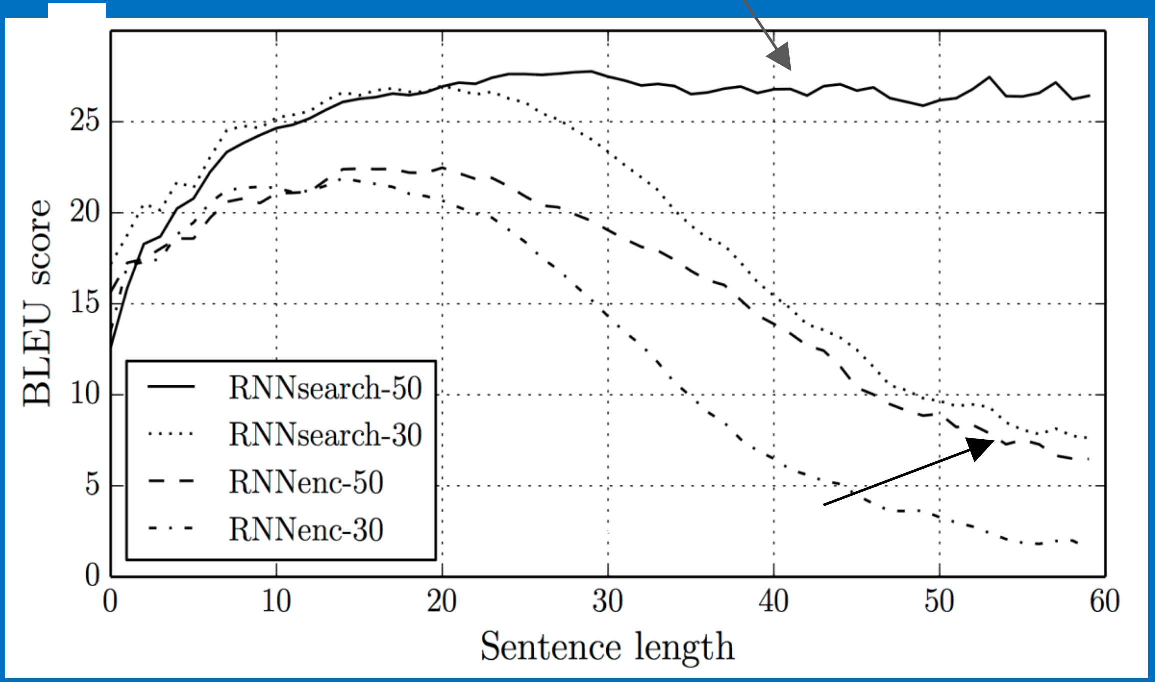
Attention

NMT with LSTMs



NMT with LSTMs + attention

with attention



Bahdanau et al. 2014
Slide: Text generation with attention, GTC 2017, Valentin Malykh (2017)

Sentence Representation

Sentence Representation

Sentence Embedding

ELMO

GPT

BERT

XLNET

RoBERTa

Sentence Representation

들어 가기전에…

전이 학습 (Transfer Learning)

언어 모델 (Language Model)

전이 학습 (Transfer Learning)

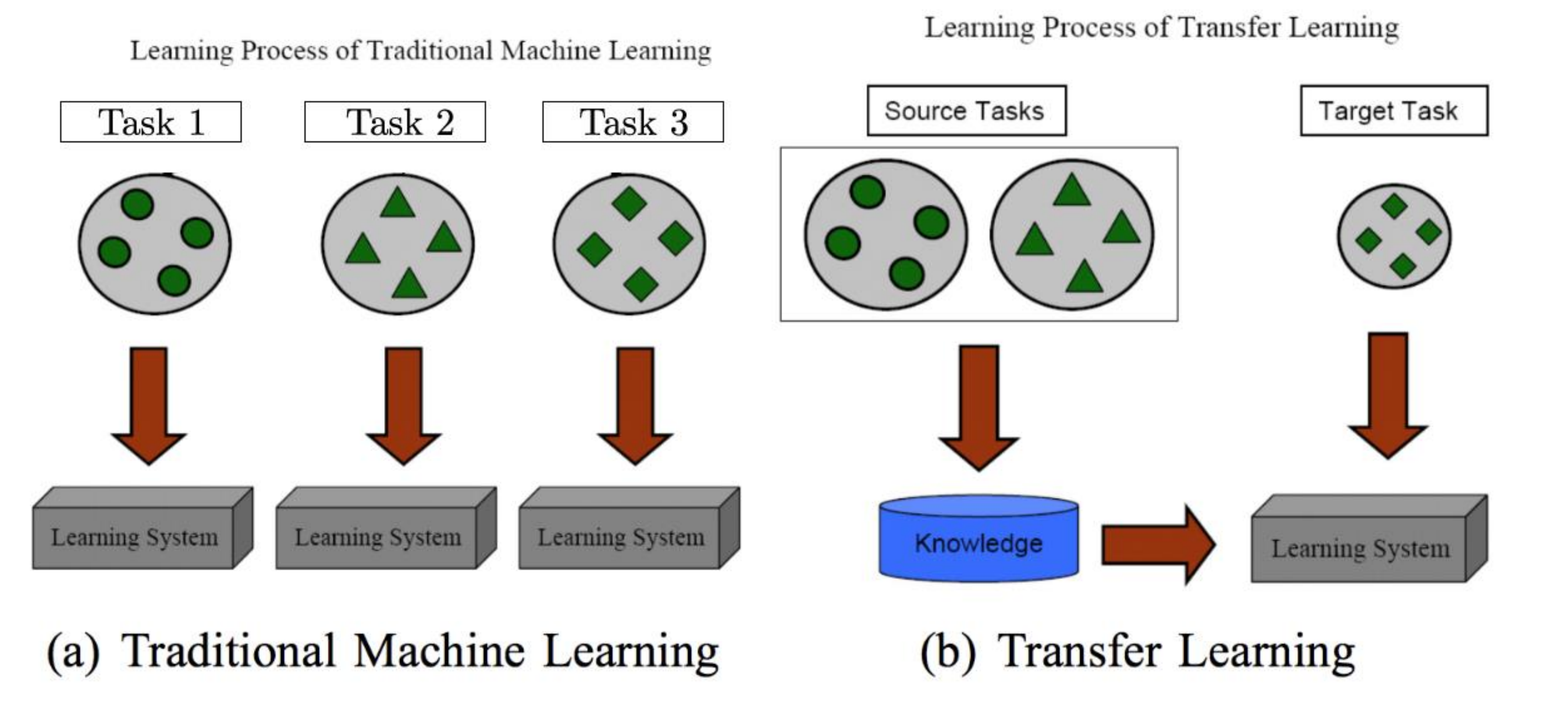
사람도 무언가를 배울때 다양한 배경지식을 바탕으로 정보를 습득

전이학습 모델 역시, 대규모 말뭉치를 미리 학습 (Pre-training)한

단어의 의미적 문법적 관계 정보등을 활용해, 다양한 Task에 활용 (Fine-tuning)

-> 빵 반죽 & 발효를 잘 해 놓으면, 피자, 케이크, 식빵 등 어떤 빵을 만들어도 맛

전이 학습 (Transfer Learning)

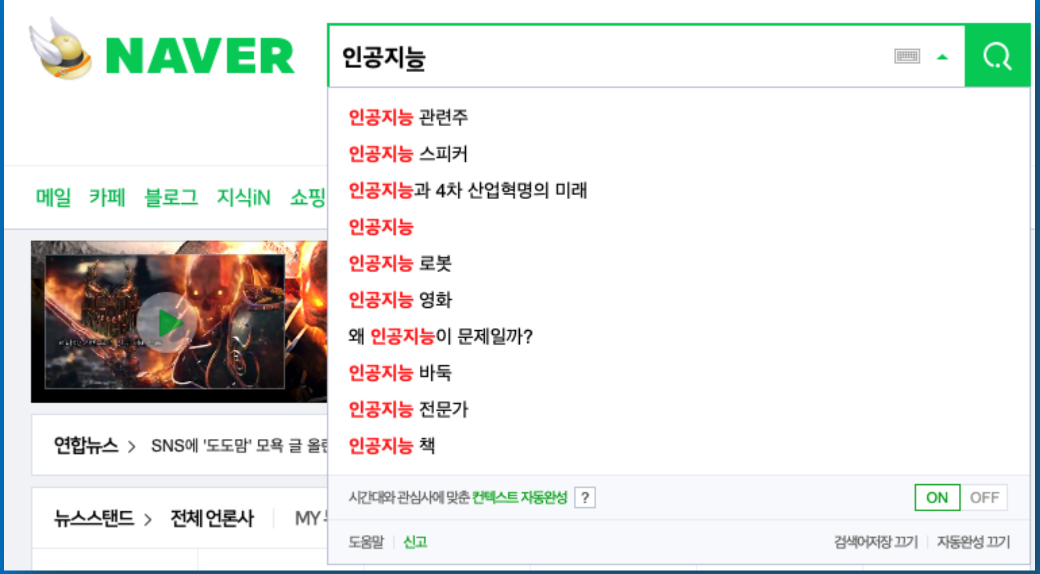


(a) Traditional Machine Learning (b) Transfer Learning

언어 모델 (Language Model)

문장의 확률을 예측하거나, 이전 단어를 기반으로, 다음 단어가 나올 확률 예측

예) 요즘 대세 딥러닝 프레임워크 하면 ________ 이다.



인공지능

인공지능 관련주
인공지능 스피커
인공지능과 4차 산업혁명의 미래
인공지능
인공지능 로봇
인공지능 영화
왜 인공지능이 문제일까?
인공지능 바둑
인공지능 전문가
인공지능 책

시간대와 관심사에 맞춘 컨텍스트 자동완성 ?

ELMO

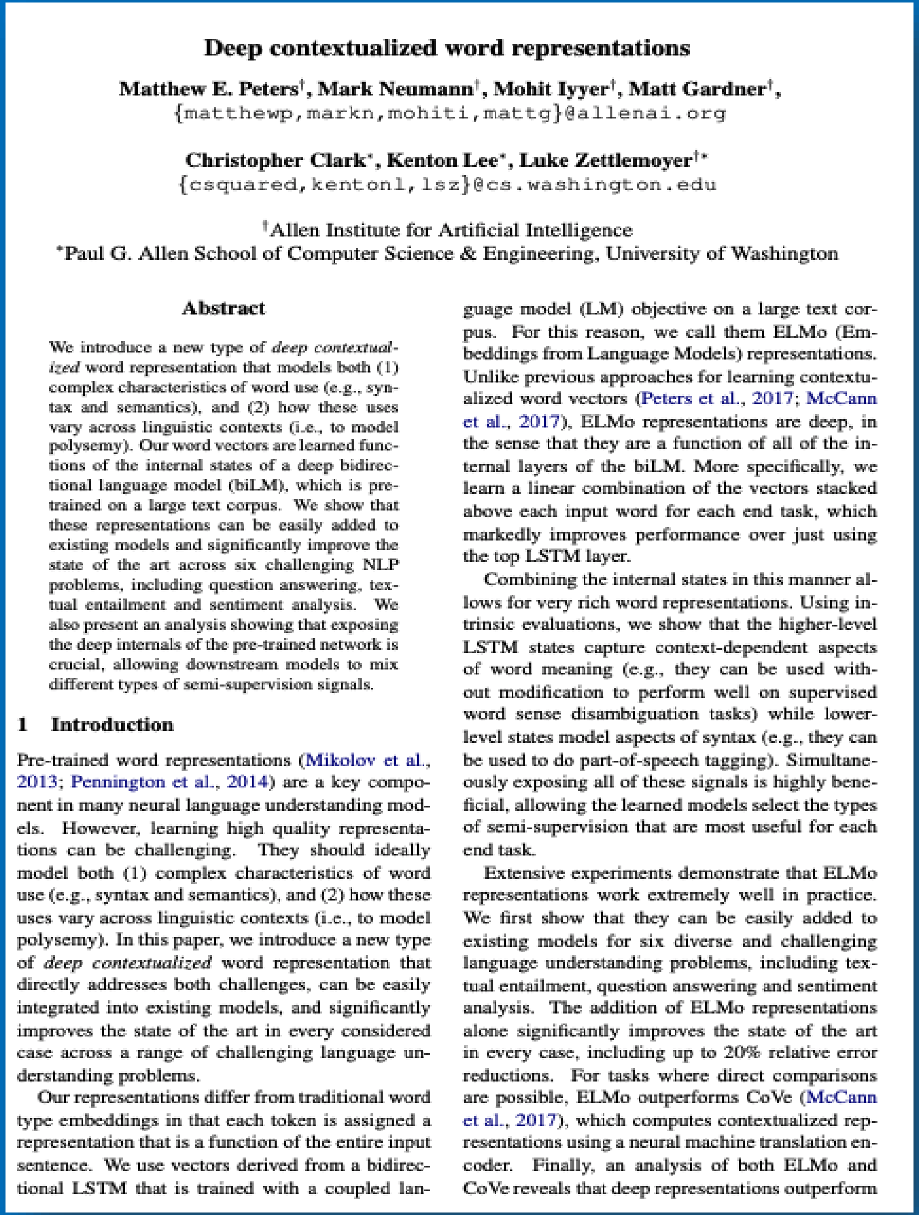
Peters et al., 2018

ELMO (Embeddings from Language Models)

미국 연구기관 Allen AI & 워싱턴대학 공동연구팀 발표

Context 정보를 반영하는 방법 제시

*전이 학습 (Transfer learning)을 접목해 주목



Deep contextualized word representations

Matthew E. Peters†, Mark Neumann†, Mohit Iyyer†, Matt Gardner†,
{matthewp,markn,mohiti,mattg}@allenai.org

Christopher Clark*, Kenton Lee*, Luke Zettlemoyer†*
{csquared,kentonl,lsz}@cs.washington.edu

†Allen Institute for Artificial Intelligence
*Paul G. Allen School of Computer Science & Engineering, University of Washington

Abstract

We introduce a new type of *deep contextualized* word representation that models both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). Our word vectors are learned functions of the internal states of a deep bidirectional language model (biLM), which is pre-trained on a large text corpus. We show that these representations can be easily added to existing models and significantly improve the state of the art across six challenging NLP problems, including question answering, textual entailment and sentiment analysis. We also present an analysis showing that exposing the deep internals of the pre-trained network is crucial, allowing downstream models to mix different types of semi-supervision signals.

1 Introduction

Pre-trained word representations (Mikolov et al., 2013; Pennington et al., 2014) are a key component in many neural language understanding models. However, learning high quality representations can be challenging. They should ideally model both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). In this paper, we introduce a new type of *deep contextualized* word representation that directly addresses both challenges, can be easily integrated into existing models, and significantly improves the state of the art in every considered case across a range of challenging language understanding problems.

Our representations differ from traditional word type embeddings in that each token is assigned a representation that is a function of the entire input sentence. We use vectors derived from a bidirectional LSTM that is trained with a coupled language model (LM) objective on a large text corpus. For this reason, we call them ELMo (Embeddings from Language Models) representations. Unlike previous approaches for learning contextualized word vectors (Peters et al., 2017; McCann et al., 2017), ELMo representations are deep, in the sense that they are a function of all of the internal layers of the biLM. More specifically, we learn a linear combination of the vectors stacked above each input word for each end task, which markedly improves performance over just using the top LSTM layer.

Combining the internal states in this manner allows for very rich word representations. Using intrinsic evaluations, we show that the higher-level LSTM states capture context-dependent aspects of word meaning (e.g., they can be used without modification to perform well on supervised word sense disambiguation tasks) while lower-level states model aspects of syntax (e.g., they can be used to do part-of-speech tagging). Simultaneously exposing all of these signals is highly beneficial, allowing the learned models select the types of semi-supervision that are most useful for each end task.

Extensive experiments demonstrate that ELMo representations work extremely well in practice. We first show that they can be easily added to existing models for six diverse and challenging language understanding problems, including textual entailment, question answering and sentiment analysis. The addition of ELMo representations alone significantly improves the state of the art in every case, including up to 20% relative error reductions. For tasks where direct comparisons are possible, ELMo outperforms CoVe (McCann et al., 2017), which computes contextualized representations using a neural machine translation encoder. Finally, an analysis of both ELMo and CoVe reveals that deep representations outperform

*전이 학습: 이미 학습된 모델을 다른 딥러닝의 모델의 입력값 또는 부분으로 재사용

ELMO

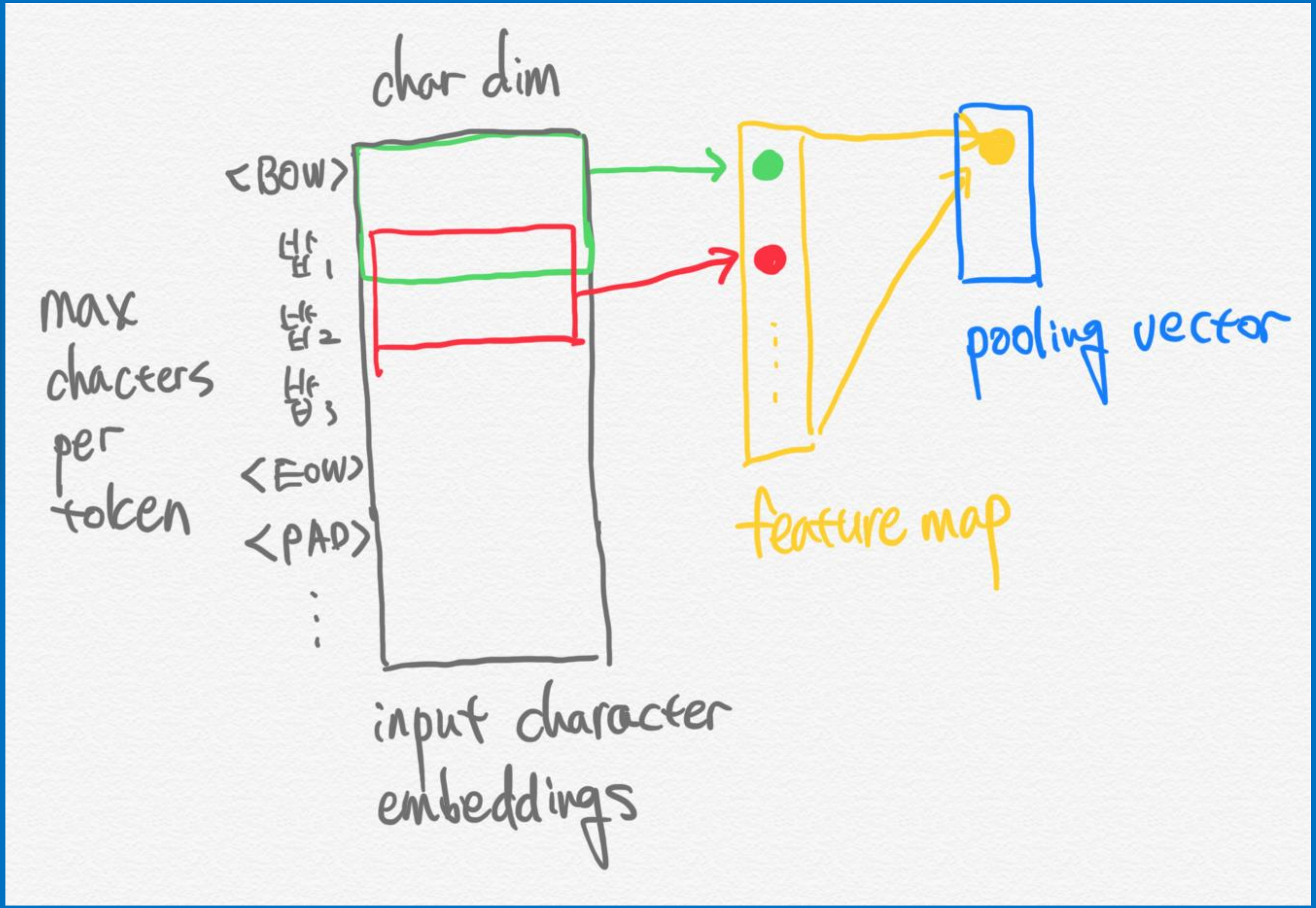
ELMO의 구성요소

문자 단위 컨볼루션 신경망 (Character Convolution Neural Network)

양방향 LSTM 레이어 (Bi-directional LSTM Layer)

ELMO 레이어

문자 단위 컨볼루션 신경망 (Char-CNN)



ELMO의 입력은 Char:

