

We are Pythonistas

금융 언어 이해를 위해 개발된 ALBERT 토크아보기 with Transformers

오성우

발표자 소개

오성우

이메일: david.oh0126@gmail.com

블로그: sakco.tistory.com

다양한 산업의 살아있는 데이터를 직접 보는 것을 좋아합니다
비즈니스 문제 해결이 훨씬 더 중요하다고 생각하는 요즘입니다
공유의 힘을 믿으며 조그만 것이라도 공유하려 합니다

NLP - Domain Adaptation, Pruning, Input adaptive inference

AutoML, TDA, ...

기여하고자 하는 부분

작지만 공헌... NLP 발전을 위해, 그 중에서도 금융 & 한국어 NLP 발전을 위해

도움이 되었으면...

- 은행이 공개했다는 KB-ALBERT가 뭔지 궁금하신 분들
- 최신 SOTA 기술 등을 손쉽게 사용해보고 싶으신 분들
- 자신이 가진 금융 텍스트로 분석해보고 싶으신 분들

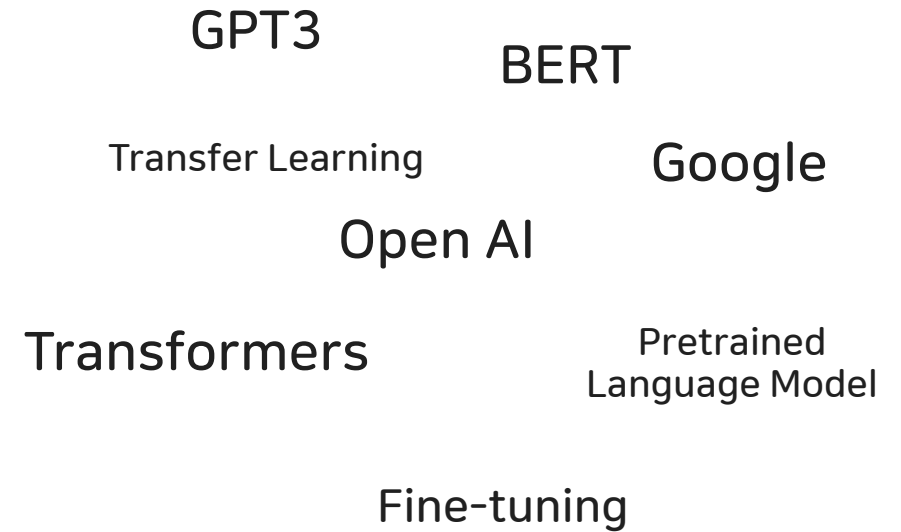
발표 순서

1. 금융 도메인에 특화된 언어모델, KB-ALBERT 소개
2. Hugging Face's Transformers 소개
3. Fine-tuning KB-ALBERT with Transformers

최신 인공지능? 언어모델?



Photo by Brett Jordan on Unsplash



금융 도메인에 특화된 언어모델 KB-ALBERT?

금융 도메인에 특화된 언어모델 KB-ALBERT?

언어모델이란?

“A language model is a probability distribution over sequences of words.”

– *Wikipedia*

언어모델은 언어적 특징을 이해하고 처리할 수 있도록 범용적인 목적으로 학습된 모델
학습된 언어모델을 활용하여 질문-답변 문제나 문서분류, 감성분석과 같은
세부적인 목적의 자연어처리 문제를 해결

인공지능 기반 언어모델의 학습은 실제 사람이 언어를 학습하는 방식과 유사

영어를 공부할 때

일반적으로 영작, 빈칸 채우기와 같은 문제풀이를 반복 수행

According to the business report, Samsung's best-selling television model is the one with a _____ screen, as consumers these days prefer a bigger visual experience.

(1) flat (2) frequent (3) large (4) spacious

빈칸에 들어갈 정답 단어는?

정답은 (3) large

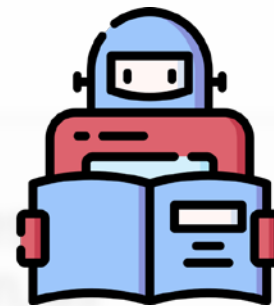
According to the business report, Samsung's best-selling television model is the one with a _____ screen, as consumers these days prefer a bigger visual experience.

(1) flat (2) frequent **(3) large** (4) spacious

인공지능 언어모델의 사전학습



사람은 영작을 하거나 빈칸에 올 단어를 추론
하는 퀴즈 등을 풀이하면서
영어의 어휘나 문법 등을 학습

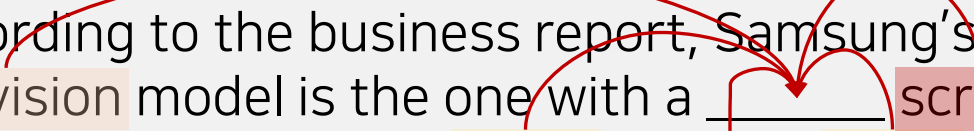


인공지능 언어모델은 수십억 개의 문장으로부터
빈칸에 올 단어를 예측하는 훈련을 반복하면서
언어의 어휘나 구조적 특징 등을 학습

정답을 찾기 위한 추론 과정

빈칸에 들어올 정답 단어를 추론하기 위해 일부 주변 단어에 집중하여 풀이

According to the business report, Samsung's best-selling television model is the one with a _____ screen, as consumers these days prefer a bigger visual experience.



(1) flat (2) frequent **(3) large** (4) spacious

인공지능 언어모델의 집중을 통한 예측

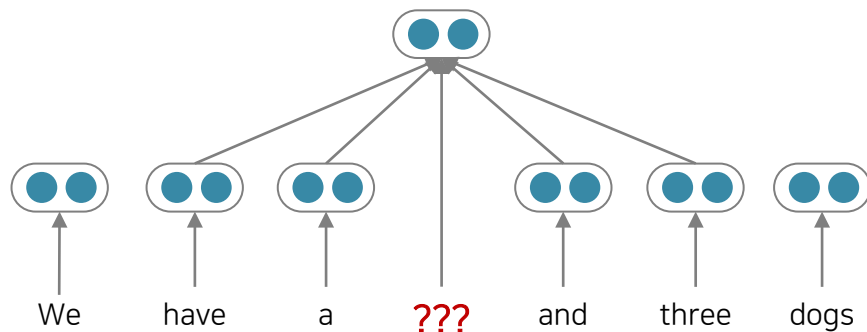
문장 안에서 단어들은 자기상관적인 형태로
주변 단어들과의 관계를 학습하면서 **좀 더 집중**해야 하는 단어가 무엇인지를 보고
빈칸의 단어를 예측

According to the business report, Samsung's best-selling
television model is the one with a _____ screen, as
consumers these days prefer a bigger visual experience.