대응하는 유니코드 ID로 치환

Convolution Filter는 2개 문자

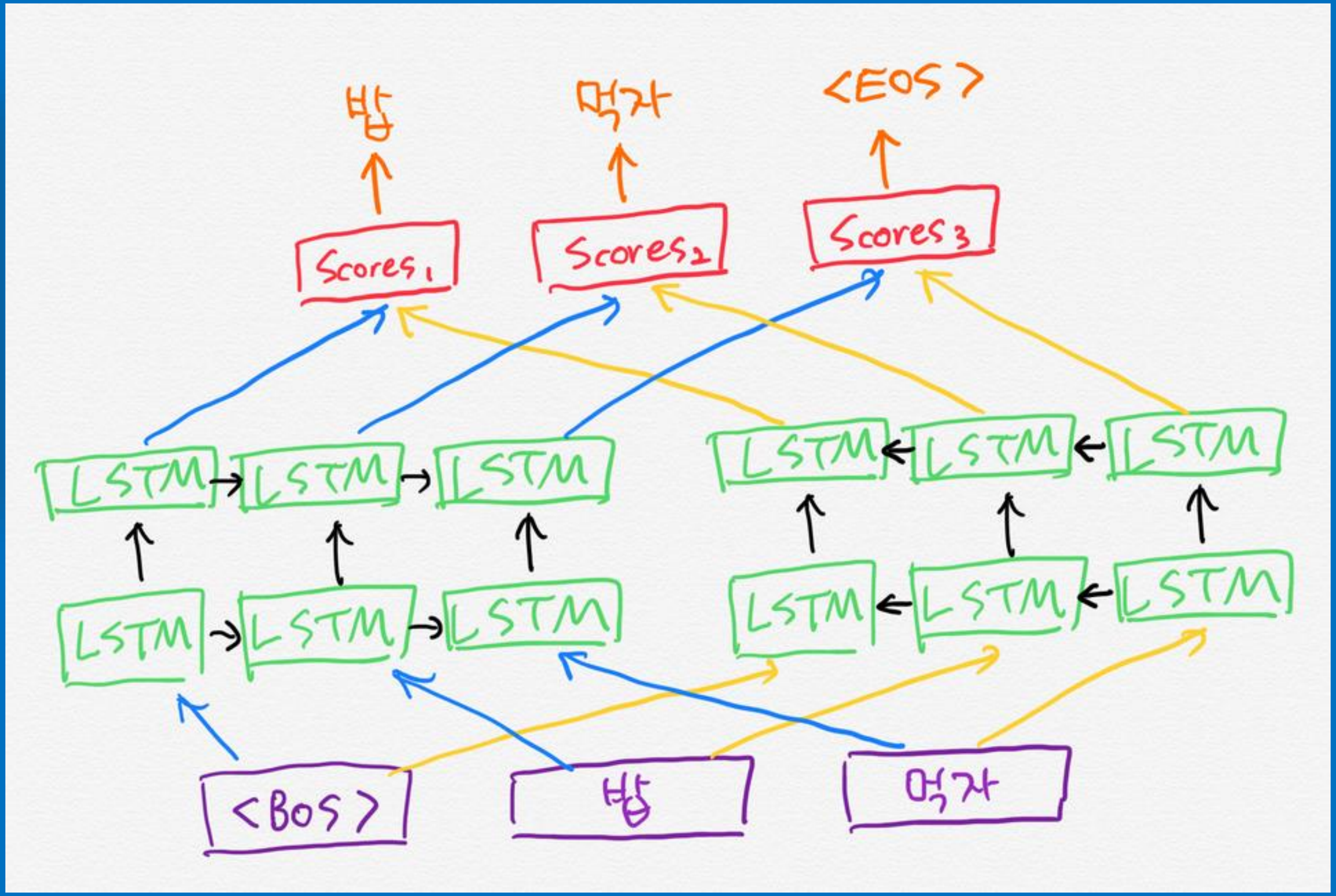
임베딩의 차원수

단어의 변형에 강건하고,

OOV에도 견고

이기창님의 한국어 임베딩

양방향 LSTM 레이어 (Bi-LSTM Layer)



문자 단위 컨볼루션 신경망

입력을 바탕으로 양방향

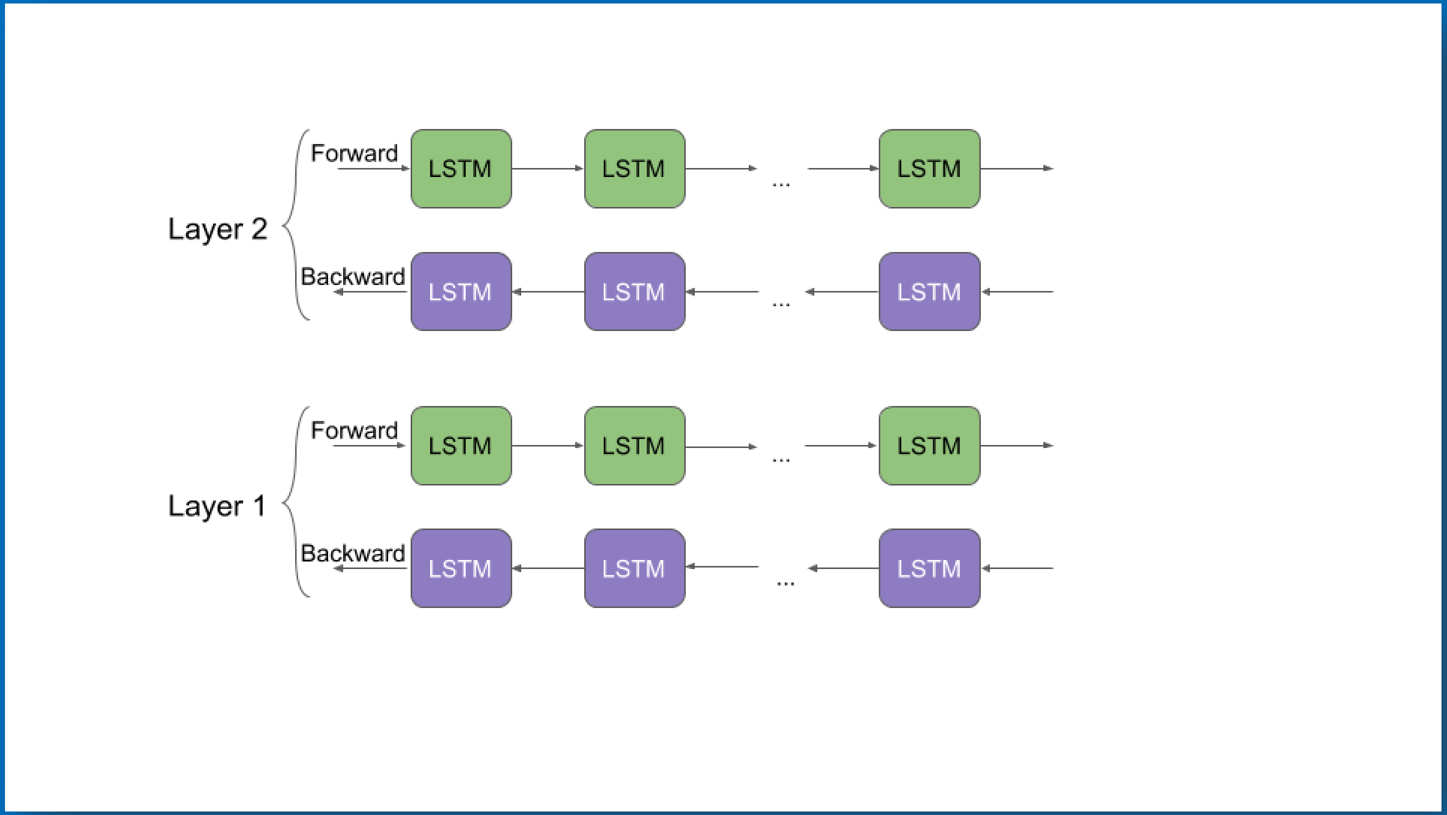
LSTM을 통한 학습

한쪽은 순방향, 그 반대는

역방향 LSTM Layer (n = 2)

이기창님의 한국어 임베딩

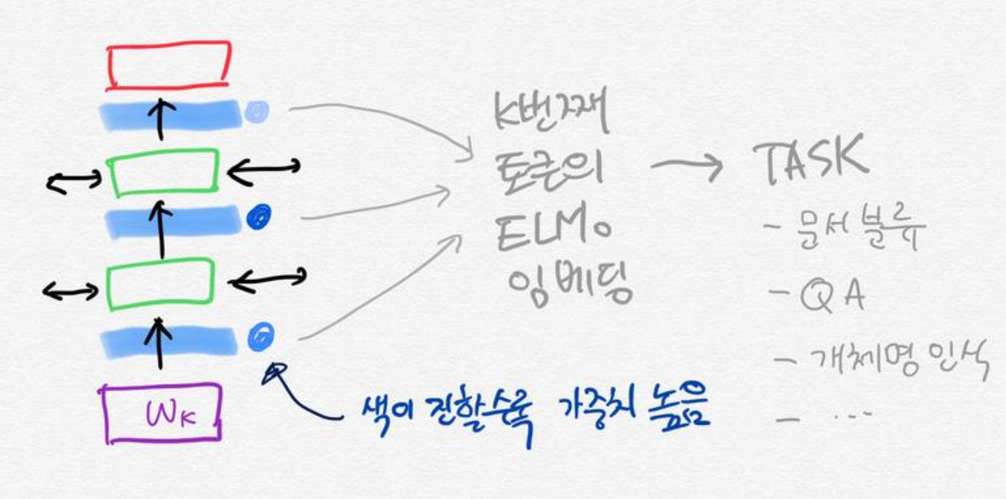
ELMO 레이어



https://www.analyticsvidhya.com/blog/2019/03/learn-to-use-elmo-to-extract-features-from-text/

ELMO 레이어

$$ELMo_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} h_{k,j}^{LM}$$

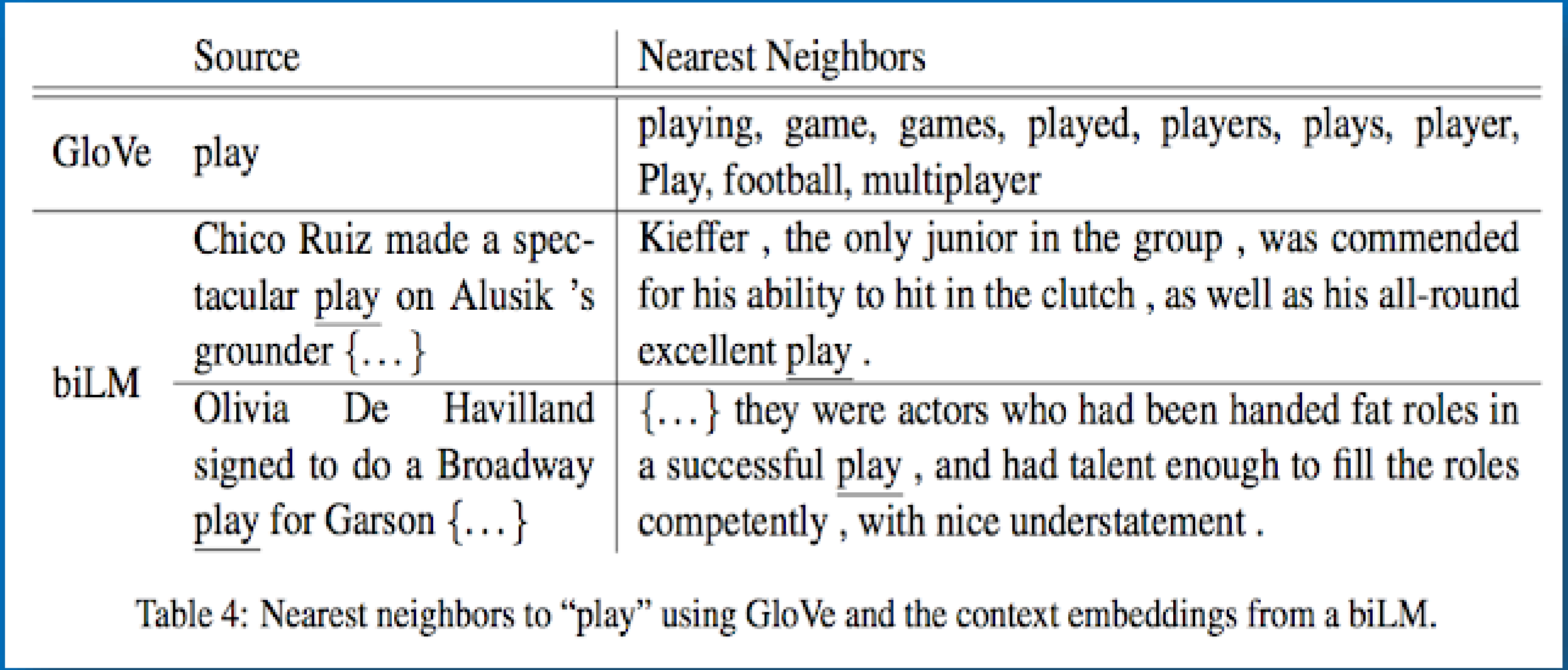


다운스트림 Task를 1개 수행 시 s_j^{task}는 아래 그림에서 동그라미 색상을 가르킴

분류 Task라면, s_j^{task} 학습 손실을 최소하는 방향으로 업데이트 되는 학습 파라메터,

이기창님의 한국어 임베딩

ELMO Result 특징



	Source	Nearest Neighbors
GloVe	play	playing, game, games, played, players, plays, player, Play, football, multiplayer
biLM	Chico Ruiz made a spectacular play on Alusik 's grounder {...}	Kieffer , the only junior in the group , was commended for his ability to hit in the clutch , as well as his all-round excellent play .
	Olivia De Havilland signed to do a Broadway play for Garson {...}	{...} they were actors who had been handed fat roles in a successful play , and had talent enough to fill the roles competently , with nice understatement .

Table 4: Nearest neighbors to "play" using GloVe and the context embeddings from a biLM.

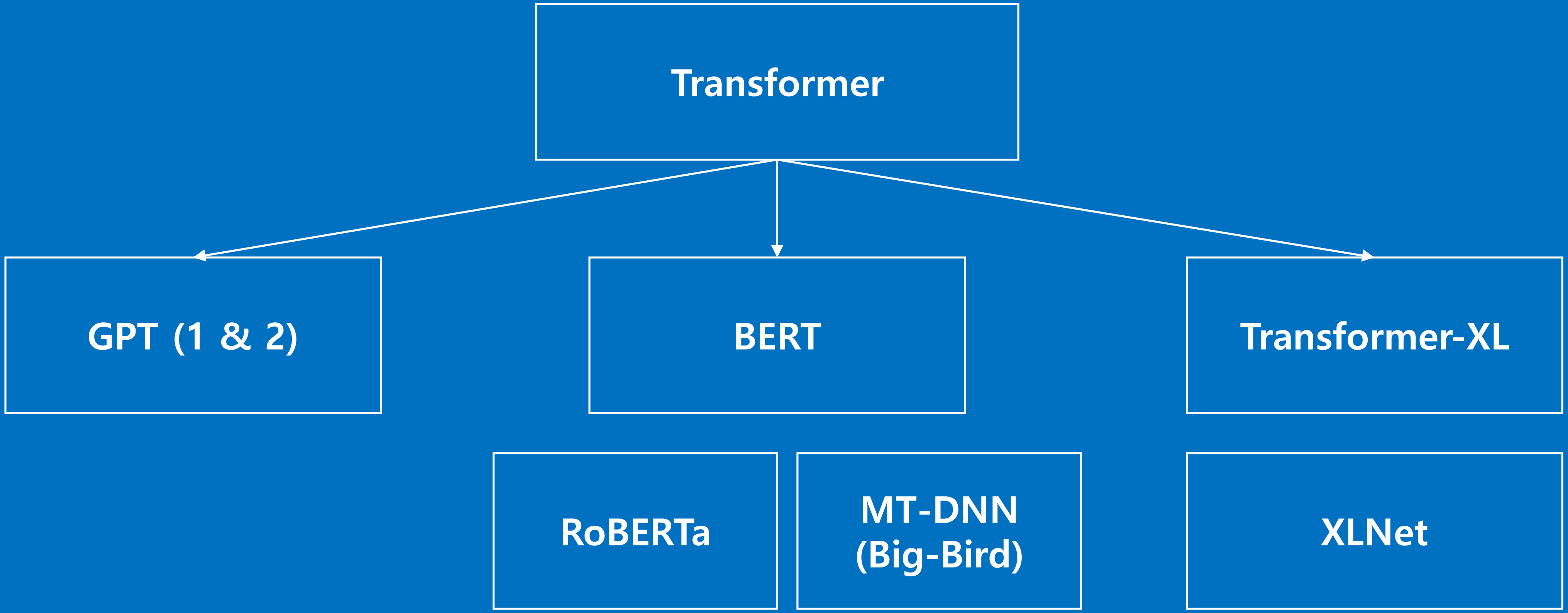
GloVe: 단어 Representation 고정, Play처럼 다양한 뜻을 가진 단어를 표기 할 수 없음

BiLM: 문장 단위로 Play의 백터값이 변한다. 첫번째는 운동경기, 두번째는 연극

https://www.analyticsvidhya.com/blog/2019/03/learn-to-use-elmo-to-extract-features-from-text/

트렌스포머 (Transformer) Google Brain

RNN계열 CNN계열을 이겨내고, 현재 나오는 모든 SOTA 모델의 근간



Transformer

Scaled Dot-Product Attention

Multi-Head Attention

Point-wise Feed-Forward Networks

Attention Is All You Need

Ashish Vaswani[*]
Google Brain
avaswani@google.com

Noam Shazeer[*]
Google Brain
noam@google.com

Niki Parmar[*]
Google Research
nikip@google.com

Jakob Uszkoreit[*]
Google Research
usz@google.com

Llion Jones[*]
Google Research
llion@google.com

Aidan N. Gomez[*] [†]
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser[*]
Google Brain
lukaszkaiser@google.com

Illia Polosukhin[*] [‡]
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

1 Introduction

Recurrent neural networks, long short-term memory [13] and gated recurrent [7] neural networks in particular, have been firmly established as state of the art approaches in sequence modeling and

[*] Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

[†] Work performed while at Google Brain.

[‡] Work performed while at Google Research.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

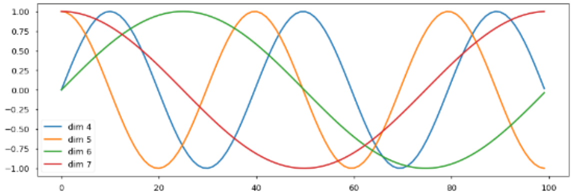
Transformer

residual layer outpu
t

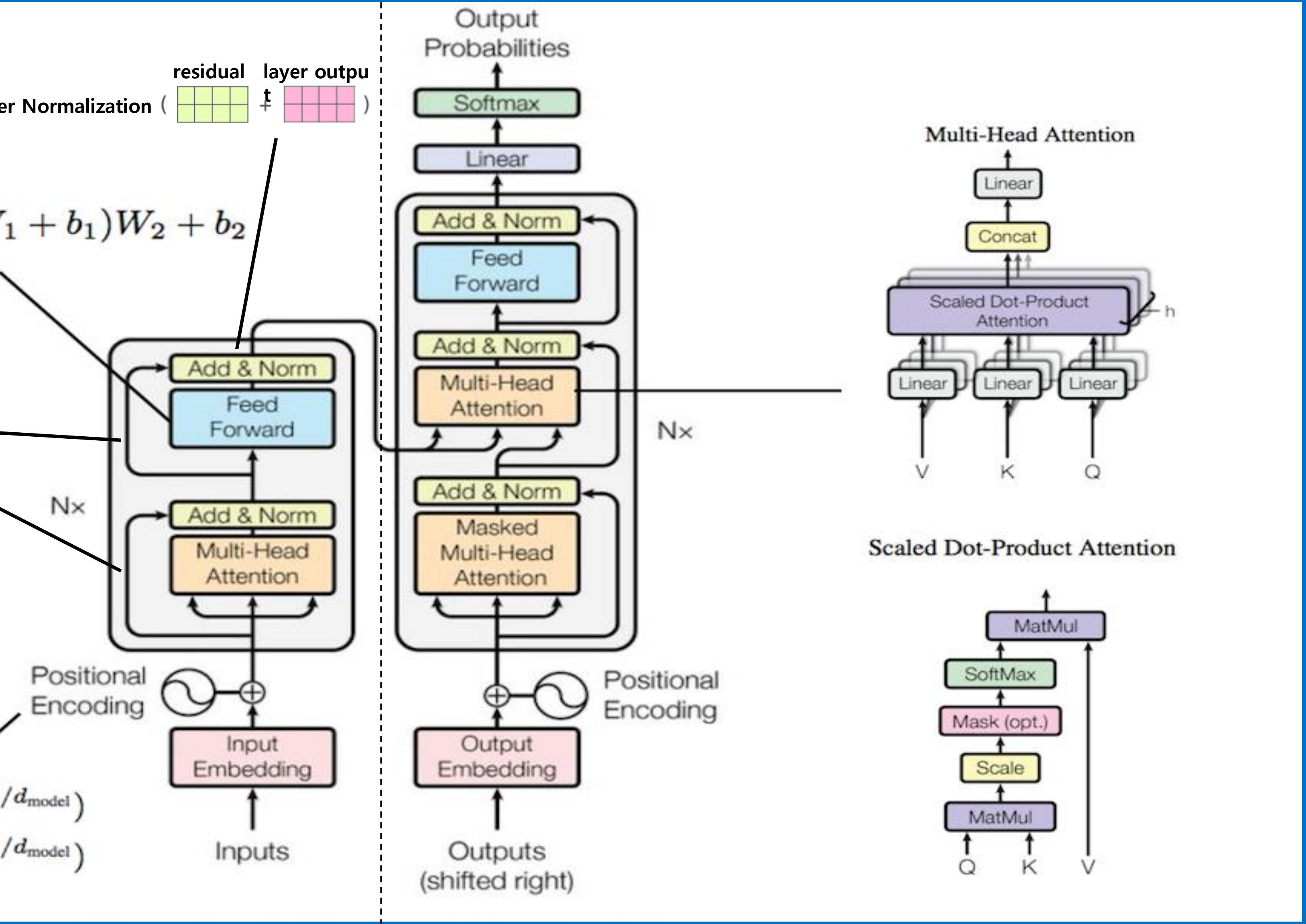
Layer Normalization (+)