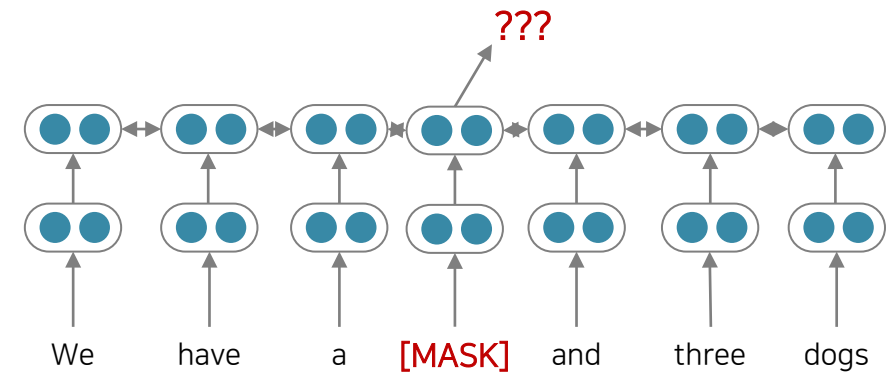
Word Representation with Pretraining

사전학습(Pretraining)을 통해 언어모델은
실제 사람이 주변 단어를 통해 빈칸에 들어갈 말을 추론하듯이
주변 맥락으로부터 word sequence에 특정 단어가 발생할 확률을 학습
최근 BERT 등의 transformer 아키텍처를 사용하며 Context-dependent 형태로 발전

Word2Vec (Mikolov et al., 2013)



BERT (Devlin et al., 2019)



Contextual Embedding Vector

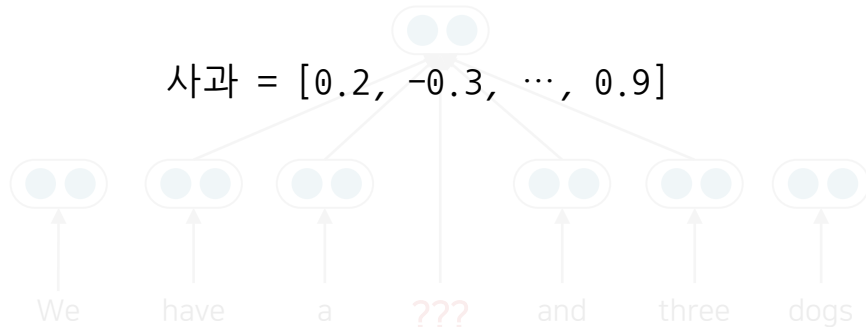
Context-Independent Word Vectors

하나의 vector 안에 특정 단어가 가질 수 있는
다양한 의미를 함축적으로 표현

Word2Vec (Mikolov et al., 2013)

배 = $[0.4, -0.5, \dots, 0.1]$

사과 = $[0.2, -0.3, \dots, 0.9]$



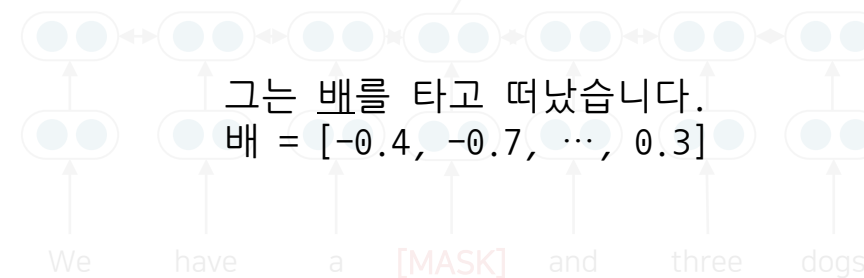
Context-Dependent Word Vectors

주변 단어와의 관계, 단어의 문장 내 위치 등에 의해
같은 단어라도 다양한 vector로 표현

BERT (Devlin et al., 2019)

나는 어제 사과와 배를 샀습니다.
배 = $[0.4, -0.3, \dots, 0.2]$

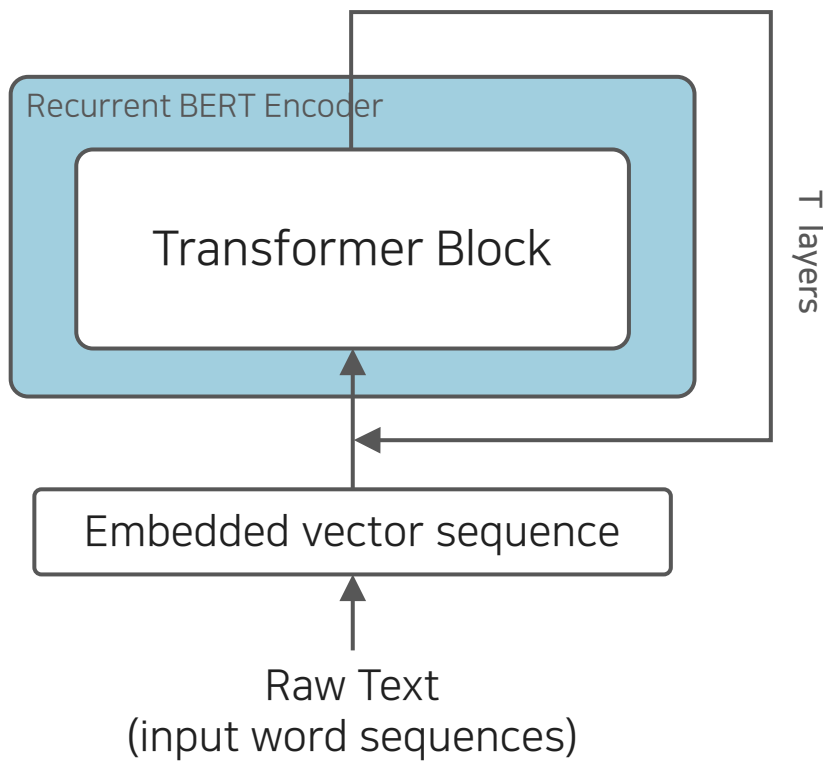
그는 배를 타고 떠났습니다.
배 = $[-0.4, -0.7, \dots, 0.3]$



금융 도메인에 특화된 언어모델 **KB-ALBERT?**

ALBERT

ALBERT(Lan et al., 2019)는 A Lite BERT의 약자



- Google Research에서 2019년 9월 공개
- 2018년 공개한 BERT의 경량화된 형태
- Cross-layer parameter sharing을 통해 모델 파라미터의 수가 감소 (10x~20x)
- Transformer block을 sharing하기 때문에 하나의 뇌가 여러 번 순환하여 집중을 하는 형태의 추론을 함

ALBERT와 같은 언어모델 학습이 어려운 이유

1. 대용량 학습 데이터 수집 및 정제 필요
2. 학습 알고리즘 구현 필요
3. 학습을 위한 하드웨어 필요

공개된 알고리즘과 한국어 언어모델

해외 IT 기업들의 최신 알고리즘 공개 및 모델 제공

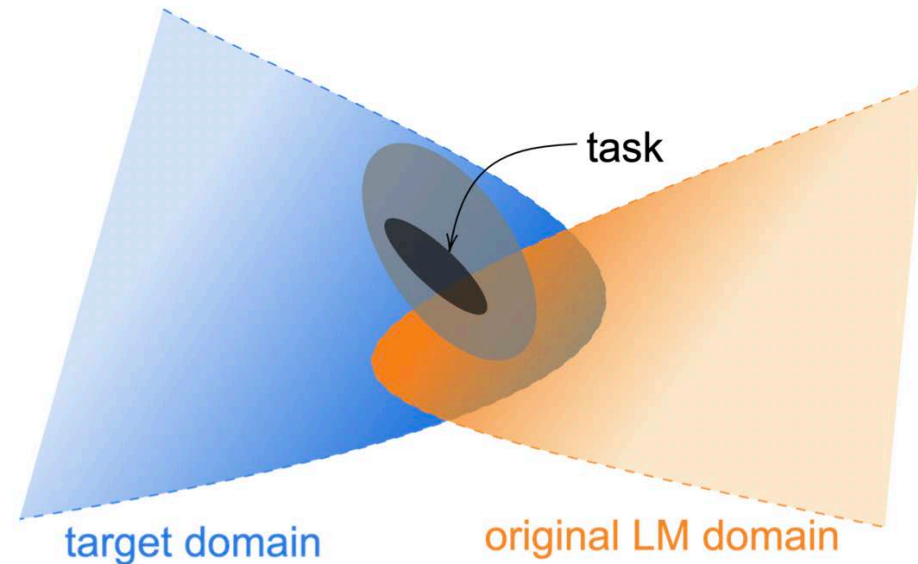
- Google BERT
- Facebook RoBERTa
- OpenAI GPT

한국어 언어모델도 공개 및 제공 중

- ETRI KorBERT
- SKT KoBERT
- ...

도메인의 특수성 고려 필요

다른 도메인에 미세조정(fine-tuning) 시
데이터 분포의 차이로 인한 성능 저하 발생



도메인 전문 용어가 포함된 데이터에 대한 학습 필요

“ 지난 프로젝트에서 여신 담당자와 함께 프로토타입 모형을 개발했다. ”

일반



어떤 **사람**과 함께
시제품을 만들었다는건가???

금융



대출 관련 서비스를
만들었구나!!!

도메인 전문 용어가 포함된 데이터에 대한 학습 필요

일반적인 도서로 영어를 학습하여 실력이 늘었다 할지라도
영문으로 된 경제 전문서적을 쉽게 이해하기 어려움



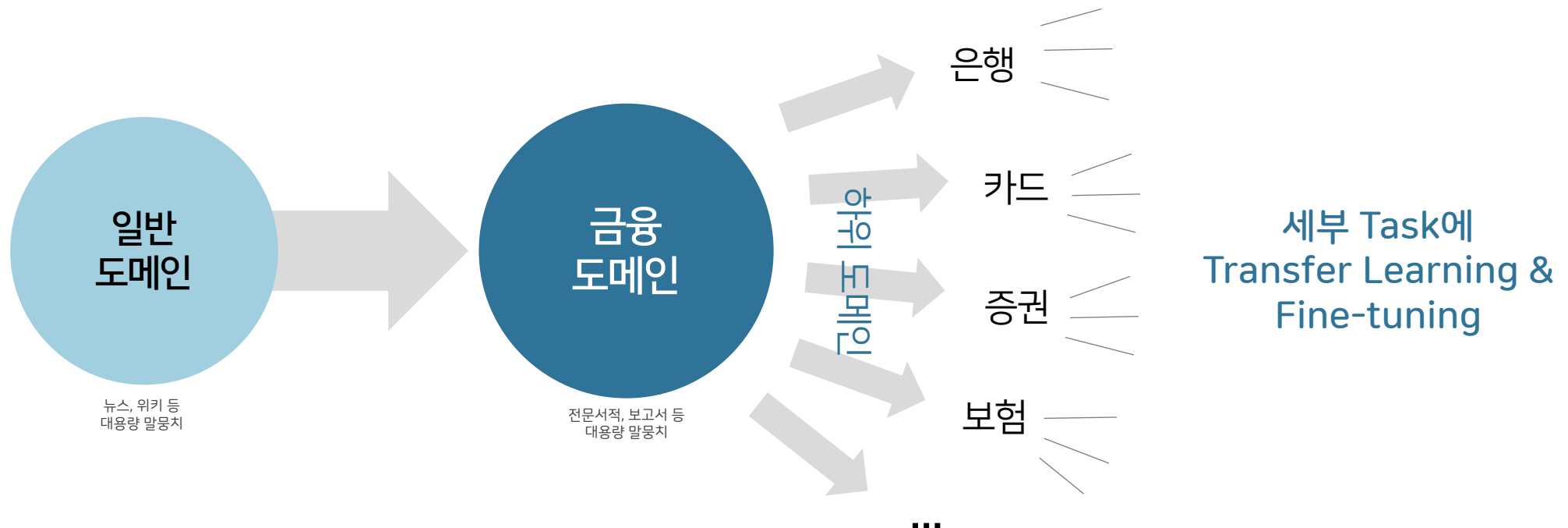
금융 도메인에 특화된 언어모델 KB-ALBERT?

Domain Adaptation*

일반화된 성능을 유지하면서도 도메인의 특수성이 고려된 언어모델 학습 필요

Pretraining phase를 여러 단계로 나누어 언어모델 학습전략을 구성

도메인 데이터를 추가해가는 Continual learning** 방식의 사전학습 + 전이학습 수행



Domain Adaptation을 위한 금융 특화 사전 개발

어근과 어미를 먼저 분리한 후 Wordpiece 사전과 토큰나이저 개발
사전은 일반 사전 대비 금융 관련 용어가 추가된 형태 (아래 결과 예시)

- 주요 금융기관명 등 포함
 - ✓ FOMC, JP모건, ...
- 증권 관련 주요 지수 등 포함
 - ✓ HSCEI, DSR, PBR, ...
- 기타 KB에서 자주 사용하는 단어 등 포함
 - ✓ 리브, 보금자리론, ...

금융 특화 사전의 토큰나이저 예시 (1)

원 문장

올 상반기 시중은행이 판매한 **방카슈랑스**는 지난해 같은 기간보다 30%나 증가한 것으로 나타났다.

일반 토큰나이저 tokenizing

'올', '상반기', '시중', '##은행', '##이', '판매', '##한', '방', '##카', '##슈', '##랑스', '##는', '지난', '##
해', '같', '##은', '기간', '##보', '##다', '30', '%', '나', '증가', '##한', '것', '##으로', '나타났', '##다', '.'