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

Residual connections



$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

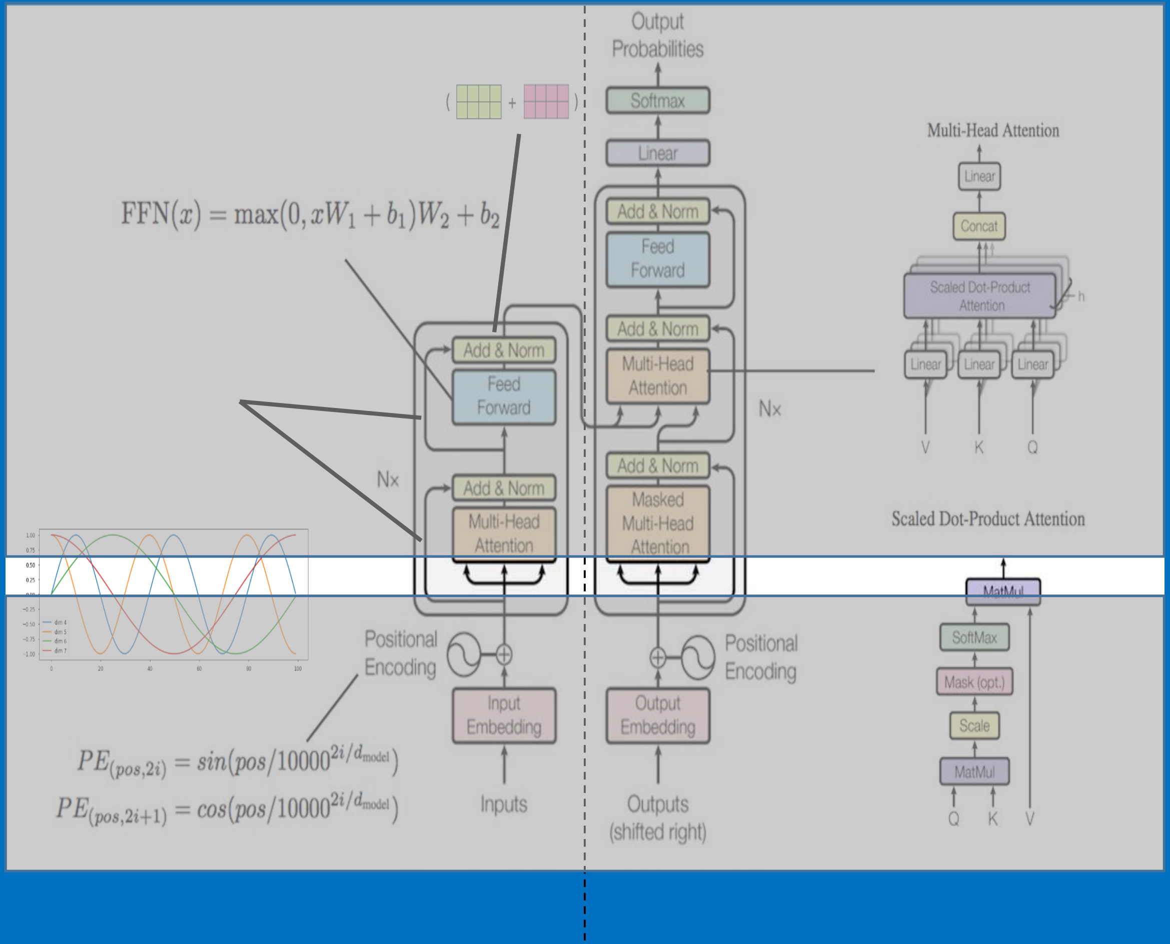


Encoder Decoder

Vaswani et al. 2017

Transformer

Scaled Dot-Product Attention



Created by fahmionline
from Noun Project

Query

Created by Smalllike
from Noun Project

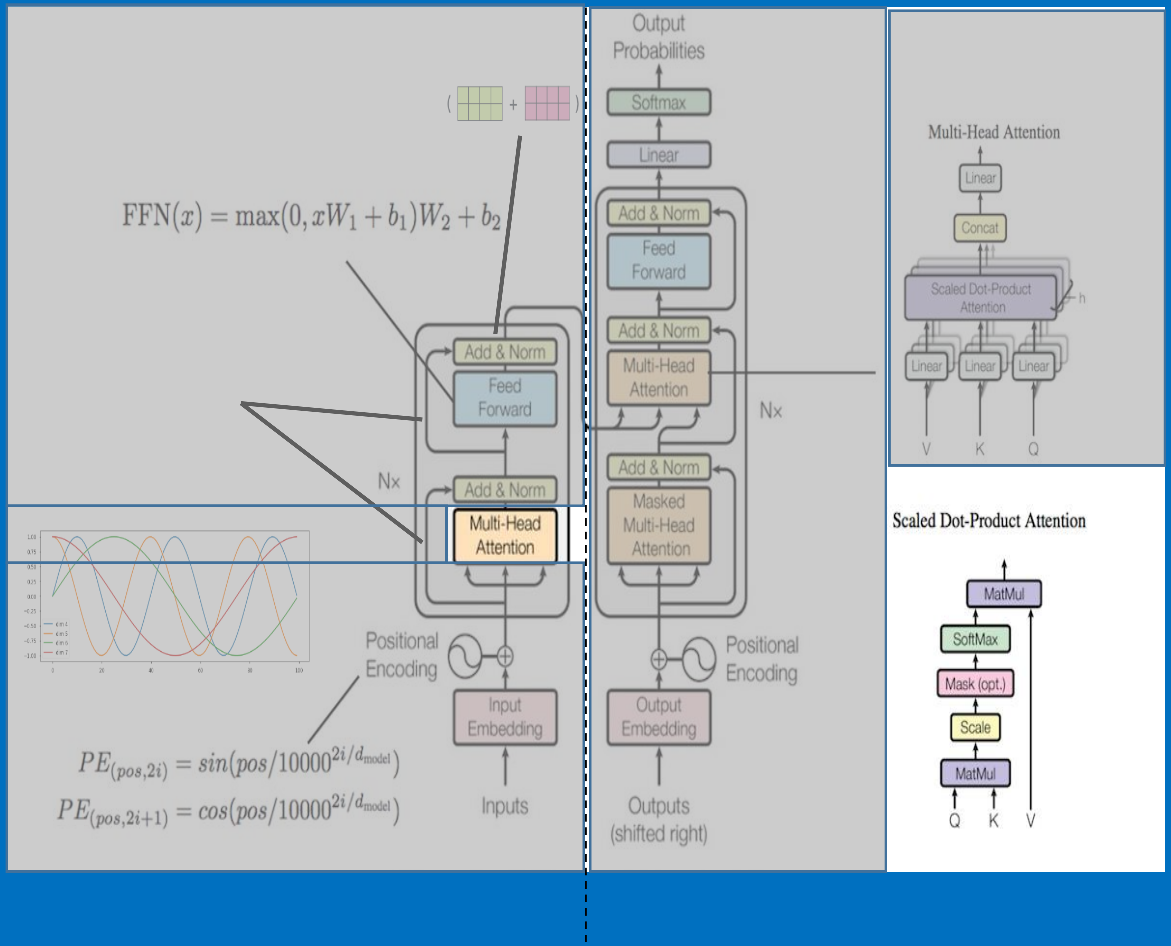
Key

Created by Creative Stall
from Noun Project

Value

Transformer

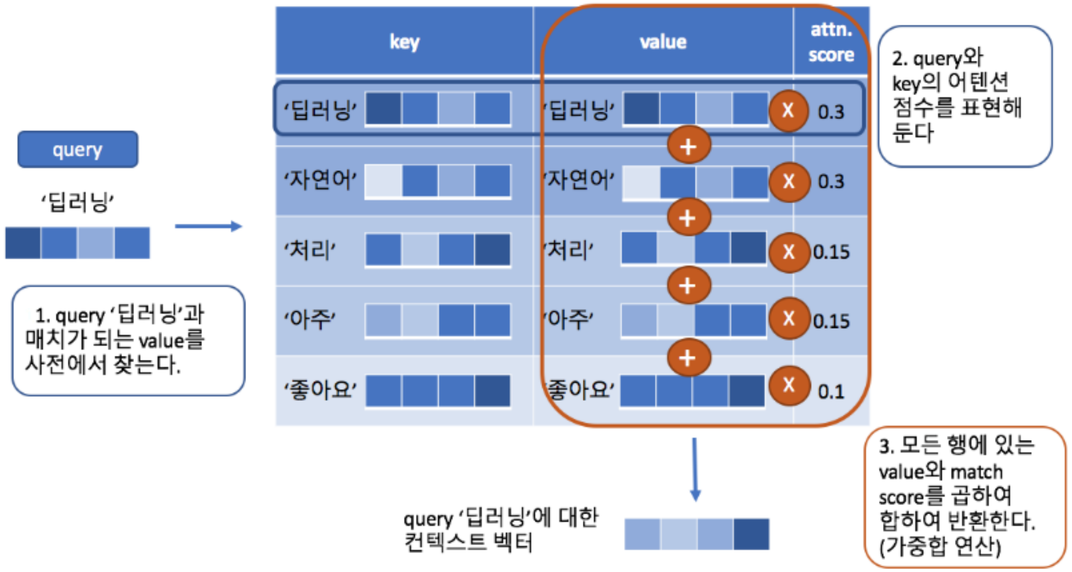
Scaled Dot-Product Attention



$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$



1. query '딥러닝'과 매치가 되는 value를 사전에서 찾는다.

2. query와 key의 어텐션 점수를 표현해 둔다

3. 모든 행에 있는 value와 match score를 곱하여 합하여 반환한다. (가중합 연산)

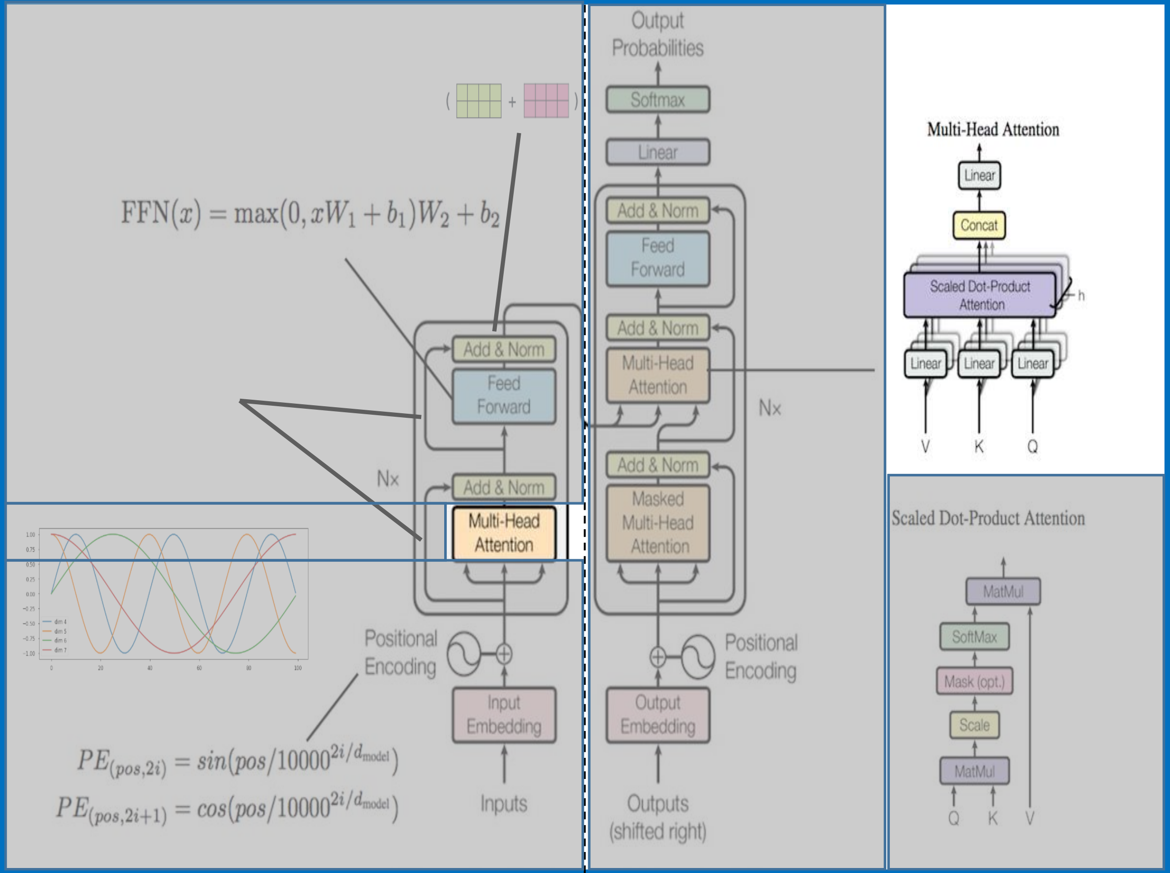
query '딥러닝'에 대한 컨텍스트 벡터

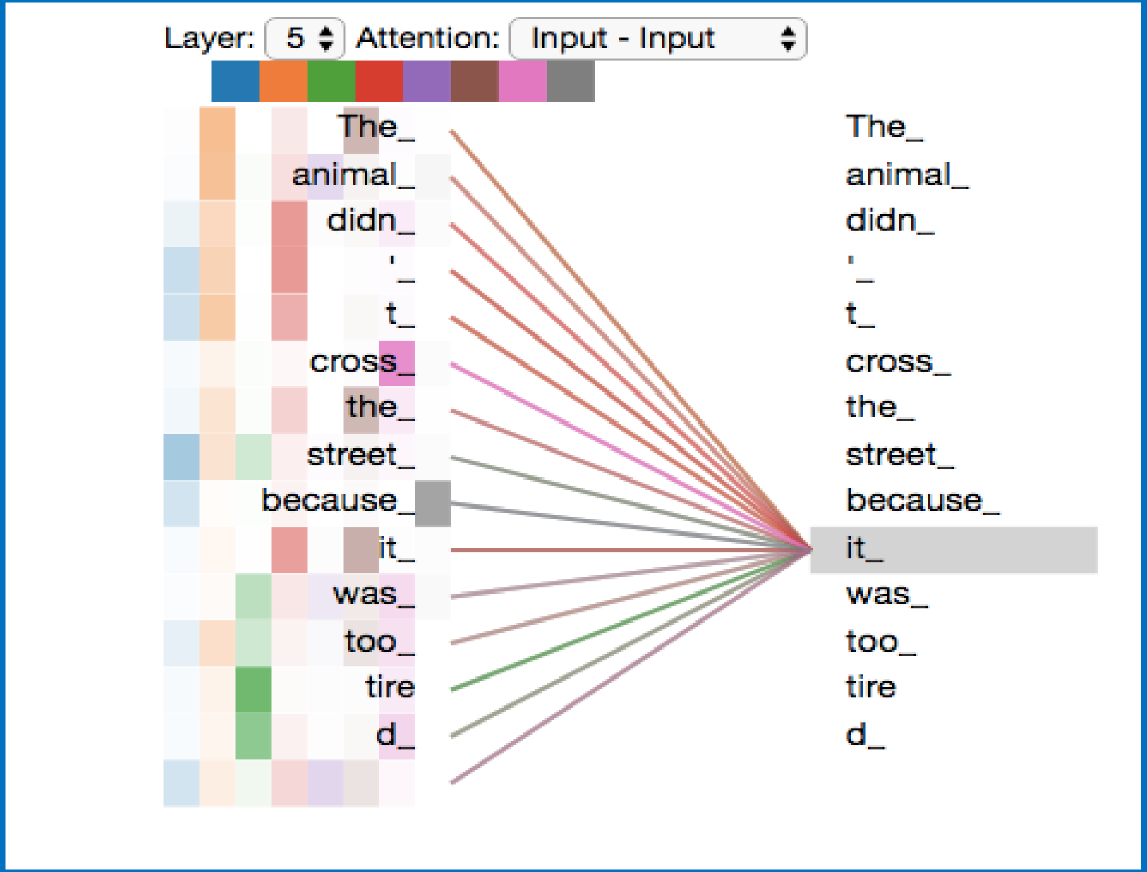
그림. key, query, value로 표현한 셀프 어텐션

$$\text{Attention}(Q, K, V) = \text{softmax}\ \frac{QK^T}{\sqrt{d_k}}\ V$$

Transformer

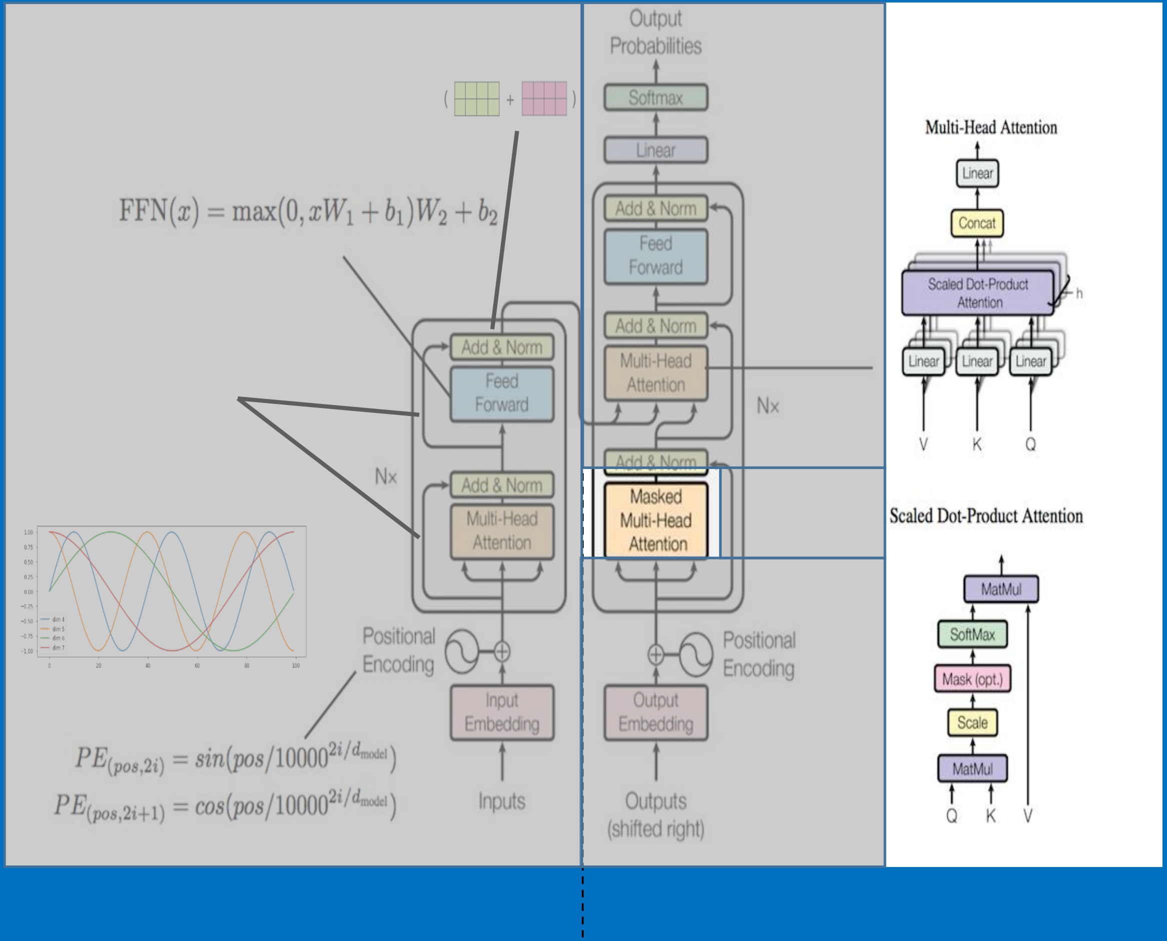
Multi-Head Attention

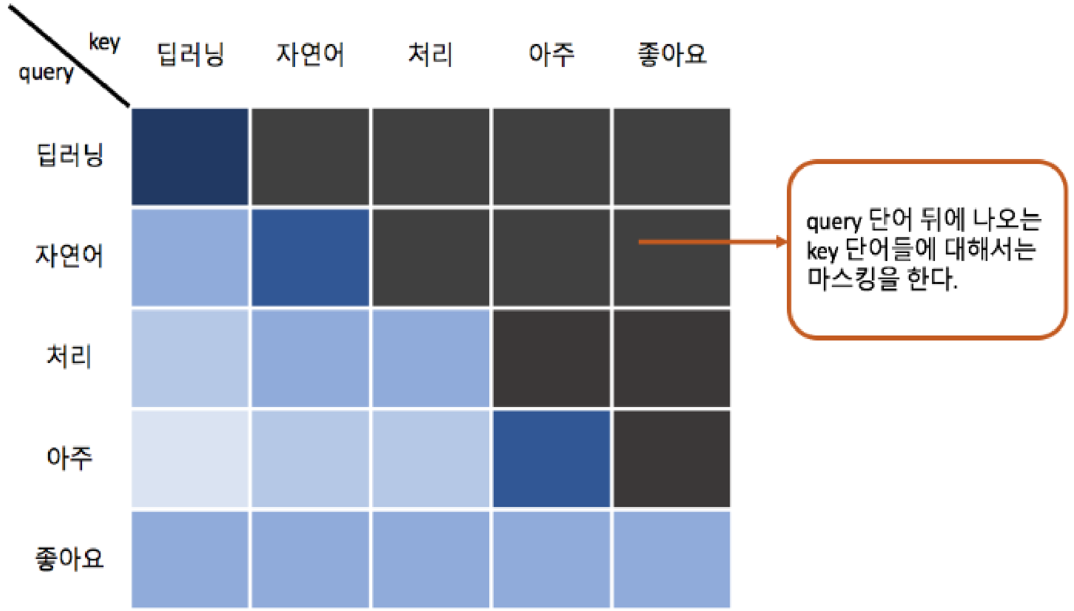




Layer: 5 Attention: Input - Input

Masked Multi-Head Attention

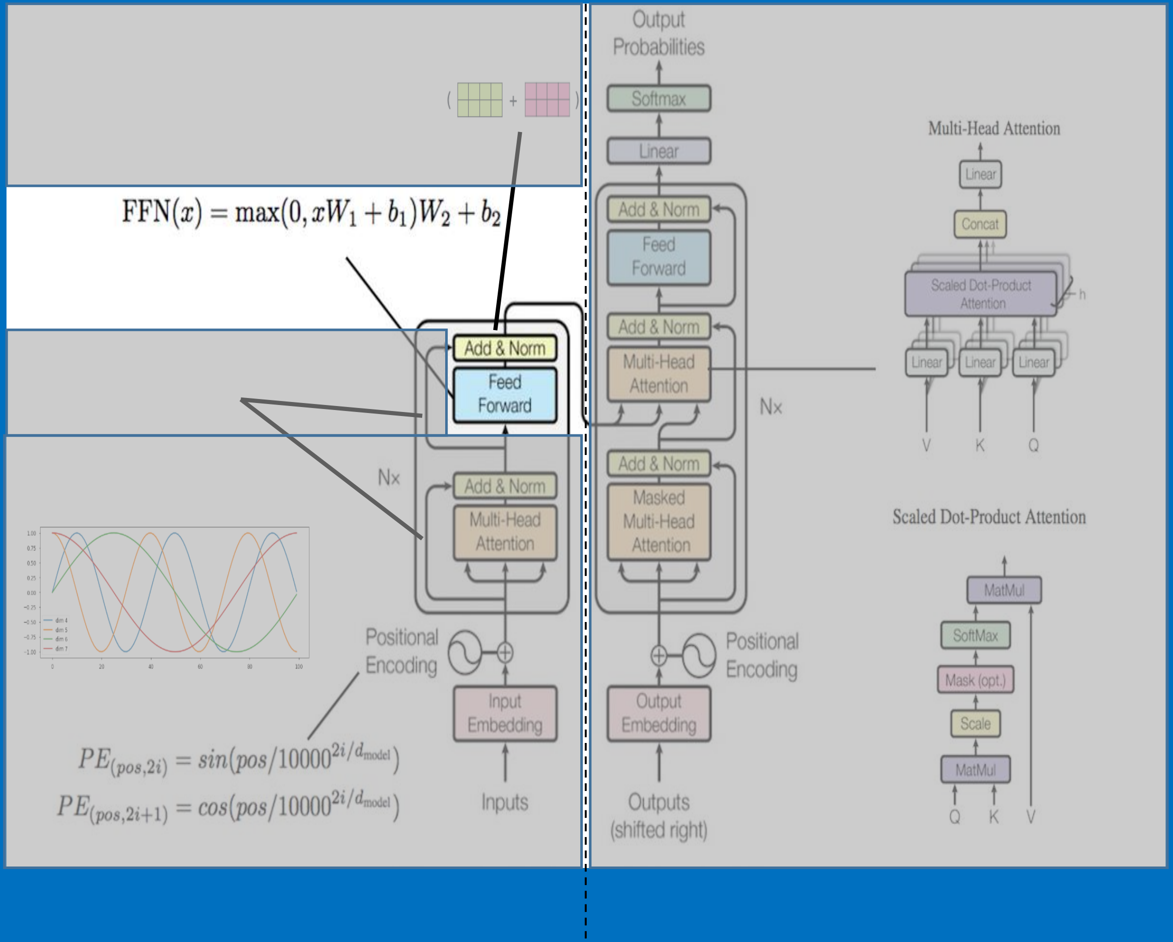




query 단어 뒤에 나오는 key 단어들에 대해서는 마스킹을 한다.