금융 토큰나이저 tokenizing

'올', '상반기', '시중', '은행', '이', '판매', '한', '방카슈랑스', '##는', '지난해', '같은', '기간', '##보다', '30',
'%', '나', '증가', '한', '것', '##으로', '나타났다', '.'

금융 특화 사전의 토크나이저 예시 (2)

원 문장

전문가들은 **PBR**을 끌어올릴 묘책은 배당 성향을 올리는 것이라고 입을 모았다.

일반 토크나이저 tokenizing

'전문가', '##들', '##은', '**PB**', '**##R**', '##을', '끌어올릴', '묘', '##책', '##은', '배당', '성향', '##을', '올리', '##는', '것', '##이', '##라', '##고', '입', '##을', '모았', '##다', '.'

금융 토크나이저 tokenizing

'전문가', '들은', '**PBR**', '##을', '끌어올릴', '묘', '##책', '은', '배당', '성향', '을', '올리는', '것', '##이라고', '입', '을', '모았다', '.'

KB-ALBERT의 활용 가능한 금융 분야

언어모델을 활용해 다양한 자연어처리 과제에 적용할 수 있으며,
이를 금융 분야에 응용하여 다양한 비즈니스 문제 해결 가능

- Market Intelligence

financial forecasting, market movement, ...

- Risk Management

reputation monitoring, compliance, fraud detection, ...

- Customer Service

chatbot, call center, sentiment analysis, customer relationship management, ...

- Knowledge Extraction

- Asset Management

영어 실력 측정을 위해 많이 보는 토익 시험

토익 시험은 영어 실력 측정을 위한 다양한 문제 유형들이 있음

READING TEST

In the Reading test, you will read a variety of texts and answer several different types of reading comprehension questions. The entire Reading test will last 75 minutes. There are three parts, and directions are given for each part. You are encouraged to answer as many questions as possible within the time allowed.

You must mark your answers on the separate answer sheet. Do not write your answers in your test book.

PART 5

Directions: A word or phrase is missing in each of the sentences below. Four answer choices are given below each sentence. Select the best answer to complete the sentence. Then mark the letter (A), (B), (C), or (D) on your answer sheet.

- | | |
|---|--|
| 101. Customer reviews indicate that many modern mobile devices are often unnecessarily ----- . (A) complication (B) complicates (C) complicate (D) complicated | 104. Among ----- recognized at the company awards ceremony were senior business analyst Natalie Obi and sales associate Peter Comeau. (A) who (B) whose (C) they (D) those |
| 102. Jamal Nawzad has received top performance reviews ----- he joined the sales department two years ago. (A) despite (B) except (C) since (D) during | 105. All clothing sold in Develyn's Boutique is made from natural materials and contains no ----- dyes. (A) immediate (B) synthetic (C) reasonable (D) assumed |
| 103. Gyeon Corporation's continuing education policy states that ----- learning new skills enhances creativity and focus. (A) regular (B) regularity (C) regulate (D) regularly | |

Questions 149-151 refer to the following article.

On Monday, Salinas Products, a large food distributor based in Mexico City, announced its plans to acquire the Pablo's restaurant chain. Pablo Benavidez, the chain's owner, had been considering holding an auction for ownership of the chain. He ultimately made the decision to sell to Salinas without seeking other offers. According to inside sources, Salinas has agreed to keep the restaurant's name as part of the deal. Mr. Benavidez started the business 40 years ago right after finishing school. He opened a small food stand in his hometown of Cancún. Following that, he opened restaurants in Puerto Vallarta and Veracruz, and there are now over 50 Pablo's restaurants nationwide.

- | | |
|--|---|
| 149. What is suggested about Mr. Benavidez? (A) He has hired Salinas Products to distribute his products. (B) He has agreed to sell his business to Salinas Products. (C) He has recently been hired as an employee of a school. (D) He has been chosen to be the new president of Salinas Products. | 151. What is indicated about the Pablo's restaurant chain? (A) It was recently sold in an auction. (B) It will soon change its name. (C) It was founded 40 years ago. (D) It operates in several countries. |
| 150. According to the article, where is Mr. Benavidez from? (A) Cancún (B) Veracruz (C) Mexico City (D) Puerto Vallarta | |

다른 유형의 토익 문제를 바로 풀 수 있을까?

Questions 149-151 refer to the following article.

기본 영어 실력이 있는 사람들은 높은 점수를 받을 수 있겠지만
진짜 고득점을 위해서는 예제 문제들을 풀어보는 것이 좋음

distributor based in Mexico City, announced its plans to acquire the Pablo's restaurant chain. Mr. Benavidez had been the owner of the chain. He ultimately made the decision to sell to Salinas without seeking other offers. According to inside sources, Salinas has agreed to keep the restaurant's name as part of the deal. Mr. Benavidez started the business 40 years ago right after finishing school. He opened a small food stand in his hometown of Cancún. Following that, he opened restaurants in Puerto Vallarta and Veracruz, and there are now over 50 Pablo's restaurants nationwide.

149. What is suggested about Mr. Benavidez?

- (A) He has hired Salinas Products to
- (B) He has hired Salinas Products to
- (C) He has been chosen to be the new president of Salinas Products.
- (D) He has been chosen to be the new president of Salinas Products.

제시된 본문의
문서 유형은?

150. According to the article, where is

- (A) Cancún
- (B) Veracruz
- (C) Mexico City
- (D) Puerto Vallarta

제시된 본문을 참조,
질문의 대답은?

151. What is indicated about the Pablo's

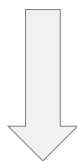
- (A) It is a family business.
- (B) It is a family business.
- (C) It is a family business.
- (D) It operates in several countries.

제시된 본문의
억양은? 감성은?

전통적인 기계학습 방식

과거에는 하나의 문제(task)마다 문제의 데이터를 활용해
각각의 학습 모델을 학습하는 방식으로 문제 해결

제시된 본문의
문서 유형은?



학습 모델1

제시된 본문의
억양은? 감성은?



학습 모델2

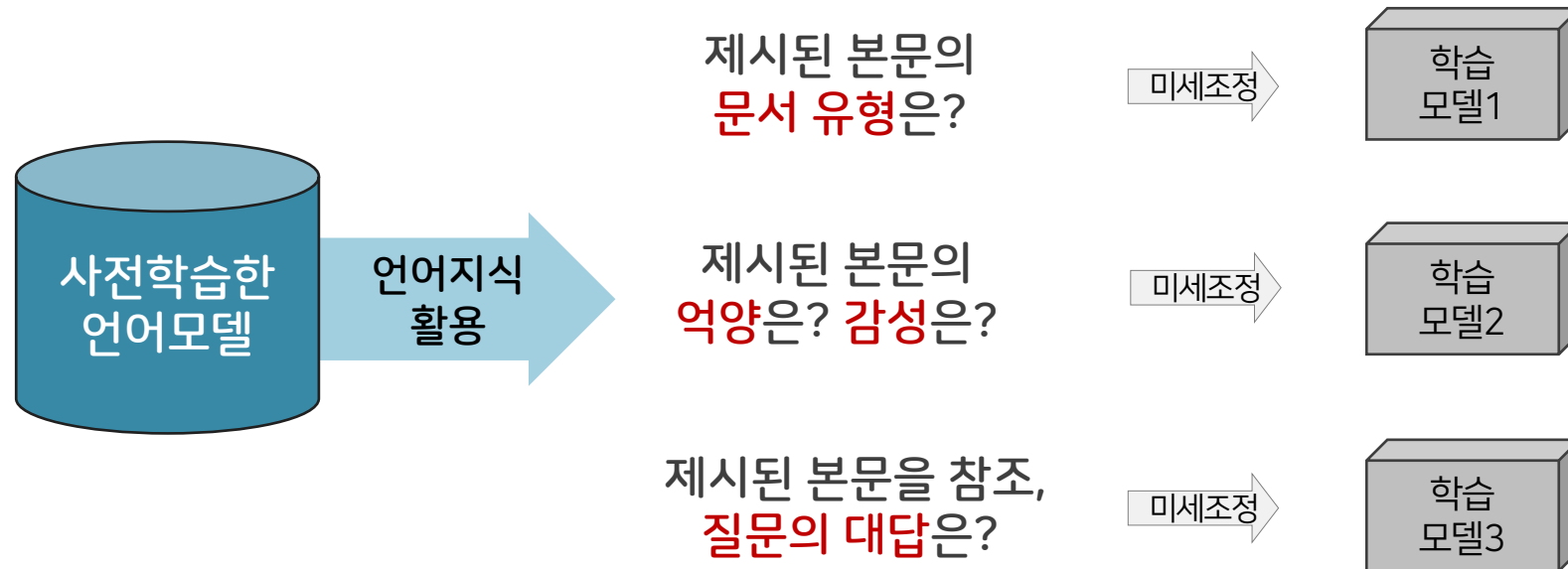
제시된 본문을 참조,
질문의 대답은?



학습 모델3

최근 사전학습한 언어모델을 통한 언어지식 활용

사전학습된 언어모델의 언어지식을 활용하여 각 문제(task)에 학습하면
더 적은 데이터를 이용하고도, 더 빠르게 학습하고, 더 좋은 성능이 나타남



Transfer Learning

사전학습된 언어모델(Pretrained language model)의 지식을 전이하는 학습 방법

언어 내부적으로 공유되는 특징이 있다는 가정에서 출발

cf. 미세조정 (Fine-tuning) : 사전학습된 언어모델의 weight 조정

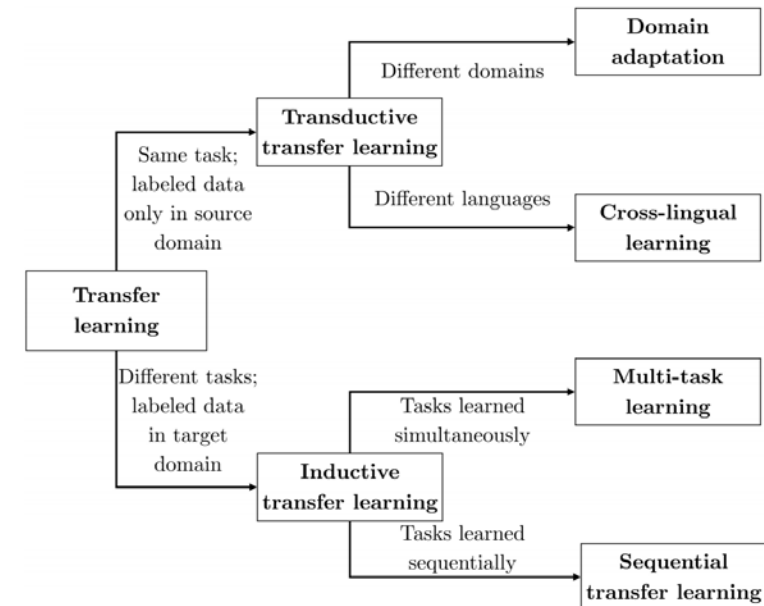
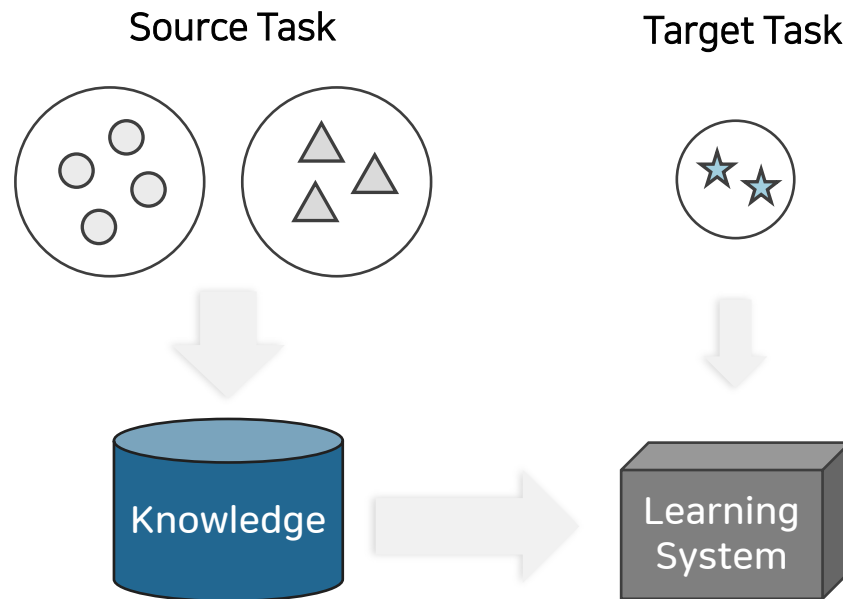


FIGURE 3.3: A taxonomy for transfer learning for NLP.