그림. 순방향 마스크가 된 어텐션 맵

Transformer



$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$
Output Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Nx
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)
Multi-Head Attention
Linear
Concat
Scaled Dot-Product Attention
h
V
K
Q
Scaled Dot-Product Attention
MatMul
SoftMax
Mask (opt.)
Scale
$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$
$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$

Point-wise Feed-Forward Networks

Self-Attention을 통해 나온,

정보를 피드 포워드 네트워크 통과

두개의 Linear transformation 구성

ReLU Fuction 사용

Self-Attention과 FFN을 통과 할 때 마다

Shortcut Connection

+

Layer Normalization

GPT

GPT (Generative Pre-training)

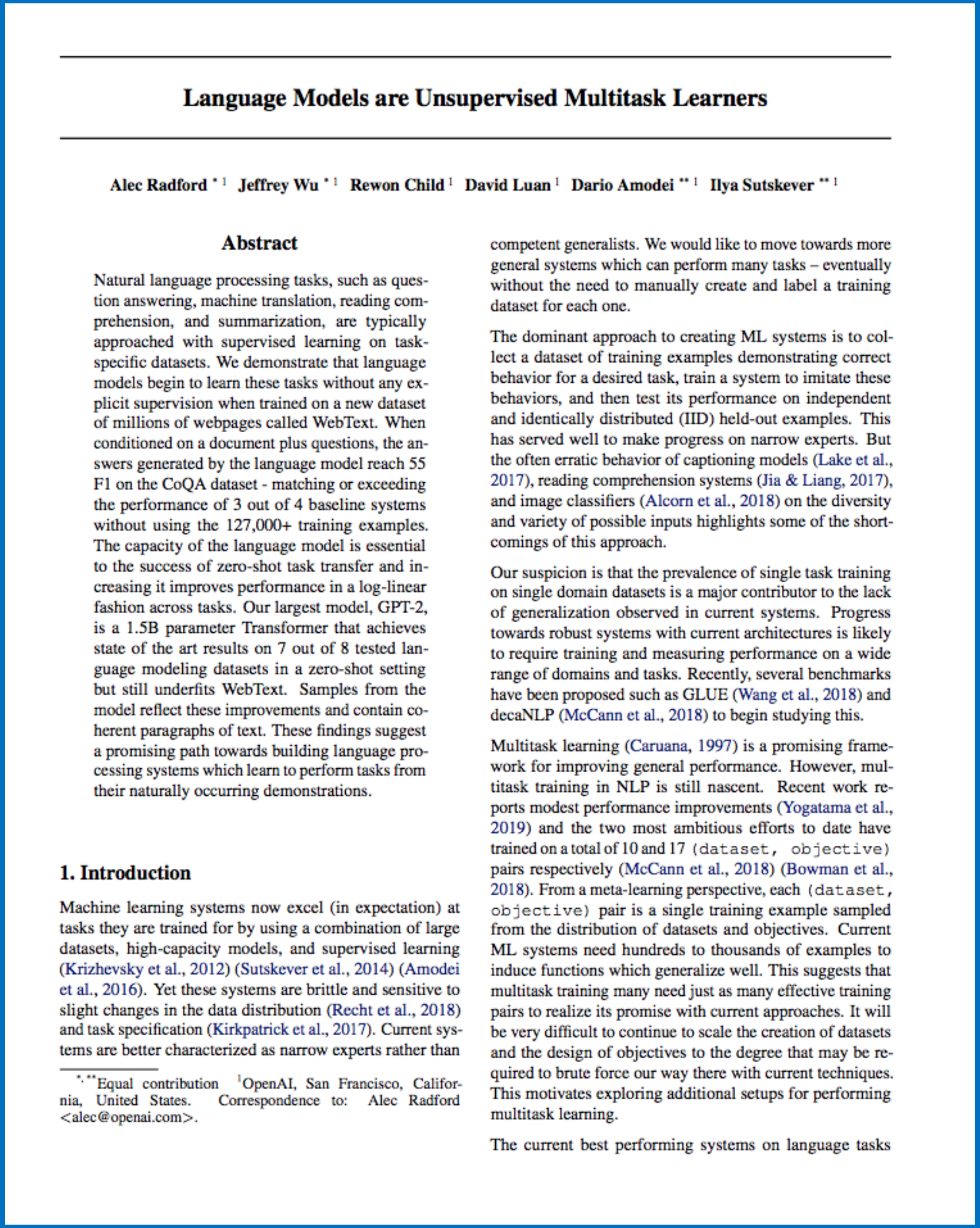
OpenAI에서 발표 (2018)

Transformer Decoder를 기반으로 한 언어모델

Generation에 좋은 성능을 보임

Zero-shot Learning을 시도

GPT1과 GPT2는 데이터 크기 및 파라메터 수 차이



Language Models are Unsupervised Multitask Learners

Alec Radford [*1] Jeffrey Wu [*1] Rewon Child [1] David Luan [1] Dario Amodei [**1] Ilya Sutskever [**1]

Abstract

Natural language processing tasks, such as question answering, machine translation, reading comprehension, and summarization, are typically approached with supervised learning on task-specific datasets. We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText. When conditioned on a document plus questions, the answers generated by the language model reach 55 F1 on the CoQA dataset - matching or exceeding the performance of 3 out of 4 baseline systems without using the 127,000+ training examples. The capacity of the language model is essential to the success of zero-shot task transfer and increasing it improves performance in a log-linear fashion across tasks. Our largest model, GPT-2, is a 1.5B parameter Transformer that achieves state of the art results on 7 out of 8 tested language modeling datasets in a zero-shot setting but still underfits WebText. Samples from the model reflect these improvements and contain coherent paragraphs of text. These findings suggest a promising path towards building language processing systems which learn to perform tasks from their naturally occurring demonstrations.

1. Introduction

Machine learning systems now excel (in expectation) at tasks they are trained for by using a combination of large datasets, high-capacity models, and supervised learning (Krizhevsky et al., 2012) (Sutskever et al., 2014) (Amodei et al., 2016). Yet these systems are brittle and sensitive to slight changes in the data distribution (Recht et al., 2018) and task specification (Kirkpatrick et al., 2017). Current systems are better characterized as narrow experts rather than competent generalists. We would like to move towards more general systems which can perform many tasks – eventually without the need to manually create and label a training dataset for each one.

The dominant approach to creating ML systems is to collect a dataset of training examples demonstrating correct behavior for a desired task, train a system to imitate these behaviors, and then test its performance on independent and identically distributed (IID) held-out examples. This has served well to make progress on narrow experts. But the often erratic behavior of captioning models (Lake et al., 2017), reading comprehension systems (Jia & Liang, 2017), and image classifiers (Alcorn et al., 2018) on the diversity and variety of possible inputs highlights some of the shortcomings of this approach.

Our suspicion is that the prevalence of single task training on single domain datasets is a major contributor to the lack of generalization observed in current systems. Progress towards robust systems with current architectures is likely to require training and measuring performance on a wide range of domains and tasks. Recently, several benchmarks have been proposed such as GLUE (Wang et al., 2018) and decaNLP (McCann et al., 2018) to begin studying this.

Multitask learning (Caruana, 1997) is a promising framework for improving general performance. However, multitask training in NLP is still nascent. Recent work reports modest performance improvements (Yogatama et al., 2019) and the two most ambitious efforts to date have trained on a total of 10 and 17 `(dataset, objective)` pairs respectively (McCann et al., 2018) (Bowman et al., 2018). From a meta-learning perspective, each `(dataset, objective)` pair is a single training example sampled from the distribution of datasets and objectives. Current ML systems need hundreds to thousands of examples to induce functions which generalize well. This suggests that multitask training many need just as many effective training pairs to realize its promise with current approaches. It will be very difficult to continue to scale the creation of datasets and the design of objectives to the degree that may be required to brute force our way there with current techniques. This motivates exploring additional setups for performing multitask learning.

The current best performing systems on language tasks

[*, **]Equal contribution [1]OpenAI, San Francisco, California, United States. Correspondence to: Alec Radford <alec@openai.com>.

GPT의 꿈

[테크M 기획] 가짜도 진짜처럼 … 글 잘 쓰는 AI GPT2 등장 – 테크 M

오픈 AI, GPT2 알고리즘 공개하지 않기로 해… 가짜뉴스 생산 우려

개발자 조차 긴장시킨 '글 너무 잘 쓰는' AI

40GB 분량 텍스트 학습… 어떤 이야기도 척척 이어줘

https://www.zdnet.co.kr/view/?no=20190219111213

http://techm.kr/bbs/board.php?bo_table=article&wr_id=5727

GPT의 꿈

"4월 어느 화창하고 쌀쌀한 날이었다. 벽시계가 13시를 가리켰다"

from. '1984' 조지 오웰

"나는 차를 타고 시애틀에 있는 새 일자리로 가는 길이었다.
가스를 넣고, 키를 꽂은 뒤 차를 달렸다.
오늘 어떤 날이 펼쳐질 지 상상했다. 지금으로부터 100년 뒤인 2045년.
나는 중국 시골의 한 가난한 지역의 교사다. 중국사와 과학 역사로 시작했다."

https://www.zdnet.co.kr/view/?no=20190219111213

http://techm.kr/bbs/board.php?bo_table=article&wr_id=5727

GPT의 꿈

GPT1

세상에 많은 Unlabeled 텍스트

이를 잘 활용하여,

모든 자연어 Task를

잘 하고 싶다.

Book Corpus 활용

(소설책 7천개 분량)

GPT2

웹페이지 크롤링을 통해, 4500만개

링크에서 Text 40GB를 활용 (소설 3만5천개)

모델 Parameter는 15억개

(Bert-Large가 3.4억개)

특정 도메인에 특화하지 않는

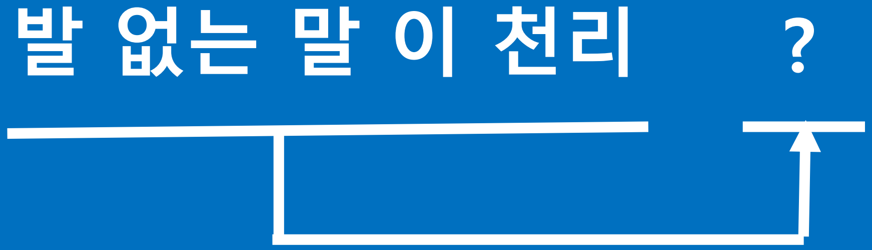
Zero-shot Learning을 테스트

GPT1 – Pre-training

언어 모델 (Language Model) + Transformer Decoder

$$L_1(\mathcal{U}) = \sum_i \log P(u_i|u_{i-k}, \ldots, u_{i-1}; \Theta)$$

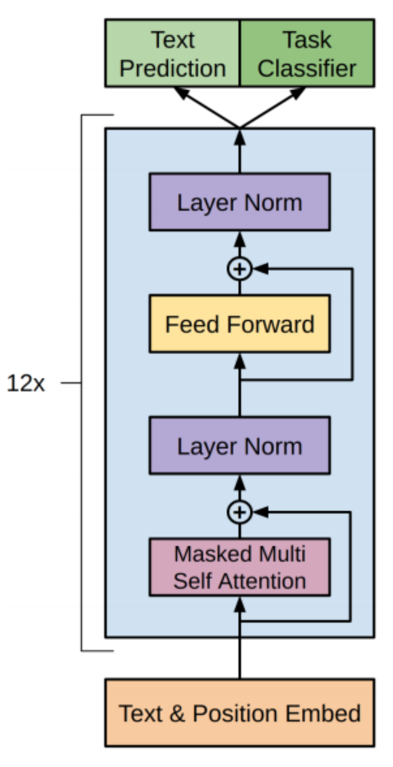
$$U = \{u_1, \ldots, u_2\}$$

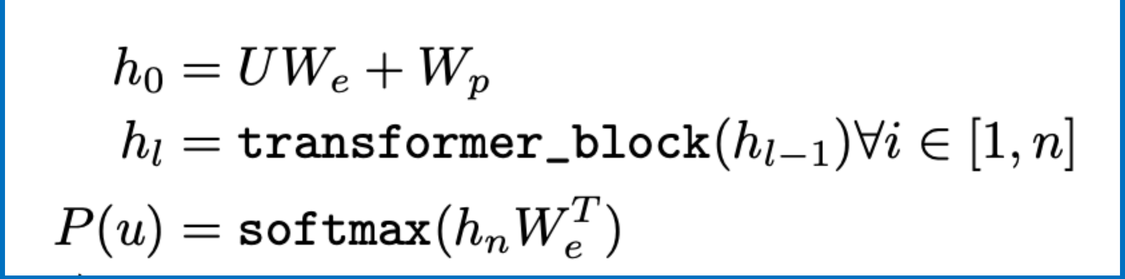


발 없는 말 이 천리 ?

GPT1 – Pre-training

언어 모델 (Language Model) + Transformer Decoder



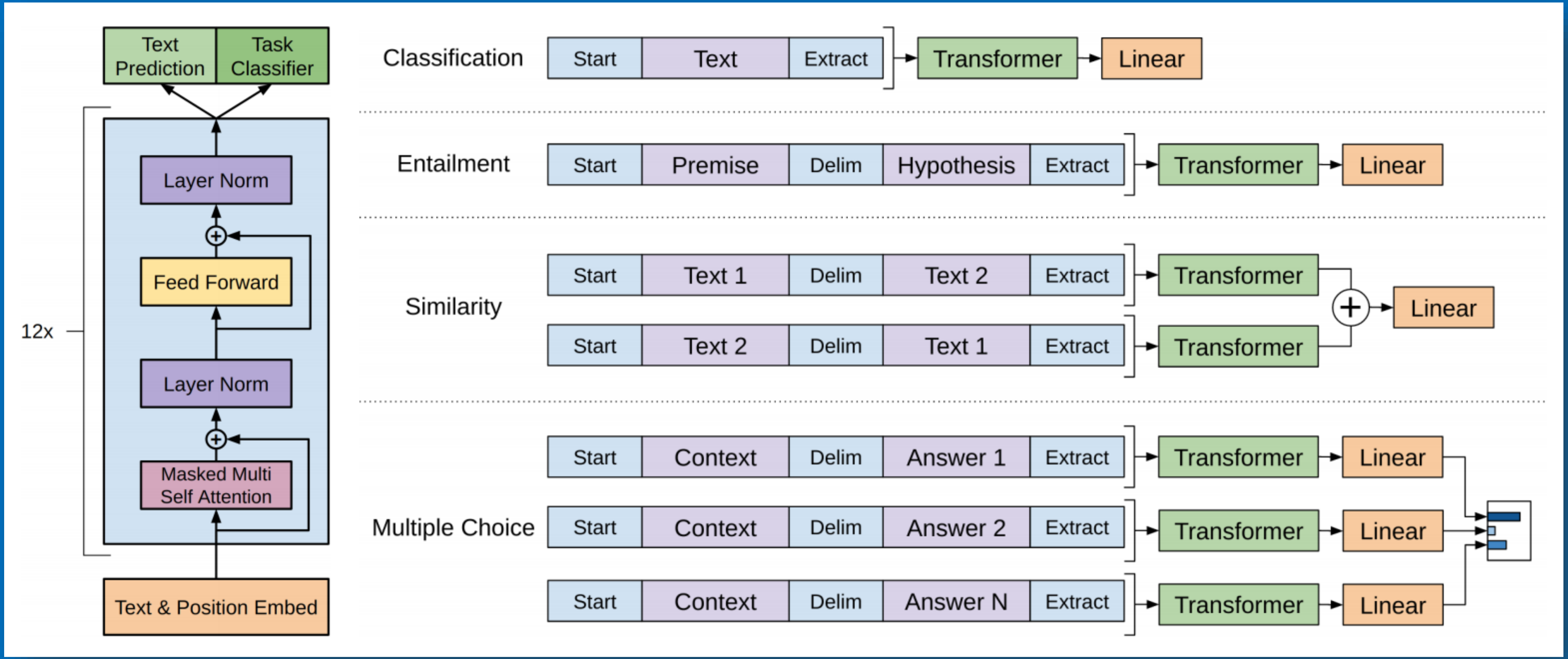


$$
\begin{aligned}
h_0 &= UW_e + W_p \\
h_l &= \texttt{transformer\_block}(h_{l-1}) \forall i \in [1, n] \\
P(u) &= \texttt{softmax}(h_n W_e^T)
\end{aligned}
$$

Transformer Decoder 부분을 활용하여,

Language Model을 학습

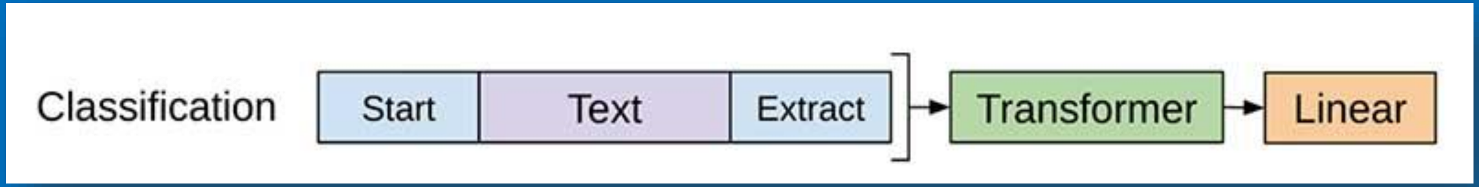
GPT1 – Fine-tuning

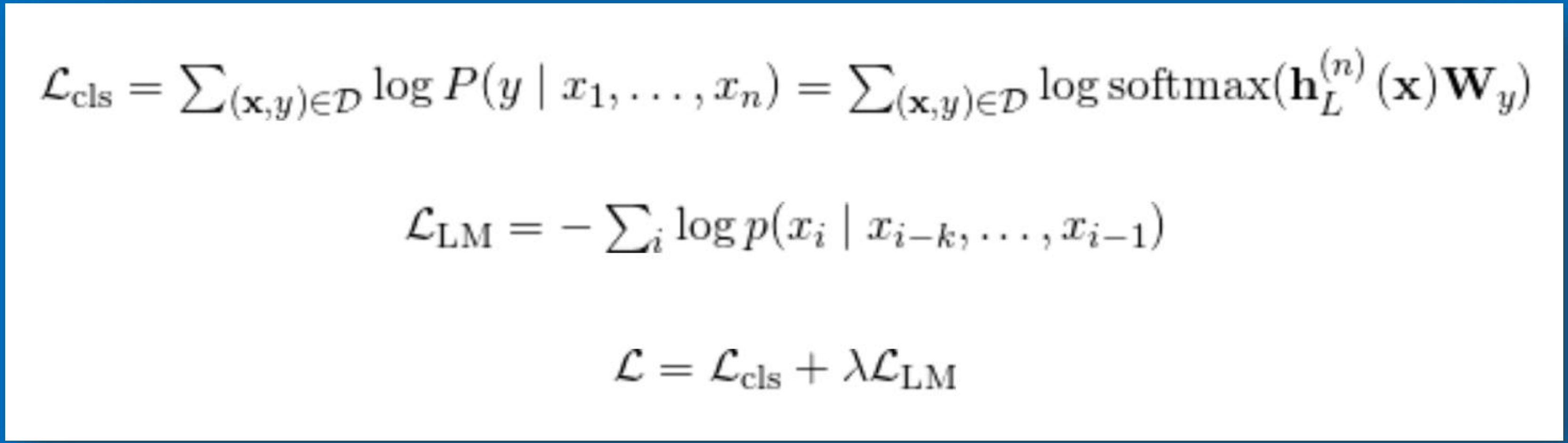


문장 분류, 추론, 유사도, QA 등 다양한 Task를 실험

GPT1 – Fine-tuning

Auxiliary Loss를 활용





$$\mathcal{L}_{\text{cls}} = \sum_{(\mathbf{x},y)\in\mathcal{D}} \log P(y \mid x_1,\ldots,x_n) = \sum_{(\mathbf{x},y)\in\mathcal{D}} \log \operatorname{softmax}(\mathbf{h}_L^{(n)}(\mathbf{x})\mathbf{W}_y)$$

$$\mathcal{L}_{\text{LM}} = -\sum_i \log p(x_i \mid x_{i-k},\ldots,x_{i-1})$$

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda\mathcal{L}_{\text{LM}}$$

(1)수렴하는 속도가 빨라지고, (2)Supervised 모델의 일반화를 향상

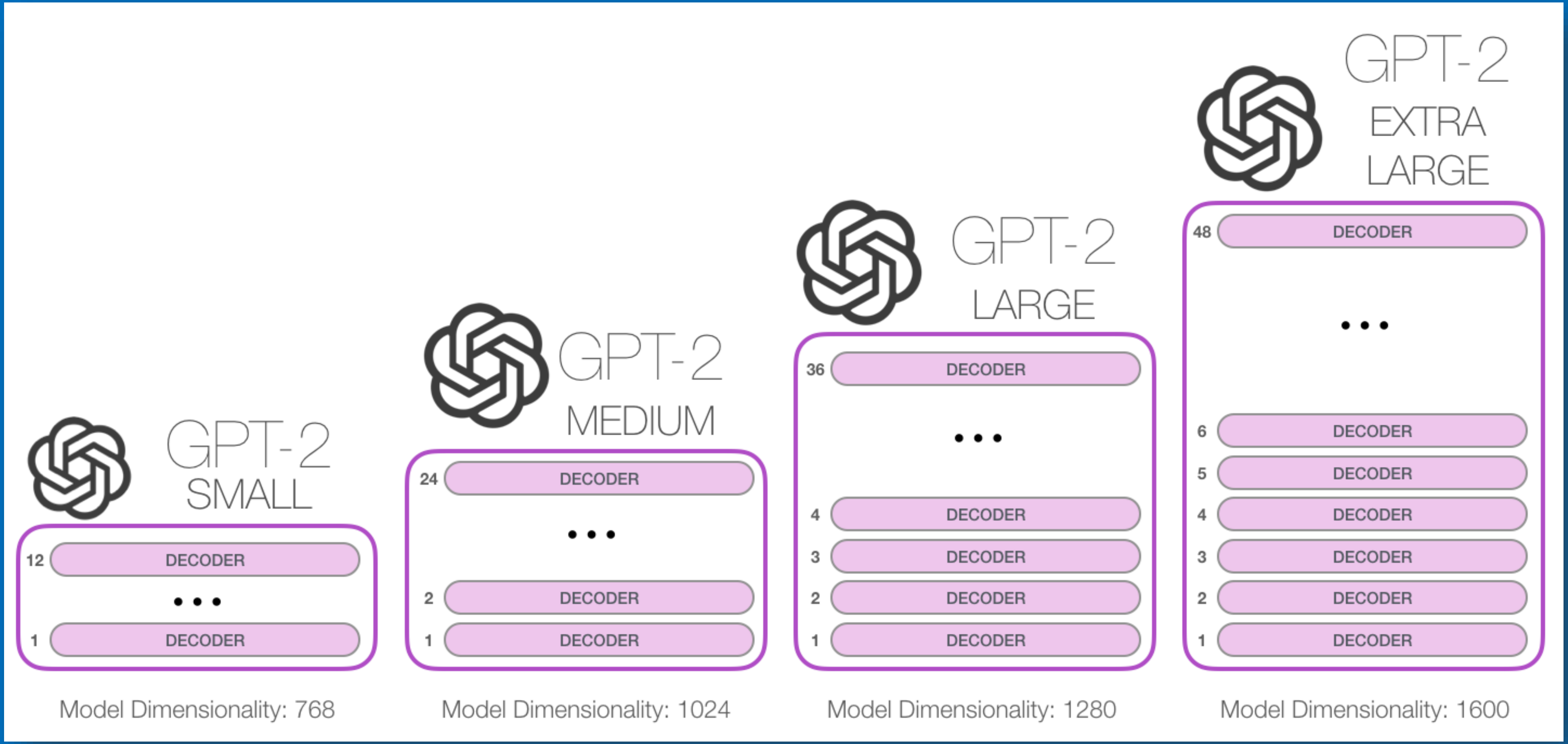
GPT2 – Pre-Training

온라인에서 많은 데이터를 수집하여, 학습

자체 제작한 WebText를 활용 (품질 향상)

- Reddit에서 카르마 '3'이상을 받은 포스트 (나름 신뢰)
- Reddit links 4500만개 링크 + Dragnet과 Newspaper 조합
- Wikipedia 제외 (다수 링크 중복)

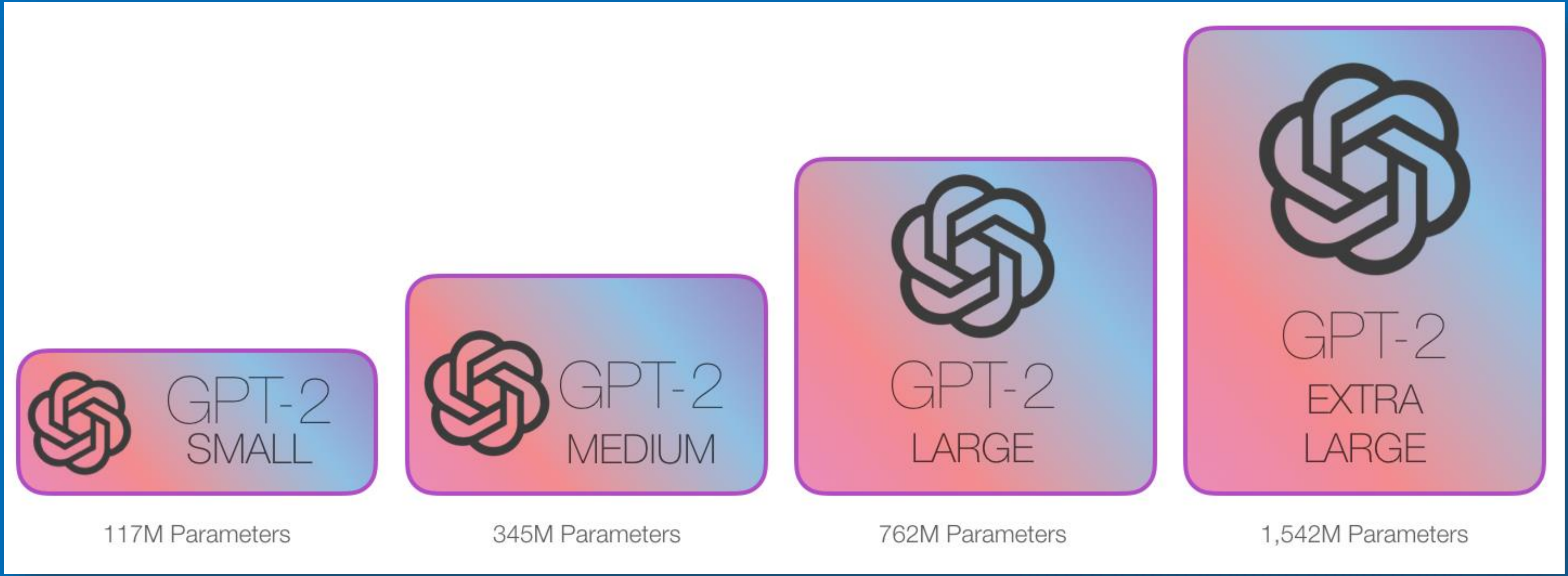
GPT2 – Model



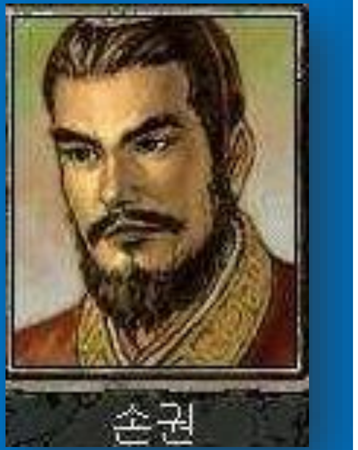
http://jalammar.github.io/illustrated-gpt2/

GPT2 – Model

BERT based 13.6배, BERT Large 4.5배!



GPT2 – Zero-Shot Learning



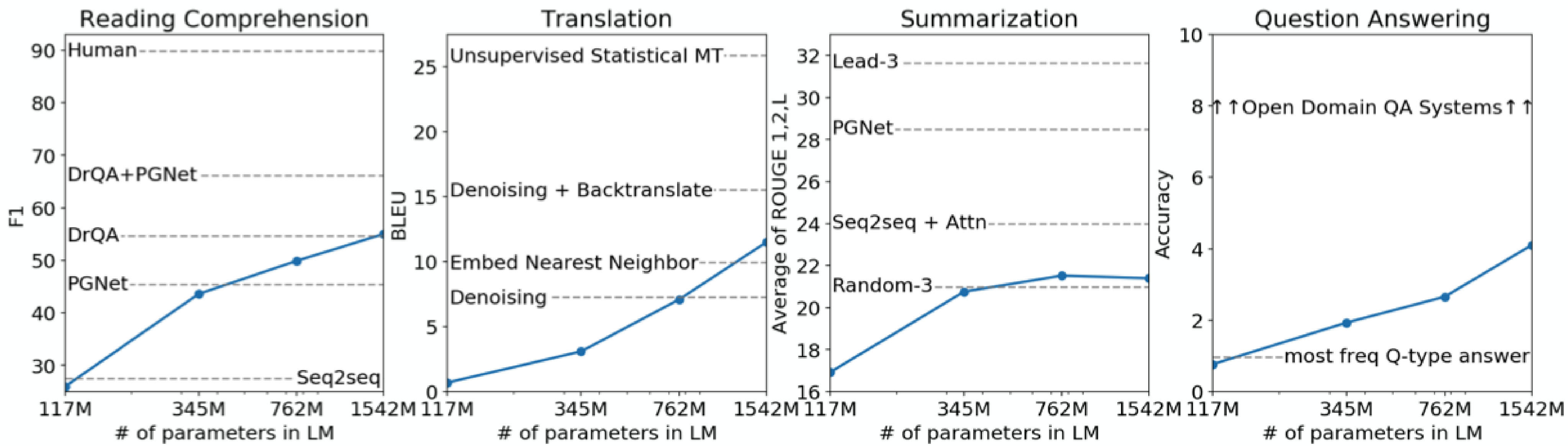
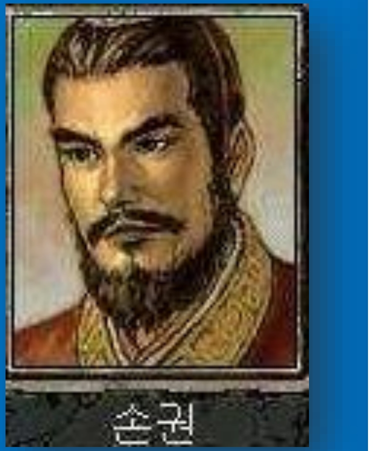
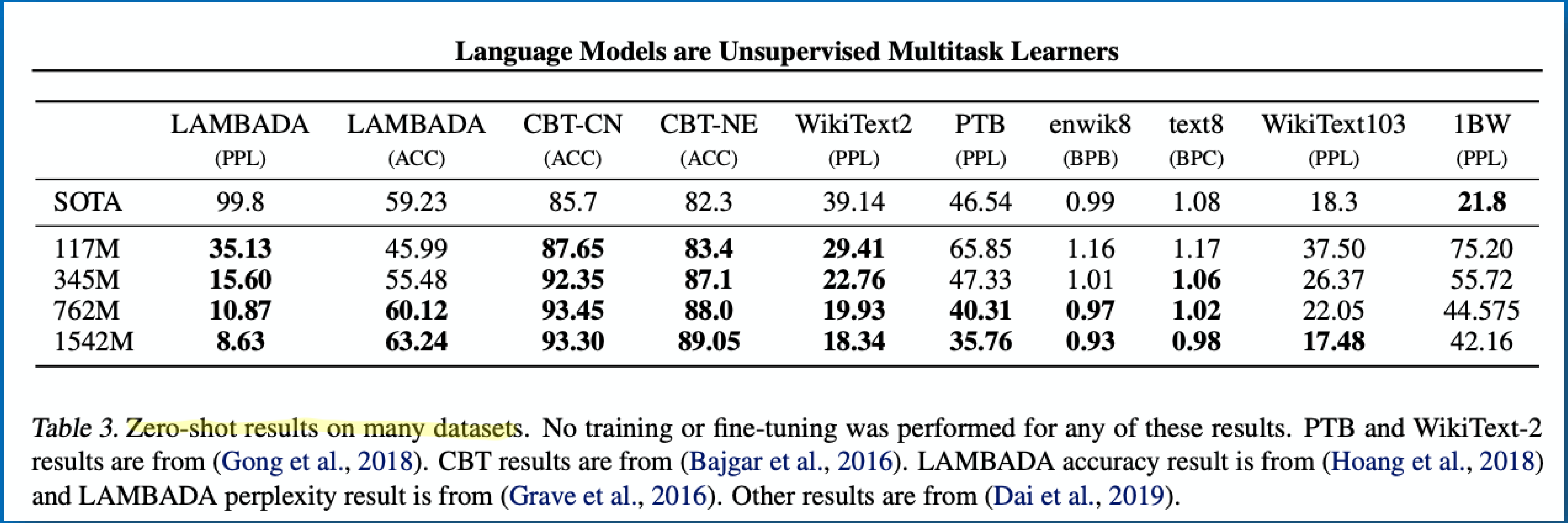


Figure 1. Zero-shot task performance of WebText LMs as a function of model size on many NLP tasks. Reading Comprehension results are on CoQA (Reddy et al., 2018), translation on WMT-14 Fr-En (Artetxe et al., 2017), summarization on CNN and Daily Mail (See et al., 2017), and Question Answering on Natural Questions (Kwiatkowski et al., 2019). Section 3 contains detailed descriptions of each result.

*Zero-Shot Learning: 특정 Task에 맞게 Finetuning 시키지 않고, 기능을 측정

GPT2 – Zero-Shot Learning





Language Models are Unsupervised Multitask Learners

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). CBT results are from (Bajgar et al., 2016). LAMBADA accuracy result is from (Hoang et al., 2018) and LAMBADA perplexity result is from (Grave et al., 2016). Other results are from (Dai et al., 2019).

BERT

Google Brain

BERT (Bidirectional Encoder Representations from Transformer)

Transformer Encoder를 기반으로 한 양방향 언어모델

최종 Layer 1층만 변경하고 SOTA를 달성

자연어처리를 한 단계 격상, "A New Era of NLP"

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**
Google AI Language
{jacobdevlin,mingweichang,kentonl,kristout}@google.com

Abstract

We introduce a new language representation model called **BERT**, which stands for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers. Unlike recent language representation models (Peters et al., 2018a; Radford et al., 2018), BERT is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, including pushing the GLUE score to 80.5% (7.7% point absolute improvement), MultiNLI accuracy to 86.7% (4.6% absolute improvement), SQuAD v1.1 question answering Test F1 to 93.2 (1.5 point absolute improvement) and SQuAD v2.0 Test F1 to 83.1 (5.1 point absolute improvement).

1 Introduction

Language model pre-training has been shown to be effective for improving many natural language processing tasks (Dai and Le, 2015; Peters et al., 2018a; Radford et al., 2018; Howard and Ruder, 2018). These include sentence-level tasks such as natural language inference (Bowman et al., 2015; Williams et al., 2018) and paraphrasing (Dolan and Brockett, 2005), which aim to predict the relationships between sentences by analyzing them holistically, as well as token-level tasks such as named entity recognition and question answering, where models are required to produce fine-grained output at the token level (Tjong Kim Sang and De Meulder, 2003; Rajpurkar et al., 2016).

There are two existing strategies for applying pre-trained language representations to downstream tasks: *feature-based* and *fine-tuning*. The feature-based approach, such as ELMo (Peters et al., 2018a), uses task-specific architectures that include the pre-trained representations as additional features. The fine-tuning approach, such as the Generative Pre-trained Transformer (OpenAI GPT) (Radford et al., 2018), introduces minimal task-specific parameters, and is trained on the downstream tasks by simply fine-tuning *all* pre-trained parameters. The two approaches share the same objective function during pre-training, where they use unidirectional language models to learn general language representations.

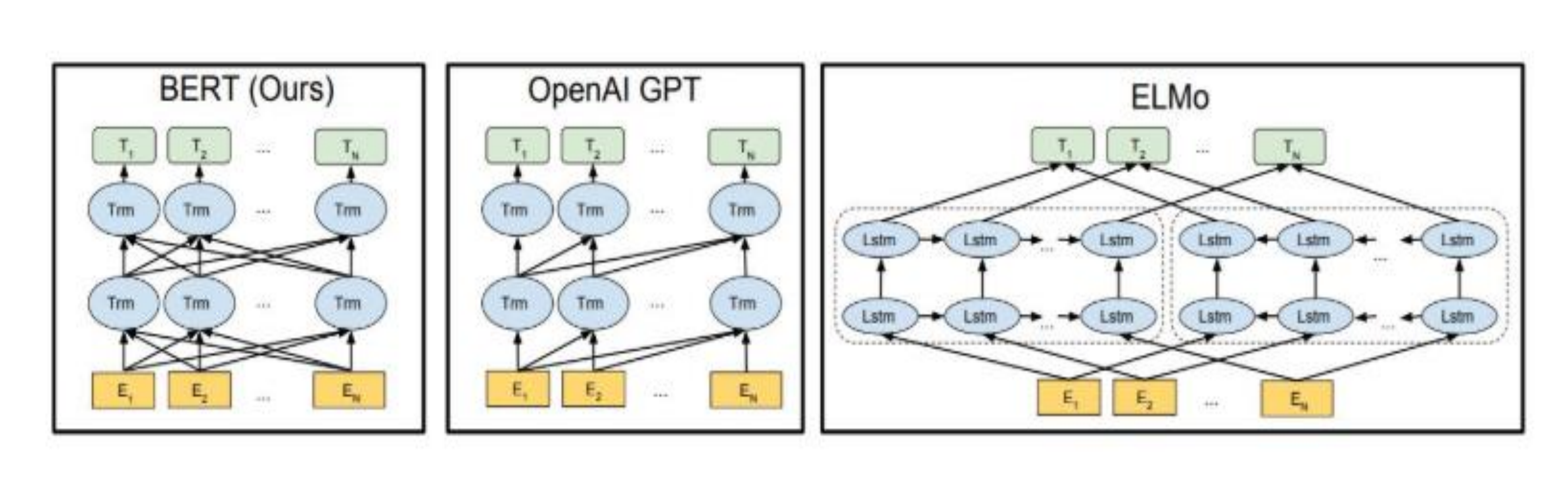
We argue that current techniques restrict the power of the pre-trained representations, especially for the fine-tuning approaches. The major limitation is that standard language models are unidirectional, and this limits the choice of architectures that can be used during pre-training. For example, in OpenAI GPT, the authors use a left-to-right architecture, where every token can only attend to previous tokens in the self-attention layers of the Transformer (Vaswani et al., 2017). Such restrictions are sub-optimal for sentence-level tasks, and could be very harmful when applying fine-tuning based approaches to token-level tasks such as question answering, where it is crucial to incorporate context from both directions.

In this paper, we improve the fine-tuning based approaches by proposing BERT: **B**idirectional **E**ncoder **R**epresentations from **T**ransformers. BERT alleviates the previously mentioned unidirectionality constraint by using a "masked language model" (MLM) pre-training objective, inspired by the Cloze task (Taylor, 1953). The masked language model randomly masks some of the tokens from the input, and the objective is to predict the original vocabulary id of the masked

BERT

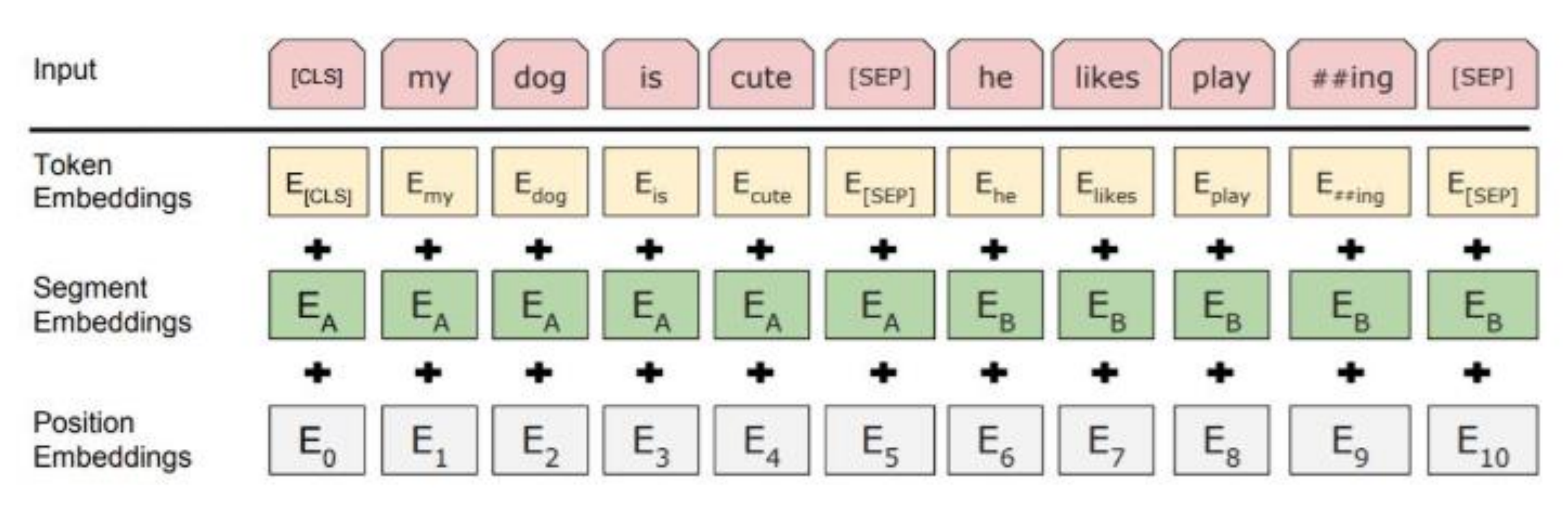
Google Brain

BERT 진정한 양방향성 (bi-directional)