(Ruder, 2019)

사전학습된 언어모델 활용의 장애물

Practical Challenges

- 사전학습된 언어모델을 활용한 fine-tuning과 모델 배포에는 고성능 자원 필요
- “연구 > 개발 > 배포” 과정에서 연속성 확보 및 협업의 어려움으로 기술 부채 증가
- 다양한 Downstream task에서 빠르게 fine-tuning하고 테스트하기 쉽지 않음

Hugging Face's Transformers

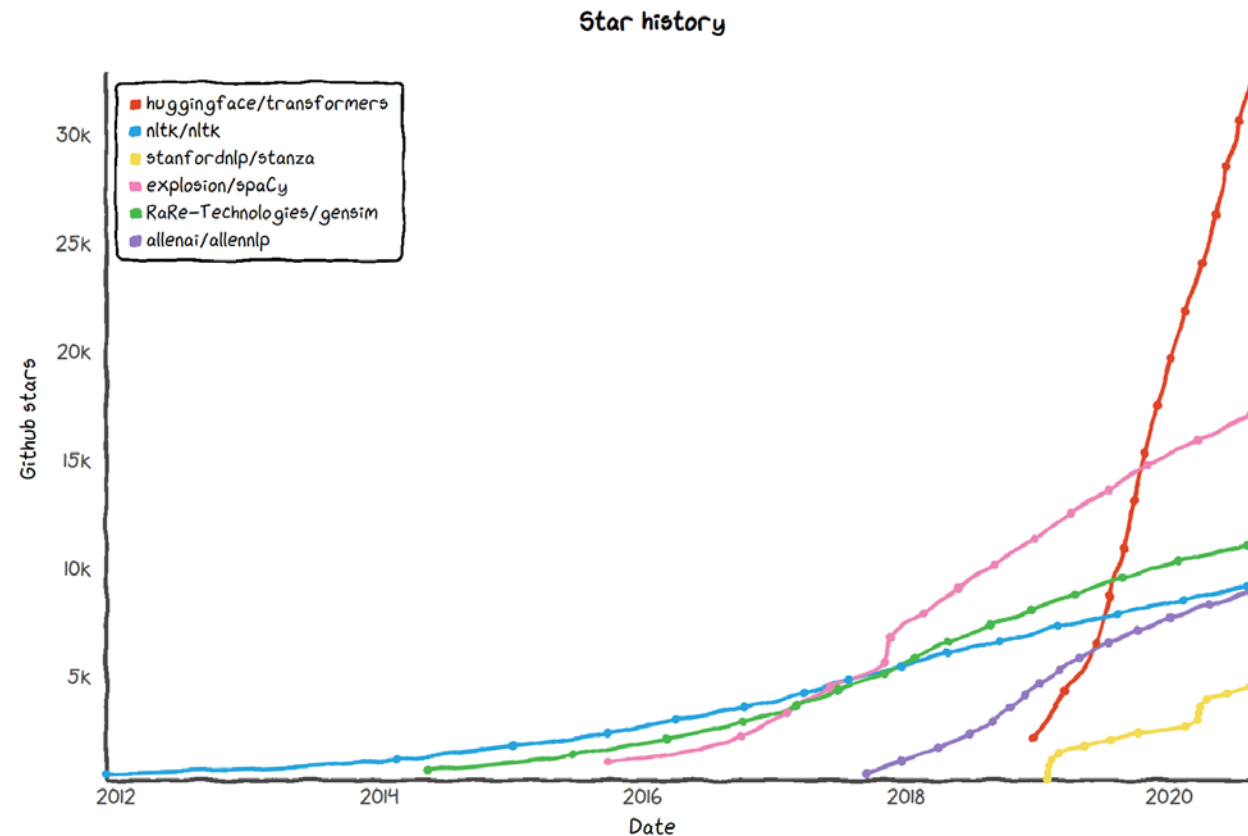


HUGGING FACE

On a mission to solve NLP,
one commit at a time.

Rising Star in NLP field

Hugging Face's Transformers는 최근 가장 인기 있는 NLP 라이브러리 중 하나



Features of Transformers



Design for Everyone

- Researchers, Practitioners, Engineers, Educators, ...

Easy and Fast to use

- SOTA NLU, NLG models with High performance

Incorporate various ML tools

- fine-tuning and serving models with Simple functionality

Why Transformers?

1. 간결하고 사용성이 좋은 API
2. PyTorch와 TensorFlow 간 높은 호환성
3. Deployment를 위한 다양한 API
4. Community

1) 간결하고 사용성이 좋은 API

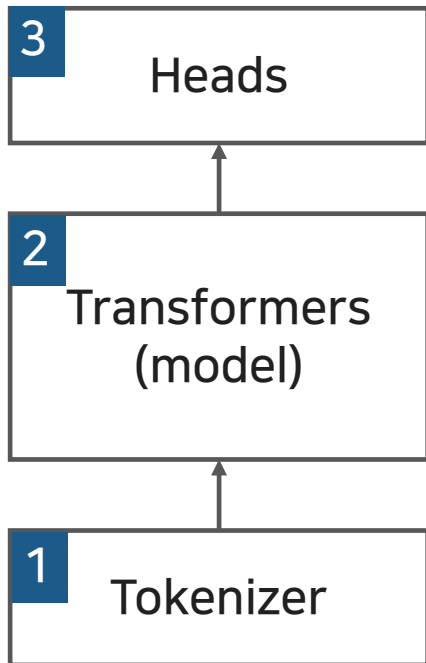
3줄의 코드 입력만으로 최신의 트랜스포머 모델을 불러와 사용 가능
불러온 토큰라이저와 모델을 이용해 손쉽게 fine-tuning 가능

```
from transformers import AutoTokenizer, AutoModel

tokenizer = AutoTokenizer.from_pretrained('namespace/pretrained_model')
model = AutoModel.from_pretrained('namespace/pretrained_model')
```

3개의 블록을 통한 손쉬운 Fine-tuning

Tokenizer, Transformers, Heads class를 조합하면
다양한 목적의 세부 task에 쉽게 fine-tuning하고 사용/배포 가능



Contextual embedded vector
→ Task-specific output vector

Sparse indices → Contextual embedded vector

Raw text → Sparse indices

1 Tokenizer

주기능은 Raw text를 Sparse index encoding 형태로 변환시키는 것

1. 입력된 텍스트에 대해 tokenize한 후
2. 사전학습된 vocab을 이용해 각 token과 매핑되는 index 값으로 변환
3. 변환된 index를 리스트로 반환

```
tokenizer = AutoTokenizer.from_pretrained('bert-base-multilingual-cased')
raw_text = '토크나이저의 결과입니다.'
input_ids = tokenizer(raw_text)['input_ids']
print(input_ids)
# [101, 9873, 20308, 16439, 10739, 48387, 85533, 58303, 48345, 119, 102]
```


2 Transformers Models

Transformers를 통해 공식 지원되는 transformer model 종류는 현재 24개

| Transformer model types | Example |
|-------------------------|------------------------------|
| Autoencoding model | BERT, RoBERTa, ... |
| Autoregressive model | GPT, GPT-2, XLNet, ... |
| Seq-to-Seq model | BART, T5, MarianMT, ... |
| Long-sequence model | Reformer, Longformer, ... |
| Efficient model | ALBERT, Electra, DistillBERT |