Google Brain

BERT 입력 구조



Input	[CLS]	my	dog	is	cute	[SEP]	he	likes	play	##ing	[SEP]
Token Embeddings	$E_{[CLS]}$	E_{my}	E_{dog}	E_{is}	E_{cute}	$E_{[SEP]}$	E_{he}	E_{likes}	E_{play}	$E_{\#\#ing}$	$E_{[SEP]}$
	+	+	+	+	+	+	+	+	+	+	+
Segment Embeddings	E_A	E_A	E_A	E_A	E_A	E_A	E_B	E_B	E_B	E_B	E_B
	+	+	+	+	+	+	+	+	+	+	+
Position Embeddings	E_0	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	E_9	E_{10}

BERT

Google Brain

MLM (Mask Language Model)

15% [MASK] token 구성

천리

발 없는 말 이 [MASK] 간다 80%

천리

발 없는 말 이 냉장고 간다 10%

천리

발 없는 말 이 천리 간다 10%

BERT

Google Brain

NSP (Next Sentence Prediction)

NLP Task 중에 QA , NLI(Natural Language Inference)

는 두 문장 사이의 관계를 이해 하는 것이 중요

비가 눈이

[CLS] 여름에는 [MASK] 내린다[SEP][MASK] 눈이 내린다[SEP]

50% sentence 앞, 뒤가 실제 Next Sentence 사용

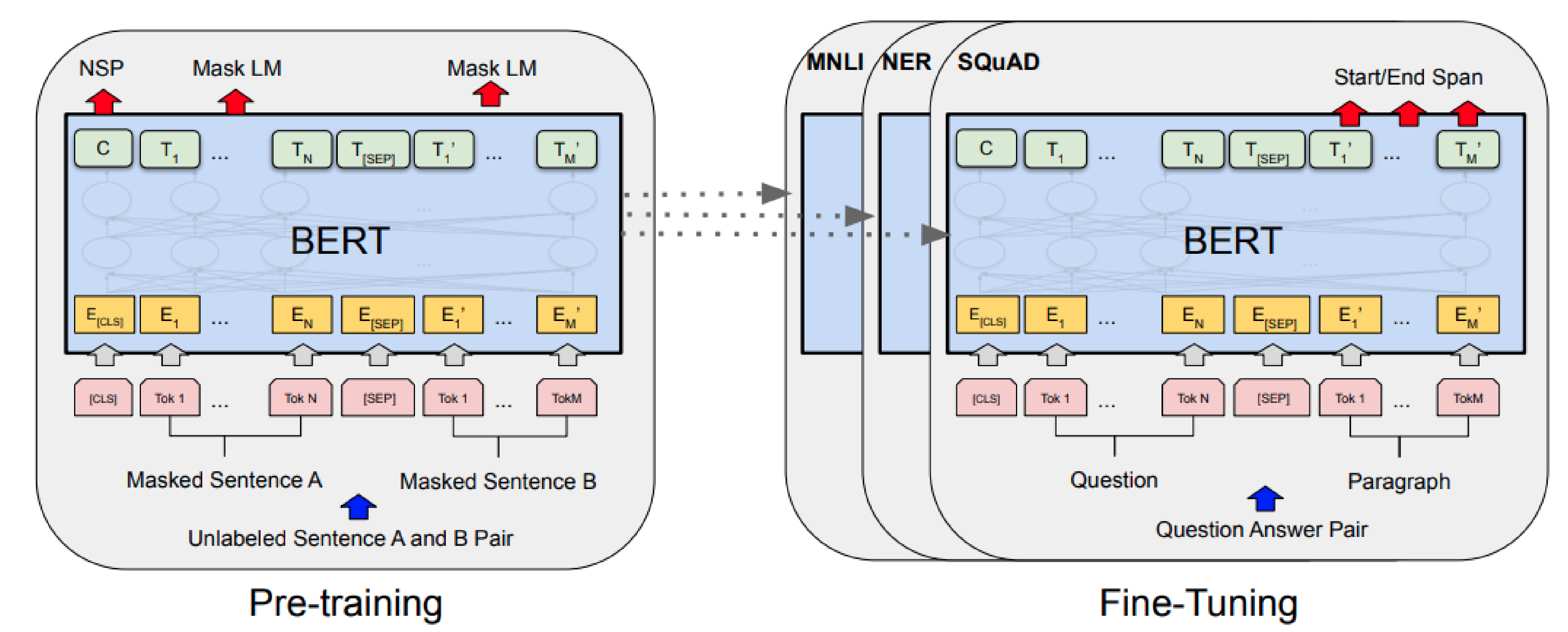
비가 눈이

[CLS] 여름에는 [MASK] 내린다[SEP][MASK] 익을수록 고개를 숙인다[SEP]

50% sentence 앞, 뒤가 Random Next Sentence 사용

Google Brain

Fine-tuning Procedure



XLNET

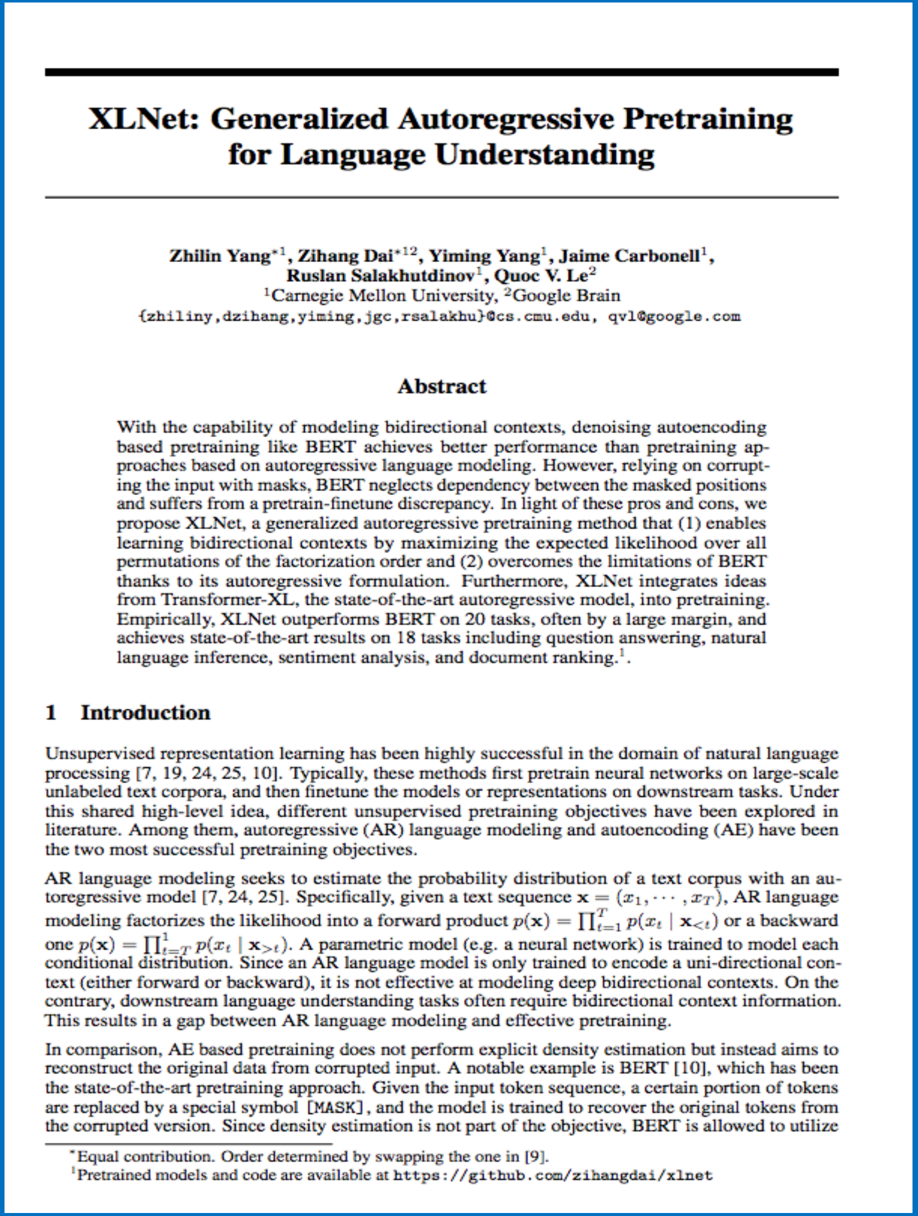
Yang et al. 2019

BERT(LARGE) 보다 Data 13GB → 130GB

Permutation Language model

Two-stream self-attention mechanism

Recurrence mechanism



XLNet: Generalized Autoregressive Pretraining for Language Understanding

Zhilin Yang[*1], **Zihang Dai**[*12], **Yiming Yang**[1], **Jaime Carbonell**[1],
Ruslan Salakhutdinov[1], **Quoc V. Le**[2]
[1]Carnegie Mellon University, [2]Google Brain
{zhiliny,dzihang,yiming,jgc,rsalakhu}@cs.cmu.edu, qvl@google.com

Abstract

With the capability of modeling bidirectional contexts, denoising autoencoding based pretraining like BERT achieves better performance than pretraining approaches based on autoregressive language modeling. However, relying on corrupting the input with masks, BERT neglects dependency between the masked positions and suffers from a pretrain-finetune discrepancy. In light of these pros and cons, we propose XLNet, a generalized autoregressive pretraining method that (1) enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and (2) overcomes the limitations of BERT thanks to its autoregressive formulation. Furthermore, XLNet integrates ideas from Transformer-XL, the state-of-the-art autoregressive model, into pretraining. Empirically, XLNet outperforms BERT on 20 tasks, often by a large margin, and achieves state-of-the-art results on 18 tasks including question answering, natural language inference, sentiment analysis, and document ranking.[1].

1 Introduction

Unsupervised representation learning has been highly successful in the domain of natural language processing [7, 19, 24, 25, 10]. Typically, these methods first pretrain neural networks on large-scale unlabeled text corpora, and then finetune the models or representations on downstream tasks. Under this shared high-level idea, different unsupervised pretraining objectives have been explored in literature. Among them, autoregressive (AR) language modeling and autoencoding (AE) have been the two most successful pretraining objectives.

AR language modeling seeks to estimate the probability distribution of a text corpus with an autoregressive model [7, 24, 25]. Specifically, given a text sequence $\mathbf{x} = (x_1, \cdots, x_T)$, AR language modeling factorizes the likelihood into a forward product $p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t \mid \mathbf{x}_{<t})$ or a backward one $p(\mathbf{x}) = \prod_{t=T}^{1} p(x_t \mid \mathbf{x}_{>t})$. A parametric model (e.g. a neural network) is trained to model each conditional distribution. Since an AR language model is only trained to encode a uni-directional context (either forward or backward), it is not effective at modeling deep bidirectional contexts. On the contrary, downstream language understanding tasks often require bidirectional context information. This results in a gap between AR language modeling and effective pretraining.

In comparison, AE based pretraining does not perform explicit density estimation but instead aims to reconstruct the original data from corrupted input. A notable example is BERT [10], which has been the state-of-the-art pretraining approach. Given the input token sequence, a certain portion of tokens are replaced by a special symbol `[MASK]`, and the model is trained to recover the original tokens from the corrupted version. Since density estimation is not part of the objective, BERT is allowed to utilize

[*] Equal contribution. Order determined by swapping the one in [9].
[1] Pretrained models and code are available at `https://github.com/zihangdai/xlnet`

XLNET

AutoRegressive Language model

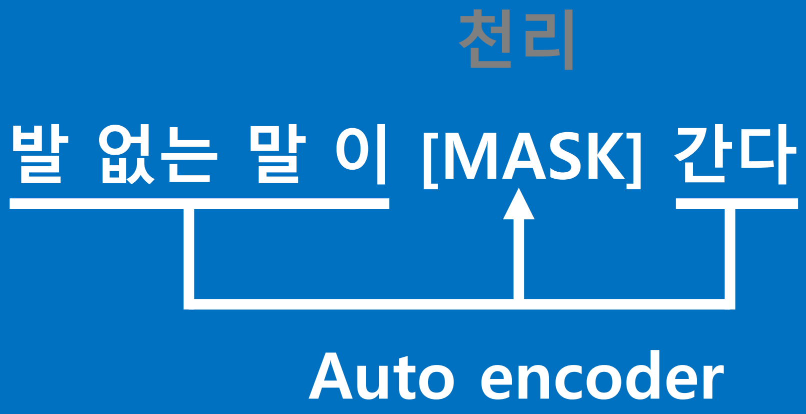
발 없는 말 이 천리 간다

→

ELMo GPT

XLNET

AutoEncoder Language model



천리

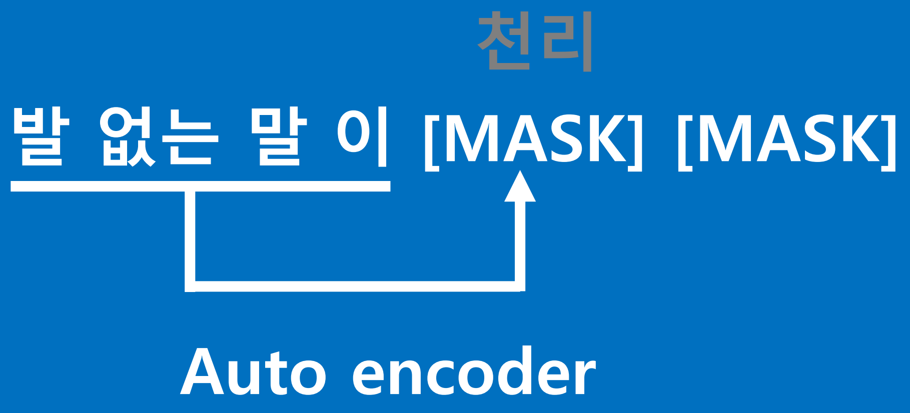
발 없는 말 이 [MASK] 간다

Auto encoder

BERT

XLNET

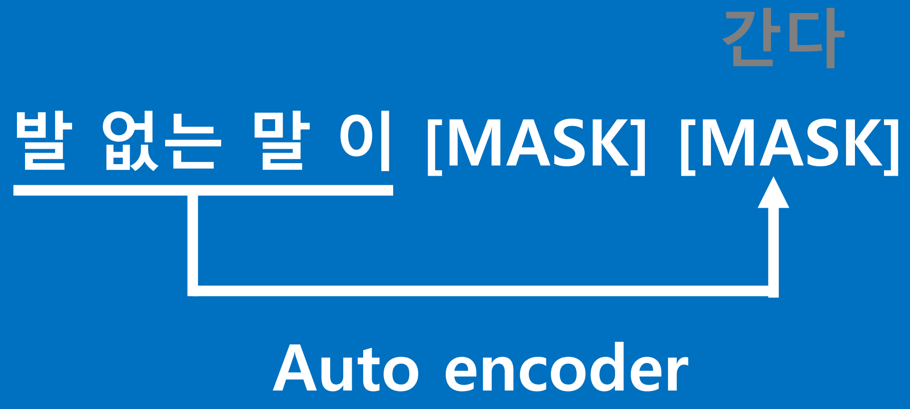
AutoEncoder Language model



천리

발 없는 말 이 [MASK] [MASK]

Auto encoder



간다

발 없는 말 이 [MASK] [MASK]

Auto encoder

BERT

XLNET

Permutation Language model

발 없는 말 이 천리 간다

없는 이 말 발 간다 천리

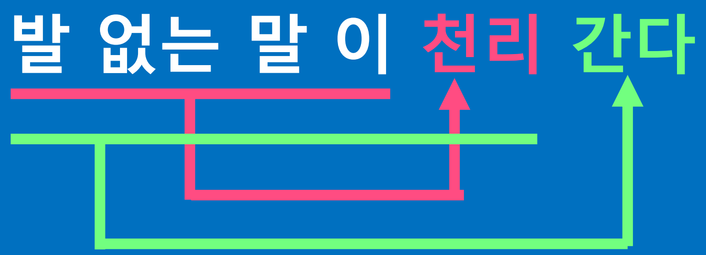
발 없는 말 이 천리 간다

Created by Arthur Shlain
from Noun Project

발 없는 천리 이 말 간다

XLNET

Permutation Language model



XLNET

Two-Stream Self Attention

[나, 어제, 학교, 갔어]

[학교, 어제, 갔어, 나] Random Shuffle 1

[학교, 어제, 나, 갔어] Random Shuffle 2

세번째 예측 한다면 ?

Two-Stream Self Attention

[나, 어제, 학교, 갔어]

[학교, 어제, 갔어, 나] Shuffle Sequence 1

[학교, 어제, 나, 갔어] Shuffle Sequence 1

세번째 예측 한다면 ?

모순이 발생한다.

앞의 단어 순서 학교, 어제는 갔으나 다른 예측 값 갔어 or 나

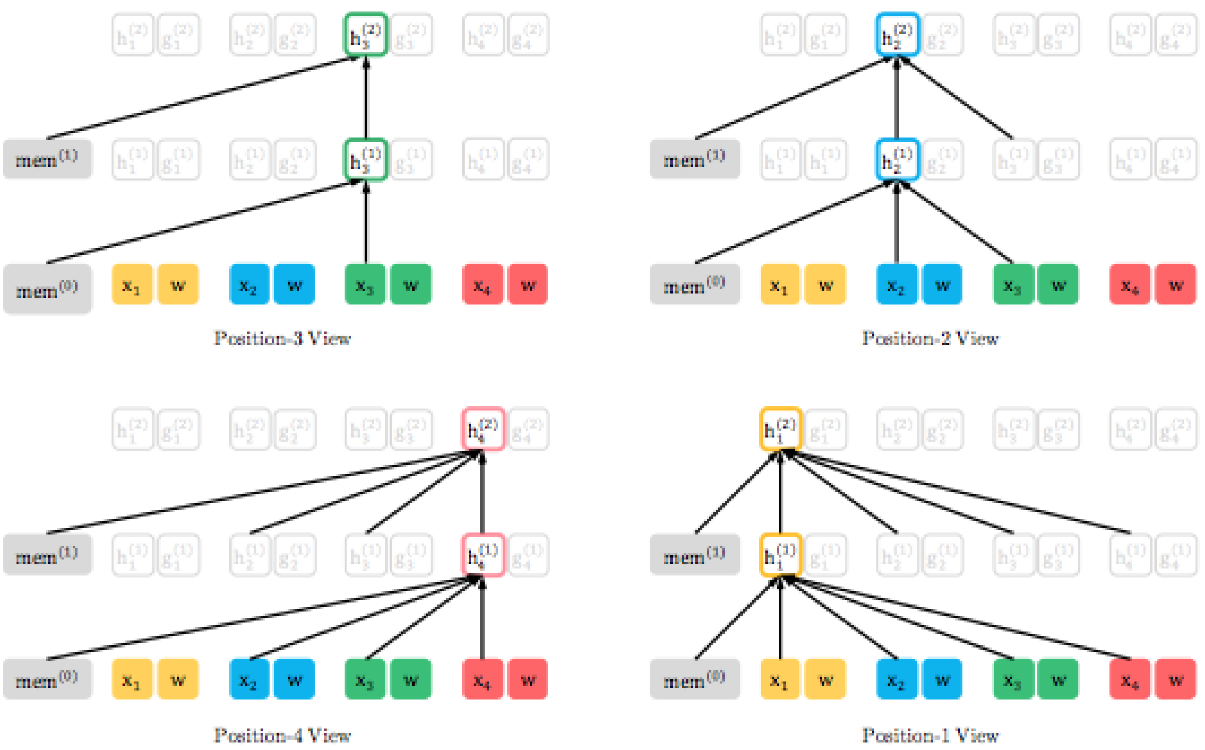
Two-Stream Self Attention

Content Stream + Query Stream

XLNET

Two-Stream Self Attention

Content Stream $h_{z_t}^{(m)} \leftarrow \text{Attention}(\text{Q} = h_{z_t}^{(m-1)}, \text{KV} = \mathbf{h}_{\mathbf{z}_{\leq t}}^{(m-1)}; \theta),$ (content stream: use both z_t and x_{z_t}).


mem(0)
mem(1)
x1 w x2 w x3 w x4 w
h1(1) g1(1) h2(1) g2(1) h3(1) g3(1) h4(1) g4(1)
h1(2) g1(2) h2(2) g2(2) h3(2) g3(2) h4(2) g4(2)

Position-3 View

Position-2 View

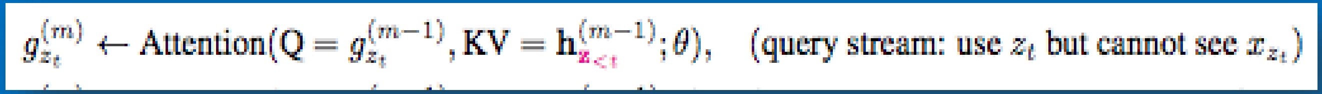
Position-4 View

Position-1 View

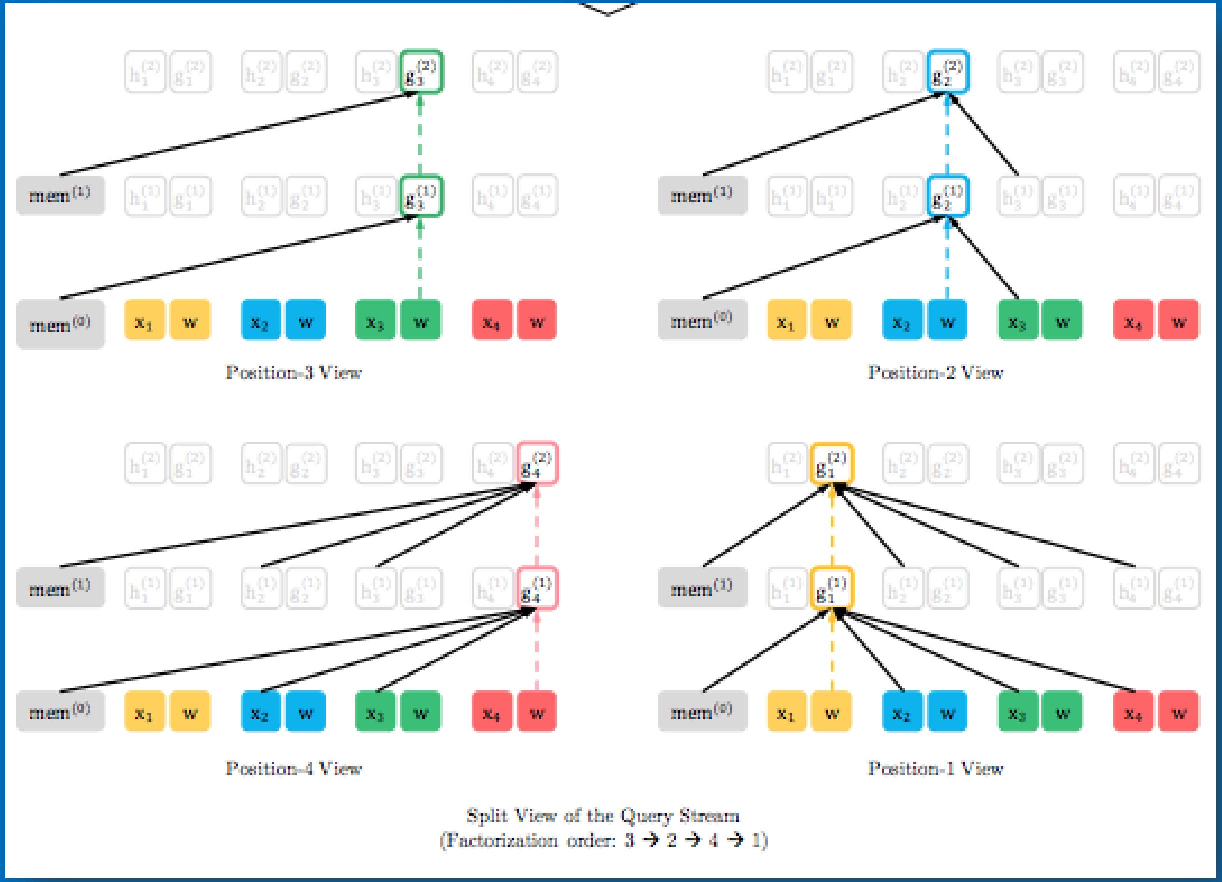
Split View of the Content Stream
(Factorization order: 3 → 2 → 4 → 1)

Two-Stream Self Attention

Query Stream



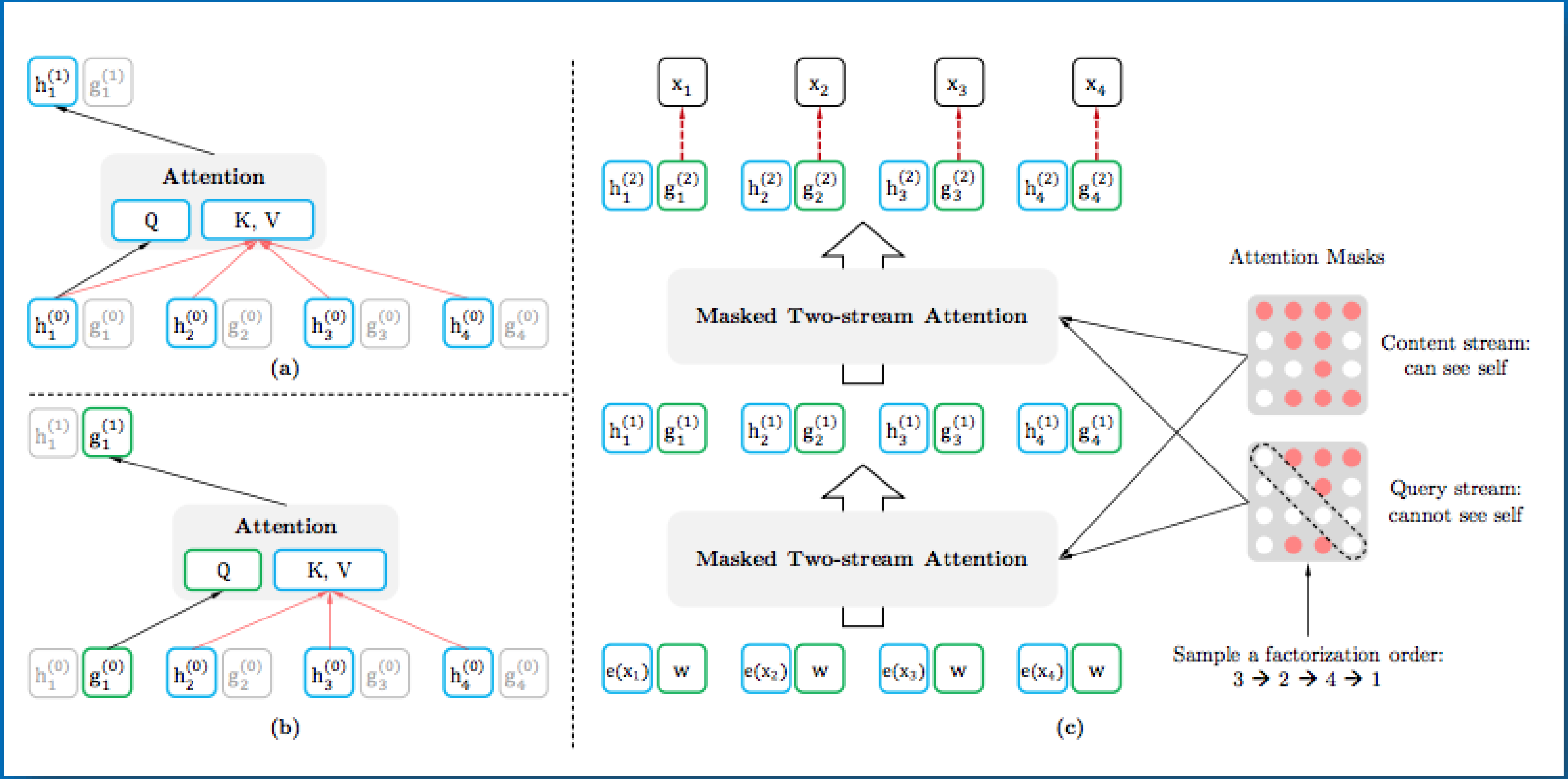
$$g_{z_t}^{(m)} \leftarrow \text{Attention}(\text{Q} = g_{z_t}^{(m-1)}, \text{KV} = \mathbf{h}_{\mathbf{z}_{<t}}^{(m-1)}; \theta), \quad \text{(query stream: use } z_t \text{ but cannot see } x_{z_t}\text{)}$$


mem$^{(0)}$
mem$^{(1)}$
x_1 w x_2 w x_3 w x_4 w
$h_1^{(1)}$ $g_1^{(1)}$ $h_2^{(1)}$ $g_2^{(1)}$ $h_3^{(1)}$ $g_3^{(1)}$ $h_4^{(1)}$ $g_4^{(1)}$
$h_1^{(2)}$ $g_1^{(2)}$ $h_2^{(2)}$ $g_2^{(2)}$ $h_3^{(2)}$ $g_3^{(2)}$ $h_4^{(2)}$ $g_4^{(2)}$
Position-3 View
Position-2 View
Position-4 View
Position-1 View

Split View of the Query Stream
(Factorization order: 3 → 2 → 4 → 1)

XLNET

Two-Stream Self Attention


$h_1^{(1)}$ $g_1^{(1)}$
Attention
Q
K, V
$h_1^{(0)}$ $g_1^{(0)}$ $h_2^{(0)}$ $g_2^{(0)}$ $h_3^{(0)}$ $g_3^{(0)}$ $h_4^{(0)}$ $g_4^{(0)}$
(a)
$h_1^{(1)}$ $g_1^{(1)}$
Attention
Q
K, V
$h_1^{(0)}$ $g_1^{(0)}$ $h_2^{(0)}$ $g_2^{(0)}$ $h_3^{(0)}$ $g_3^{(0)}$ $h_4^{(0)}$ $g_4^{(0)}$
(b)
x_1 x_2 x_3 x_4
$h_1^{(2)}$ $g_1^{(2)}$ $h_2^{(2)}$ $g_2^{(2)}$ $h_3^{(2)}$ $g_3^{(2)}$ $h_4^{(2)}$ $g_4^{(2)}$
Masked Two-stream Attention
$h_1^{(1)}$ $g_1^{(1)}$ $h_2^{(1)}$ $g_2^{(1)}$ $h_3^{(1)}$ $g_3^{(1)}$ $h_4^{(1)}$ $g_4^{(1)}$
Masked Two-stream Attention
$e(x_1)$ w $e(x_2)$ w $e(x_3)$ w $e(x_4)$ w
Attention Masks
Content stream: can see self
Query stream: cannot see self
Sample a factorization order: 3 → 2 → 4 → 1
(c)

Two-Stream Self Attention

[나, 어제, 학교, 갔어]

[학교, 어제, 갔어, 나] Random Shuffle 1

[학교, 어제, 나, 갔어] Random Shuffle 2

세번째 예측 한다면 ?

모순을 해결한다.

Recurrence mechanism

Transformer-XL

Yang et al. 2019

transformer 확장 Version

Transformer의 단점 고정된 길이의 문맥 정보만

활용할 수 있는 부분 개선(Segment Recurrence)

Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

Zihang Dai*[12], Zhilin Yang*[12], Yiming Yang[1], Jaime Carbonell[1],
Quoc V. Le[2], Ruslan Salakhutdinov[1]
[1]Carnegie Mellon University, [2]Google Brain
{dzihang,zhiliny,yiming,jgc,rsalakhu}@cs.cmu.edu, qvl@google.com

Abstract

Transformers have a potential of learning longer-term dependency, but are limited by a fixed-length context in the setting of language modeling. We propose a novel neural architecture *Transformer-XL* that enables learning dependency beyond a fixed length without disrupting temporal coherence. It consists of a segment-level recurrence mechanism and a novel positional encoding scheme. Our method not only enables capturing longer-term dependency, but also resolves the context fragmentation problem. As a result, Transformer-XL learns dependency that is 80% longer than RNNs and 450% longer than vanilla Transformers, achieves better performance on both short and long sequences, and is up to 1,800+ times faster than vanilla Transformers during evaluation. Notably, we improve the state-of-the-art results of bpc/perplexity to 0.99 on enwiki8, 1.08 on text8, 18.3 on WikiText-103, 21.8 on One Billion Word, and 54.5 on Penn Treebank (without finetuning). When trained only on WikiText-103, Transformer-XL manages to generate reasonably coherent, novel text articles with thousands of tokens. Our code, pretrained models, and hyperparameters are available in both Tensorflow and PyTorch[1].

1 Introduction

Language modeling is among the important problems that require modeling long-term dependency, with successful applications such as unsupervised pretraining (Dai and Le, 2015; Peters et al., 2018; Radford et al., 2018; Devlin et al., 2018). However, it has been a challenge to equip neural networks with the capability to model long-term dependency in sequential data. Recurrent neural networks (RNNs), in particular Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997), have been a standard solution to language modeling and obtained strong results on multiple benchmarks. Despite the wide adaption, RNNs are difficult to optimize due to gradient vanishing and explosion (Hochreiter et al., 2001), and the introduction of gating in LSTMs and the gradient clipping technique (Graves, 2013) might not be sufficient to fully address this issue. Empirically, previous work has found that LSTM language models use 200 context words on average (Khandelwal et al., 2018), indicating room for further improvement.

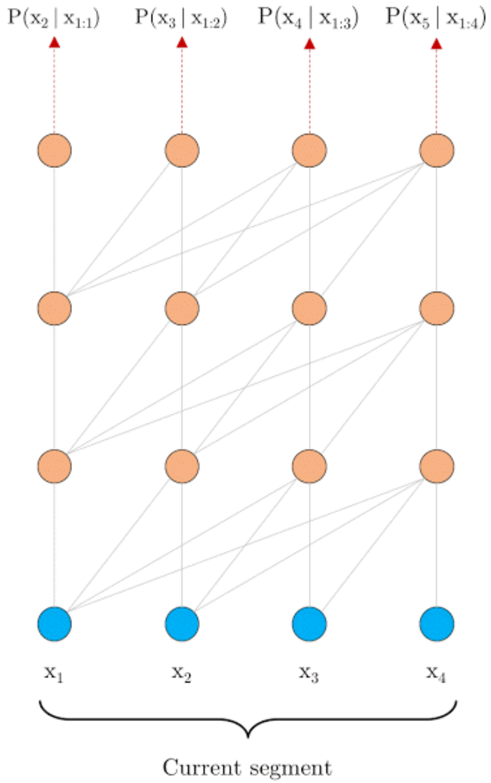
On the other hand, the direct connections between long-distance word pairs baked in attention mechanisms might ease optimization and enable the learning of long-term dependency (Bahdanau et al., 2014; Vaswani et al., 2017). Recently, Al-Rfou et al. (2018) designed a set of auxiliary losses to train deep Transformer networks for character-level language modeling, which outperform LSTMs by a large margin. Despite the success, the LM training in Al-Rfou et al. (2018) is performed on separated fixed-length segments of a few hundred characters, without any information flow across segments. As a consequence of the fixed context length, the model cannot capture any longer-term dependency beyond the predefined context length. In addition, the fixed-length segments are created by selecting a consecutive chunk of symbols without respecting the sentence or any other semantic boundary. Hence, the model lacks necessary contextual information needed to well predict the first few symbols, leading to inefficient optimization and inferior performance. We refer to this problem as *context fragmentation*.

To address the aforementioned limitations of fixed-length contexts, we propose a new architecture called Transformer-XL (meaning extra long). We introduce the notion of recurrence into our

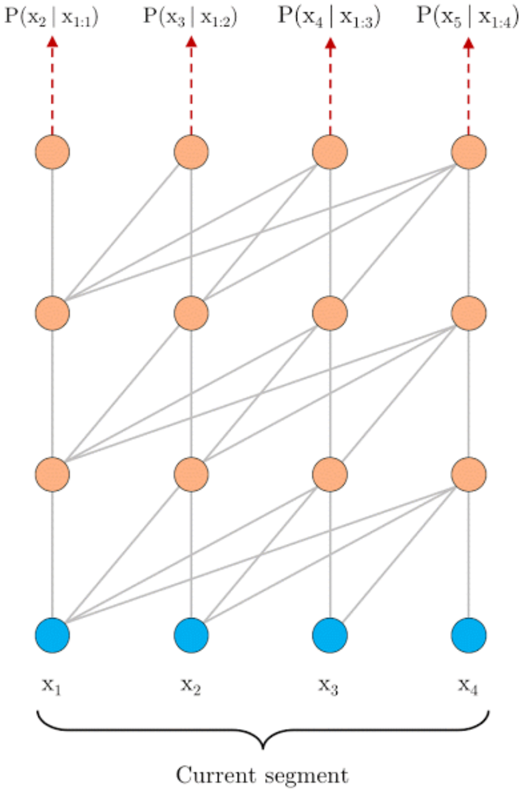
*Equal contribution. Order determined by swapping the one in Yang et al. (2017).

[1] https://github.com/kimiyoung/transformer-xl

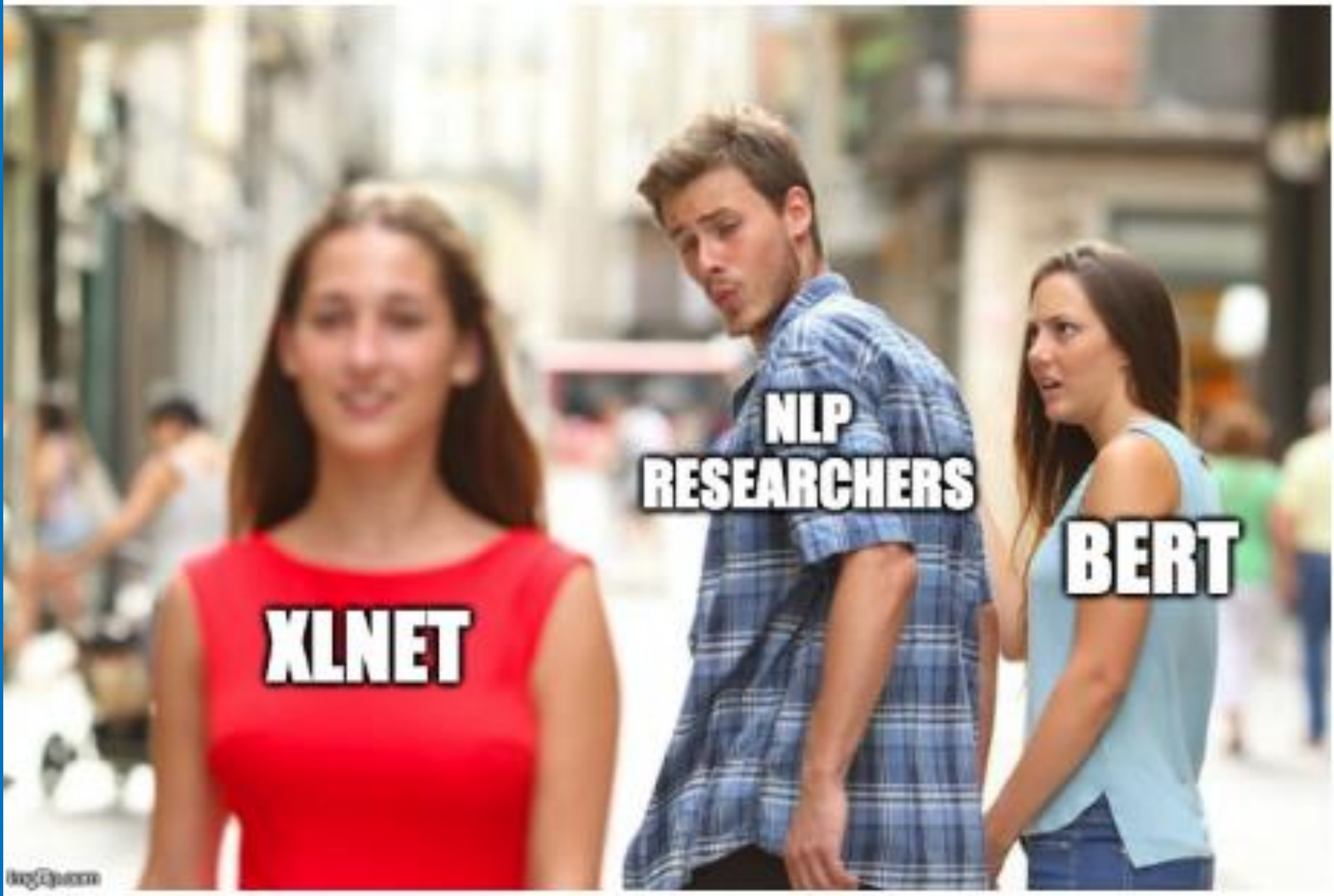
Segment Recurrence



Segment Recurrence



XLNET



RoBERTa

Liu et al. (2019)

BERT 해부해서 최적화 시킴

BERT(LARGE) 보다 Data 13GB → 160GB

LSP(Next Sentence Prediction) 제거

Dynamic masking 사용

Batch Size 8K 조정

Adam Beta2 0.98 조정

Max Steps 500K(오십만) 조정

RoBERTa: A Robustly Optimized BERT Pretraining Approach

Yinhan Liu*§ Myle Ott*§ Naman Goyal*§ Jingfei Du*§ Mandar Joshi†
Danqi Chen§ Omer Levy§ Mike Lewis§ Luke Zettlemoyer†§ Veselin Stoyanov§

† Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA
{mandar90,lsz}@cs.washington.edu

§ Facebook AI
{yinhanliu,myleott,naman,jingfeidu,
danqi,omerlevy,mikelewis,lsz,ves}@fb.com

Abstract

Language model pretraining has led to significant performance gains but careful comparison between different approaches is challenging. Training is computationally expensive, often done on private datasets of different sizes, and, as we will show, hyperparameter choices have significant impact on the final results. We present a replication study of BERT pretraining (Devlin et al., 2019) that carefully measures the impact of many key hyperparameters and training data size. We find that BERT was significantly undertrained, and can match or exceed the performance of every model published after it. Our best model achieves state-of-the-art results on GLUE, RACE and SQuAD. These results highlight the importance of previously overlooked design choices, and raise questions about the source of recently reported improvements. We release our models and code.[1]

1 Introduction

Self-training methods such as ELMo (Peters et al., 2018), GPT (Radford et al., 2018), BERT (Devlin et al., 2019), XLM (Lample and Conneau, 2019), and XLNet (Yang et al., 2019) have brought significant performance gains, but it can be challenging to determine which aspects of the methods contribute the most. Training is computationally expensive, limiting the amount of tuning that can be done, and is often done with private training data of varying sizes, limiting our ability to measure the effects of the modeling advances.

We present a replication study of BERT pretraining (Devlin et al., 2019), which includes a careful evaluation of the effects of hyperparmeter tuning and training set size. We find that BERT was significantly undertrained and propose an improved recipe for training BERT models, which we call RoBERTa, that can match or exceed the performance of all of the post-BERT methods. Our modifications are simple, they include: (1) training the model longer, with bigger batches, over more data; (2) removing the next sentence prediction objective; (3) training on longer sequences; and (4) dynamically changing the masking pattern applied to the training data. We also collect a large new dataset (CC-NEWS) of comparable size to other privately used datasets, to better control for training set size effects.

When controlling for training data, our improved training procedure improves upon the published BERT results on both GLUE and SQuAD. When trained for longer over additional data, our model achieves a score of 88.5 on the public GLUE leaderboard, matching the 88.4 reported by Yang et al. (2019). Our model establishes a new state-of-the-art on 4/9 of the GLUE tasks: MNLI, QNLI, RTE and STS-B. We also match state-of-the-art results on SQuAD and RACE. Overall, we re-establish that BERT's masked language model training objective is competitive with other recently proposed training objectives such as perturbed autoregressive language modeling (Yang et al., 2019).[2]

In summary, the contributions of this paper are: (1) We present a set of important BERT design choices and training strategies and introduce

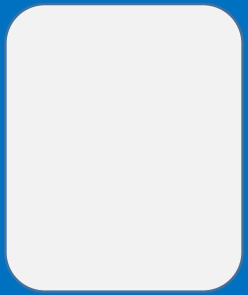
*Equal contribution.

[1] Our models and code are available at: https://github.com/pytorch/fairseq

[2] It is possible that these other methods could also improve with more tuning. We leave this exploration to future work.