2 Transformers Models

주기능은 Sparse index encoding을 contextual embedding 형태로 변환

1. 각 index에 해당하는 embedding vector로 매핑하여 변환한 후
2. Transformer model에 inference한 후
3. Output으로 contextual embedded vector 반환

```
tokenizer = AutoTokenizer.from_pretrained('bert-base-multilingual-cased')
model = AutoModel.from_pretrained('bert-base-multilingual-cased')

inputs = tokenizer('토큰라이저의 결과입니다.', return_tensors='pt')
outputs = model(**inputs)[0]
print(outputs)
# tensor([[ -3.7025, -6.0605, -8.3605, -8.0640, -8.6640, -9.1799, -4.2138, -6.8612,
-7.5265, -7.7255, -3.2571, -8.4079, -5.8422, -7.5061, -6.9417, -1.3916, -8.3326, -
4.2127]], ... )
```

3 Heads

Transformer model 위에 fine-tuning을 위해 쌓는 output layer를 지칭

각 transformer model은 task 유형에 따라 다양한 head 제공

- Head 내부적으로는 사전학습한 transformer model을 불러와 output layer와 fine-tuning
- ex) XXXForSequenceClassification, XXXForTokenClassification

필요에 따라 기본 제공되는 Head 외에 Customized Head를 개발하여 사용할 수 있음

| Name | Input | Heads | | Ex. Datasets |
|-------------------------|------------------------|-----------------------------|---------------------------------------|-----------------------------|
| | | Output | Tasks | |
| Language Modeling | $x_{1:n-1}$ | $x_n \in \mathcal{V}$ | Generation | WikiText-103 |
| Sequence Classification | $x_{1:N}$ | $y \in \mathcal{C}$ | Classification, Sentiment Analysis | GLUE, SST, MNLI |
| Question Answering | $x_{1:M}, x_{M:N}$ | $y \text{ span } [1 : N]$ | QA, Reading Comprehension | SQuAD, Natural Questions |
| Token Classification | $x_{1:N}$ | $y_{1:N} \in \mathcal{C}^N$ | NER, Tagging | OntoNotes, WNUT |
| Multiple Choice | $x_{1:N}, \mathcal{X}$ | $y \in \mathcal{X}$ | Text Selection | SWAG, ARC |
| Masked LM | $x_{1:N \setminus n}$ | $x_n \in \mathcal{V}$ | Pretraining | Wikitext, C4 |
| Conditional Generation | $x_{1:N}$ | $y_{1:M} \in \mathcal{V}^M$ | Translation, Summarization | WMT, IWSLT, CNN/DM, XSum |

(Wolf et al., 2020)

[별첨] Customized Head

Transformers의 *PretrainedModel* 클래스를 상속 받아 API 구조를 따라 자신의 목적에 맞는 Customized Head 개발 가능

```
class XXXForCustomClassification(transformers.PreTrainedModel):
    def __init__(self, config):
        super().__init__(config)
        self.transformer_model = XXXModel(config)

        # customized output layer 정의
        self.custom_layer = nn.Linear(config.hidden_size, config.output_size)
        self.init_weights()

    def forward(self, **kwargs):
        contextual_embedded_vector = self.transformer_model(**kwargs)[1]
        logits = self.custom_layer(contextual_embedded_vector)

        # loss 함께 정의
        loss = loss_function(logits, labels)
        return ((loss,) + contextual_embedded_vector)
```

2) PyTorch와 TensorFlow 간 높은 호환성

Transformers에서 제공하는 모든 모델 API는 PyTorch와 TensorFlow2 간의 상호 호환이 되어 프레임워크 간의 변환이 자유롭게 될 수 있도록 설계

한 쪽 프레임워크에서 training했더라도 다른 프레임워크에서 inference로 사용 가능

```
tokenizer = BertTokenizer.from_pretrained( ' bert-base-multilingual-cased ' )
model = TFBertModel.from_pretrained( ' bert-base-multilingual-cased ' ,
                                     from_pt=True)

inputs = tokenizer( ' 토큰나이저의 결과입니다. ' , return_tensors= ' tf ' )
outputs = model(inputs)[0]
print(outputs)
# tensor([[ -3.7025, -6.0605, -8.3605, -8.0640, -8.6640, -9.1799, -4.2138, -6.8612,
-7.5265, -7.7255, -3.2571, -8.4079, -5.8422, -7.5061, -6.9417, -1.3916, -8.3326, -
4.2127]], ... )
```

3) Deployment 환경

Hugging Face는 transformer 모델의 production 환경에서의 사용을 많이 고려

- Serving in production with TorchServing
- Inference Optimization with ONNX, JAX/XLA and TVM teams
- Edge device (iOS) with CoreML
- Hyperparameter Optimization with Optuna, Ray Tune

그 밖에도 PyTorch lightning, Weight & Biases와 같은 다른 머신러닝 툴, 라이브러리들과의 연계도 넓혀가고 있음

(직접 할 수 있지만 이런 건 좀 api에 맞게 제공해줬으면 좋겠다 싶은 기능들은 이미 깃헙에 관련 PR이 올라와 있거나 개발 논의 중인 경우가 상당수임)

4) Community

가장 큰 장점은 Community Model hub을 통해 전세계 사용자들이 직접 학습한 2,000+ 개의 모델들이 공유 중 (Google, Facebook 등도 공식 모델 공유 중)

수백명의 contributor들과 다양한 organization들의 참여를 통해 다양한 case study 발생



HUGGING FACE

[Back to home](#)

All Models and checkpoints

Also check out our list of [Community contributors](#) 🏆 and [Organizations](#) 🌐.

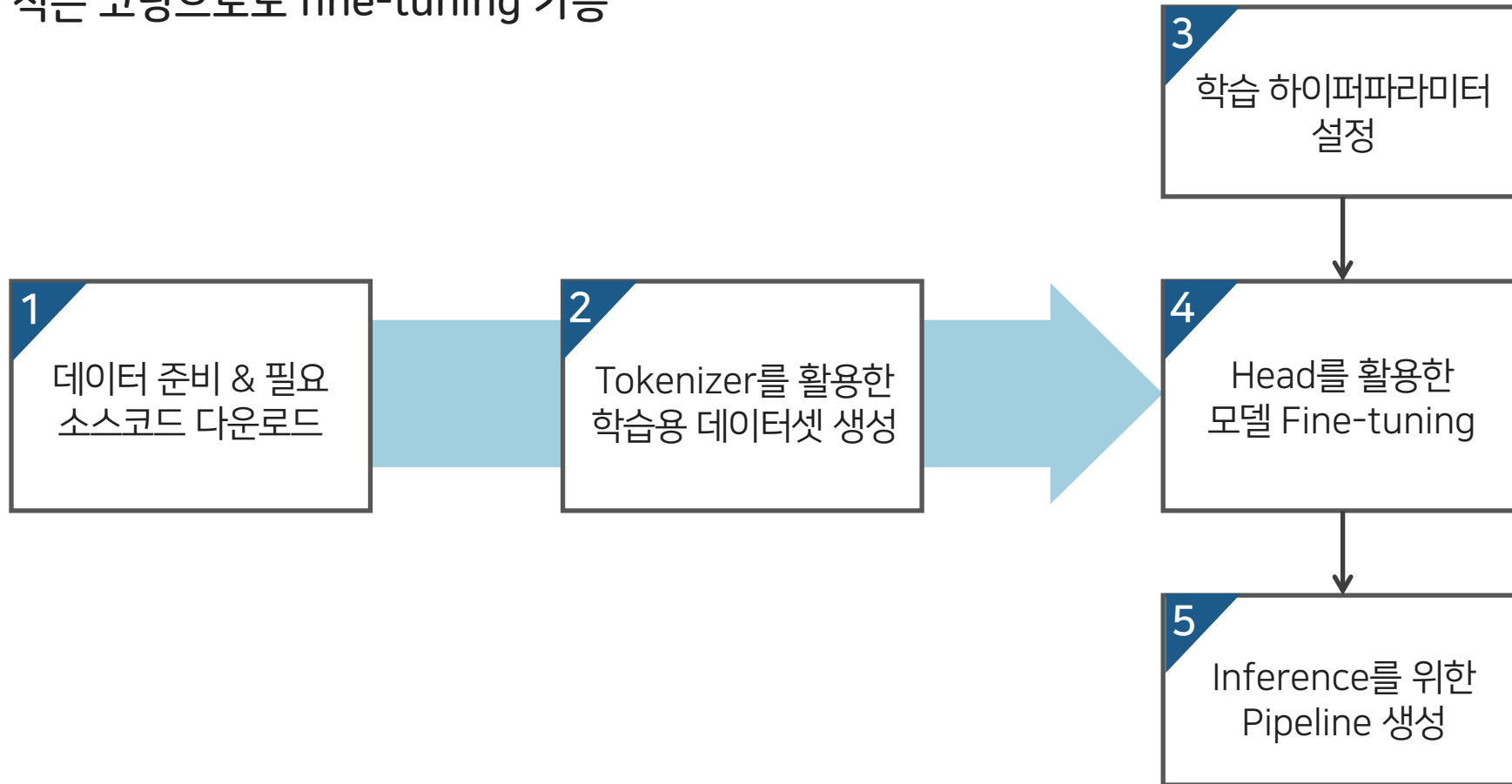
Tags: All ▼

Sort: Most downloads ▼

Fine-tuning KB-ALBERT with Transformers

Fine-tuning Process with Transformers

Transformers의 API를 사용하면 PyTorch/TensorFlow Naïve Code programming 대비 더 적은 코딩으로도 fine-tuning 가능



네이버 영화리뷰 감성분석

학습 데이터 15만개, 테스트 데이터 5만개의 네이버 영화 리뷰로 구성된 데이터
긍정적인 리뷰에는 label이 '1', 부정적인 리뷰에는 label이 '0'으로 표시

| id | document | label |
|----------|---|-------|
| 9976970 | 아 더빙.. 진짜 짜증나네요 목소리 | 0 |
| 3819312 | 흠...포스터보고 초딩영화줄....오버연기조차 가볍지 않구나 | 1 |
| 10265843 | 너무재밌었다그래서보는것을추천한다 | 0 |
| 9045019 | 교도소 이야기구먼 ..솔직히 재미는 없다..평점 조정 | 0 |
| 6483659 | 사이몬페그의 익살스런 연기가 돋보였던 영화!스파이더맨에서 늙어보이기만 했던 커스틴 ... | 1 |
| 5403919 | 막 걸음마 떼 3세부터 초등학교 1학년생인 8살용영화.ㅋㅋㅋ...별반개도 아까움. | 0 |
| 7797314 | 원작의 긴장감을 제대로 살려내지못했다. | 0 |

Tokenizer를 활용한 학습용 데이터셋 생성

```
from tokenization_kbalbert import KbAlbertTokenizer

tokenizer = KbAlbertTokenizer.from_pretrained('kb-albert-model-dir')

train_sparse_indices = tokenizer(train_texts)
train_dataset = NSMCDataset(train_sparse_indices, train_labels)
```

학습 하이퍼파라미터 설정

```
from transformers import TrainingArguments

training_args = TrainingArguments(
    num_train_epochs=2.0,
    per_device_train_batch_size=16,
    warmup_steps=500
)
```

Head를 활용한 모델 Fine-tuning

```
from transformers import AlbertForSequenceClassification, Trainer

model = AlbertForSequenceClassification.from_pretrained('kb-albert-char')

trainer = Trainer(model=model,
                  args=training_args,
                  train_dataset=train_dataset)

trainer.train()
```

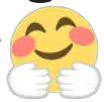
Inference를 위한 Pipeline 생성

```
from transformers import pipeline

nsmc_classifier = pipeline('sentiment-analysis',
                           model=model, tokenizer=tokenizer, framework='pt')

review = "역시 재밌네 ㅋㅋ 최고"
results = nsmc_classifier(review)
print(results)
# { 'label': 'positive', 'score': 0.9901639223098755 }
```

Transformers 사용시 주의할 점!

- 버전 업그레이드에 따른 변화가 크다!
- 엔지니어들의 빠른 업데이트로 같은 API라도 기능이 점점 좋아지면서 조금씩 기존에 작성한 개인 코드를 수정해야 할 수 있음 ㅎ  (ex. Trainer)
- Optimization을 위해서는 기본 API 구조에 Customization하는 것이 나을 수 있음
- 또는 일부 API만 사용하는 것이 좋을 수도 있음 (ex. 토큰나이저만 사용한다든지)
- Transformers 팀은 Fine-tuning과 Deployment에 집중하고 있어
오리지널 transformer 모델의 Pretraining이 필요하다면 직접 개발 필요
(PR을 환영한다고 하고 있음. 예를 들면 허깅페이스 Transformers를 이용한 ALBERT의 사전학습이 하고 싶다면 Sentence Order Prediction Loss 적용이나 Data collator 쪽 데이터 생성 등 추가 개발이 필요)

결론... 소소한 생각들...

Hugging Face의 Transformers를 사용하면서 덕분에

- 비즈니스 문제와 데이터에 좀 더 집중할 수 있었음
- 실제 과제에서 여러가지 Optimization 방법들을 쉽고 빠르게 적용해 볼 수 있었음

시간과 기회가 허락한다면 Transformers에 한국어 관련 PR도... 🙌

발표자료와 예제 소스코드 Link

<https://github.com/sackoh/pycon-korea-2020-kb-albert>

Thank you

for your time and consideration