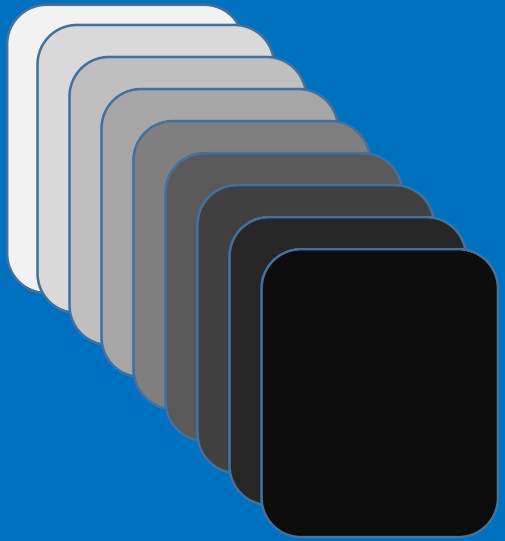
RoBERTa

Bert Static Masking



각 Epoch 마다 반복적 사용

RoBERTa Dynamic Masking



각 Epoch 마다 다른 Masking Data 사용

RoBERTa

작은 결정들이 성능에 큰 영향을 준다.

수 많은 인형 눈알 붙이는 고통을 이겨낸
Facebook AI에게 박수를 ~

RoBERTa

Finetuning

Fine-tuning이란?

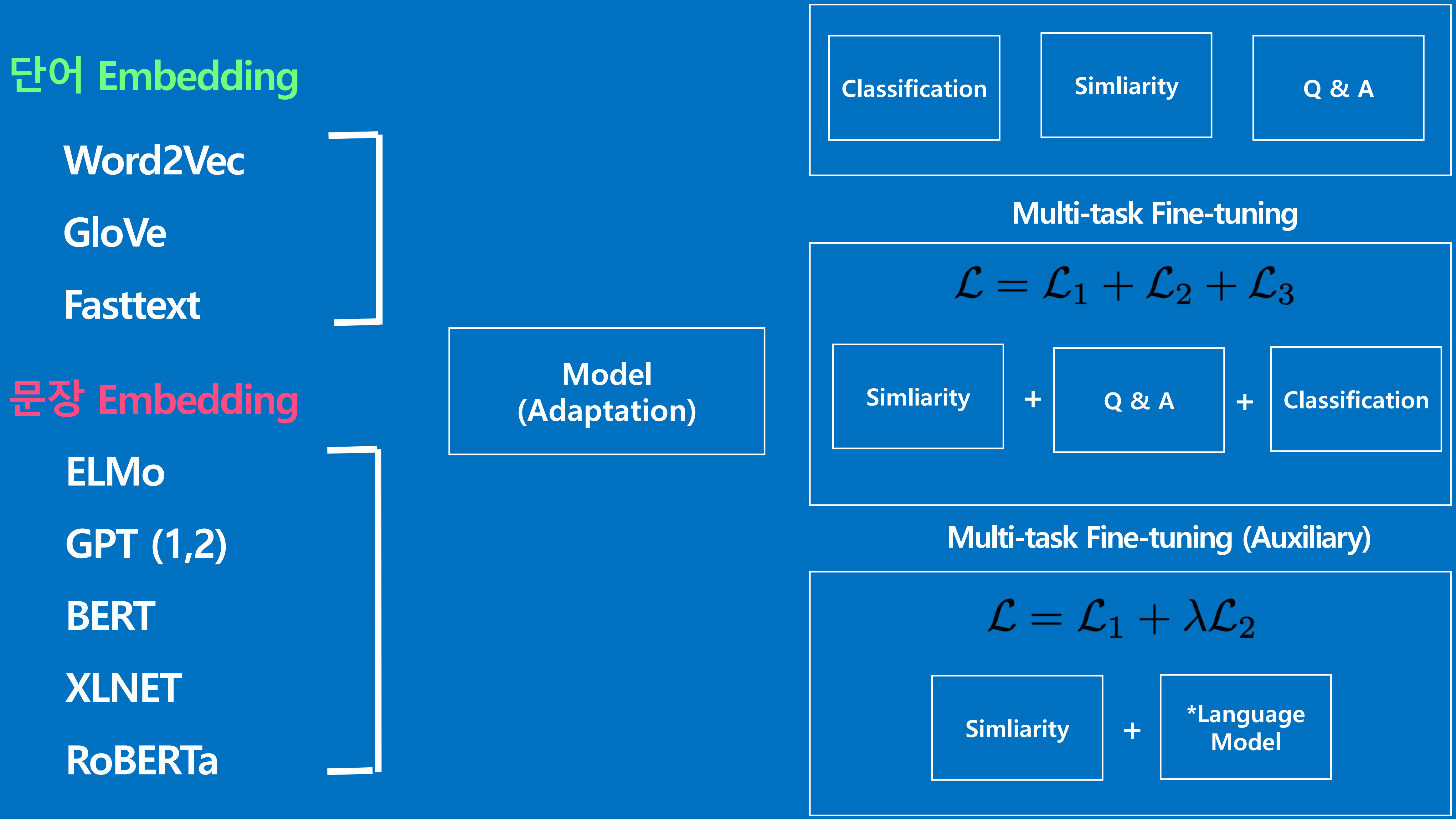
VGG16, ResNet같은 이미지 분야에서 먼저 사용되기 시작

학습되어진 모델을 기반으로 아키텍처를 특정 Task에 맞게 수정

초기에 임의의 값으로 학습하는 것보다 좋은 성능을 (Good Initial point)

Pre-training 모델을 활용 해 보자

Finetuning



단어 Embedding

Word2Vec

GloVe

Fasttext

문장 Embedding

ELMo

GPT (1,2)

BERT

XLNET

RoBERTa

Model (Adaptation)

Single Fine-tuning

Classification

Simliarity

Q & A

Multi-task Fine-tuning

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3$$

Simliarity + Q & A + Classification

Multi-task Fine-tuning (Auxiliary)

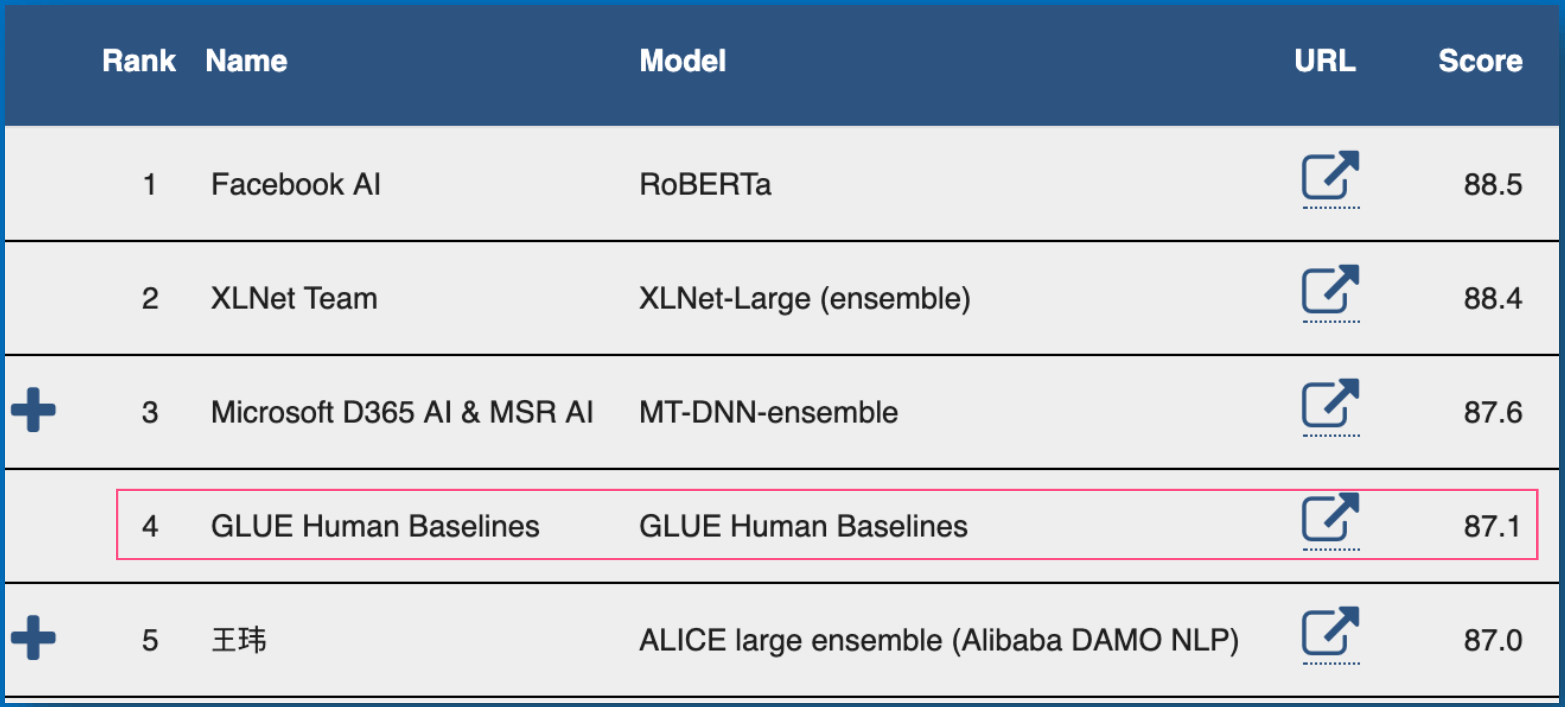
$$\mathcal{L} = \mathcal{L}_1 + \lambda\mathcal{L}_2$$

Simliarity + *Language Model

Natural Language Beyond People

Natural Language Beyond People

GLUE Task: 사람의 성능을 뛰어넘다

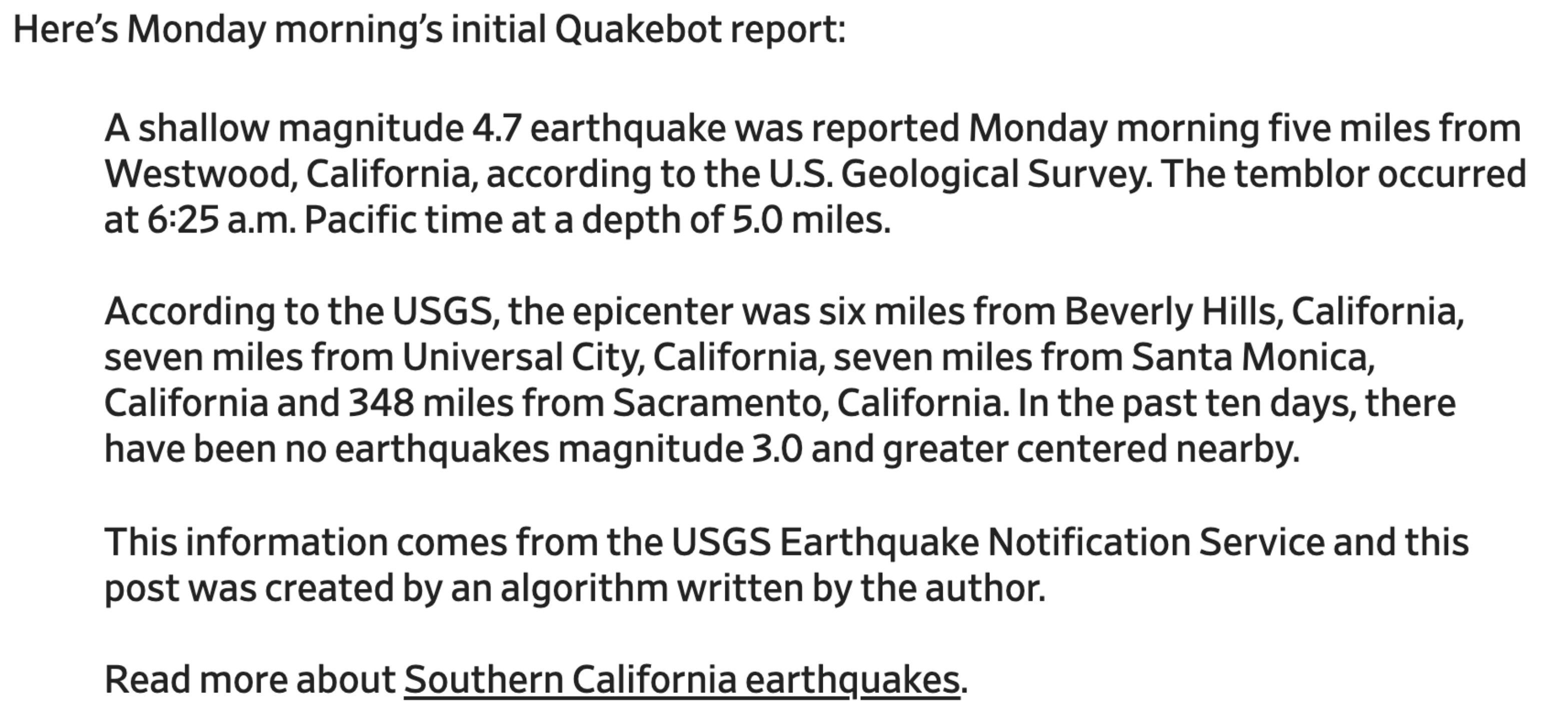


	Rank	Name	Model	URL	Score
	1	Facebook AI	RoBERTa		88.5
	2	XLNet Team	XLNet-Large (ensemble)		88.4
+	3	Microsoft D365 AI & MSR AI	MT-DNN-ensemble		87.6
	4	GLUE Human Baselines	GLUE Human Baselines		87.1
+	5	王玮	ALICE large ensemble (Alibaba DAMO NLP)		87.0

언어 이해, 언어 독해, 문장 분류 등 총 12개의 Task

**https://gluebenchmark.com/*

로봇 저널리즘: LA Times의 Quakebot



Here’s Monday morning’s initial Quakebot report:

> A shallow magnitude 4.7 earthquake was reported Monday morning five miles from Westwood, California, according to the U.S. Geological Survey. The temblor occurred at 6:25 a.m. Pacific time at a depth of 5.0 miles.
>
> According to the USGS, the epicenter was six miles from Beverly Hills, California, seven miles from Universal City, California, seven miles from Santa Monica, California and 348 miles from Sacramento, California. In the past ten days, there have been no earthquakes magnitude 3.0 and greater centered nearby.
>
> This information comes from the USGS Earthquake Notification Service and this post was created by an algorithm written by the author.
>
> Read more about Southern California earthquakes.

지진이 일어나자마자, 즉각적으로 핵심 정보를 전달

https://slate.com/technology/2014/03/quakebot-los-angeles-times-robot-journalist-writes-article-on-la-earthquake.html

AI가 생성한 대화를 찾아보세요 1

니 이름이 뭐야?

꽤 유명한데 아직 모르시나 봐요

전 언제 어디서든 도움을 주는 친구 클로바입니다.

식사는 언제 하실 꺼에요?

당신이 먹는 것만 봐도 배가 불러요

와이파이 한 조각을 먹고 왔답니다.

여행가고 싶어

바보는 방황을 하고 현명한 사람은 여행을 떠난다고 해요

여행은 인생의 좋은 경험이 될 거에요

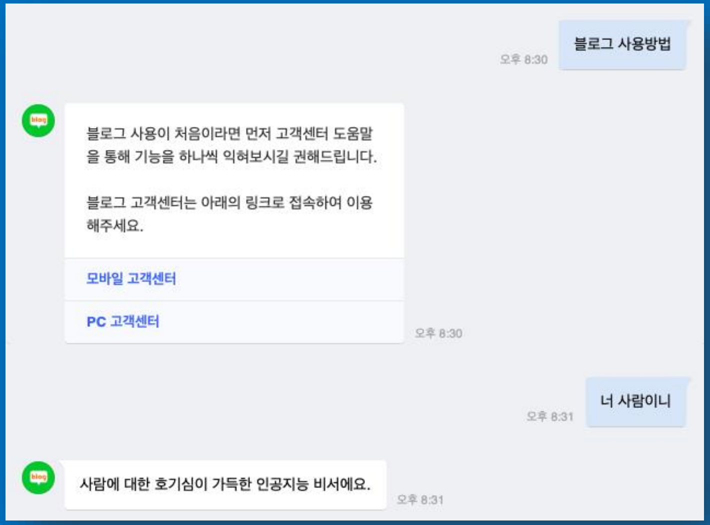
AI가 생성한 대화를 찾아보세요 2

오늘 달리기 시합 있어	커피 한잔의 여유는 필요하죠 건강을 위해 운동을 해 보세요
어이 상실	네, 부르셨나요 누구나 실수는 할 수 있어요
지금 이 순간	보고 싶은 사람이 있다는 건 좋은 일이에요 당신과 함께라면 항상 행복해요

Forward Hot Domain

일상생활에서의 변화

AI 통,번역을 통해 해외여행이 쉬워지고



블로그 사용방법
오후 8:30

블로그 사용이 처음이라면 먼저 고객센터 도움말을 통해 기능을 하나씩 익혀보시길 권해드립니다.

블로그 고객센터는 아래의 링크로 접속하여 이용해주세요.

모바일 고객센터

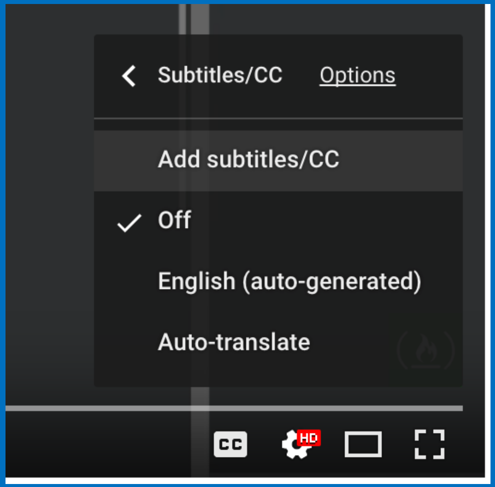
PC 고객센터

오후 8:30

너 사람이니
오후 8:31

사람에 대한 호기심이 가득한 인공지능 비서에요.
오후 8:31

챗봇을 통해 24시간 서비스에 대해 질문하며



동영상 자동 자막 & 번역을 본다

융합의 중요성

AI, 특히 자연어처리 분야는 이제 발전

앞으로 뻗어 나갈려면, 다양한 분야의 백그라운드와

아이디어의 융합이 필수

Forward Hot Domain

https://www.youtube.com/watch?v=cj9v44e0kic

융합의 중요성 – Music Generation (IBM)

5년간의 문화 데이터를 수집, 각 연도의 감성레벨 분석

뉴욕 타임즈 기사, 영화 대본, 노래 기사, 트위터 및 블로그 등을 참조하여,

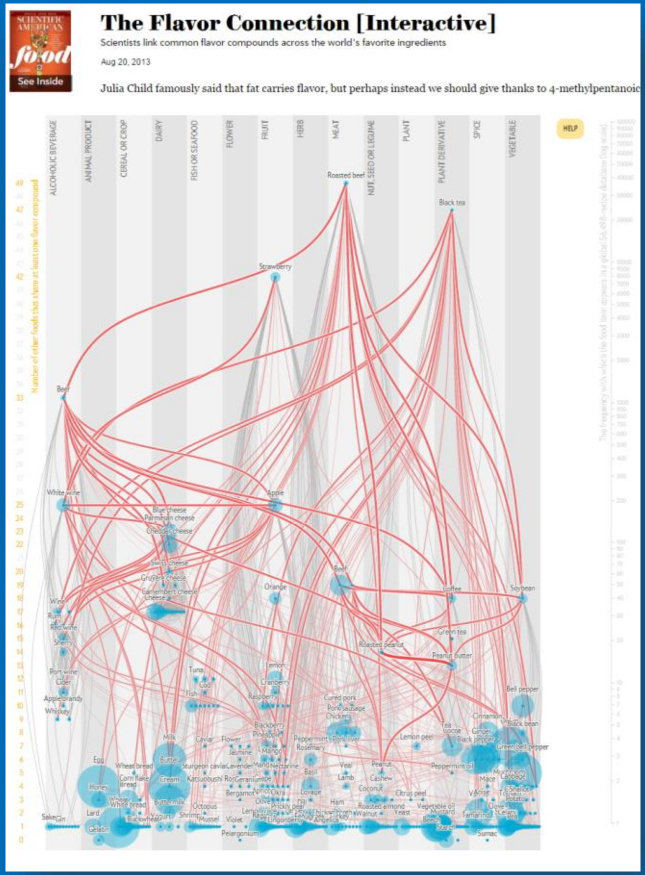
빌보드 차트 상위에 기록되었던 음악을 바탕으로

음악 가사와 음악 트랜드를 추천

융합의 중요성 – Chef Watson (IBM)

Cognitive Cooking
with Chef Watson
Recipes for Innovation from IBM & the Institute of Culinary Education

레시피 책 (Cognitive Cooking)



The Flavor Connection [Interactive]

Scientists link common flavor compounds across the world's favorite ingredients

Aug 20, 2013

Julia Child famously said that fat carries flavor, but perhaps instead we should give thanks to 4-methylpentanoic

Flavor Map: 구운 쇠고기와 잘어울리는 조합

미국 요리 잡지 Bon Appétit와 협업

사용할 재료 입력

음식의 종류 입력

요리의 스타일, 식사 분위기

요리의 스타일 및 식사의 분위기

수 많은 재료의 향과 맛,

영양의 조합을 TEXT로 공부하여,

새로운 레시피를 추천

http://chefnews.kr/archives/1497

Forward Hot Domain

https://www.youtube.com/watch?v=R2mC-NUAmMk

융합의 중요성 – Seeing AI (MS)

시각 장애인 주변 모습을 말로 설명

사람이 누구인지, 상대방의 표정이 무엇인지

바코드를 스캔하여 제품을 스캔하고, 문서를 읽어주고,

음식점의 메뉴까지 설명해주는

Forward Hot Domain

앞으로 자연어의 발전은 어떤 미래를 가져다 줄까?

음성 비서: 나에게 정말 필요한 관리 서비스를 제공

진화하는 검색: 질문 기반의 검색

전문 지식 질문 (법률, 세금, 정책, 의료, 공학 등)을 대답

"자율 주행 모델을 만드려면 어떤 논문을 참고해야 해?"

"부동산 세금을 줄이려면, 어떻게 해야 해?"

내가 원하는 콘텐츠를 만들어주는 AI: 소설 및 시나리오 작성

감사합니다.