LangCon 2019

트랜스퍼 러닝과 텍스트 문서 분류

고재선

고재선

- 통신공학 전공, 법학전문대학원 졸업
- 대학원(법학박사) 과정
- 2014년부터 변호사로 근무
- 관심분야 : 자연어처리, 디지털포렌식, 핀테크

데이터 부족할 때는,

트랜스퍼 러닝을 한 번 고려해보자.

DS 033

파이썬을 활용한 딥러닝 전이학습

디파니안 사르카르, 라그허브 발리, 타모그나 고시 지음
송영숙, 심상진, 한수미, 고재선 옮김

위키북스

파이썬을 활용한

딥러닝 전이학습

Transfer Learning으로 빠르고 손쉽게 구축하는 고급 딥러닝 모델

디파니안 사르카르, 라그허브 발리, 타모그나 고시 지음
/
송영숙, 심상진, 한수미, 고재선 옮김

DS 데이터 사이언스 시리즈 .033

위키북스

트랜스퍼 러닝(Transfer Learning)?

≒하나의 설정에서 배운 무엇인가를,

다른 설정에서도 일반화할 수 있도록 활용하는 환경*

*Deep Learning, 이안 굿펠로,요슈아 벤지오,에런 쿠빌 공저/류광 역

트랜스퍼 러닝(Transfer Learning)?

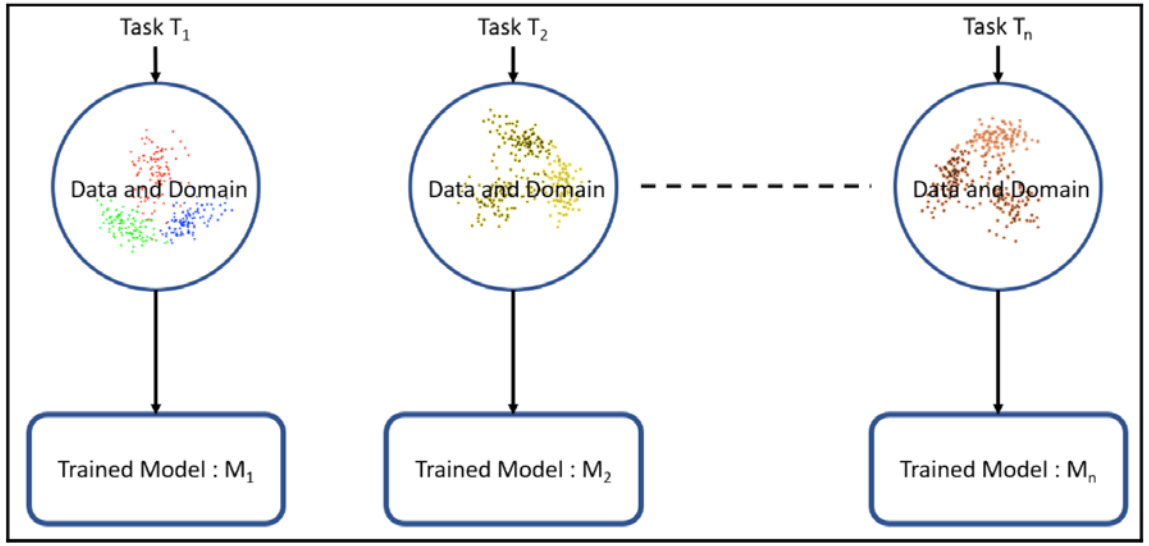
≒하나의 설정에서 배운 무엇인가를,

다른 설정에서도 일반화할 수 있도록 활용하는 환경*

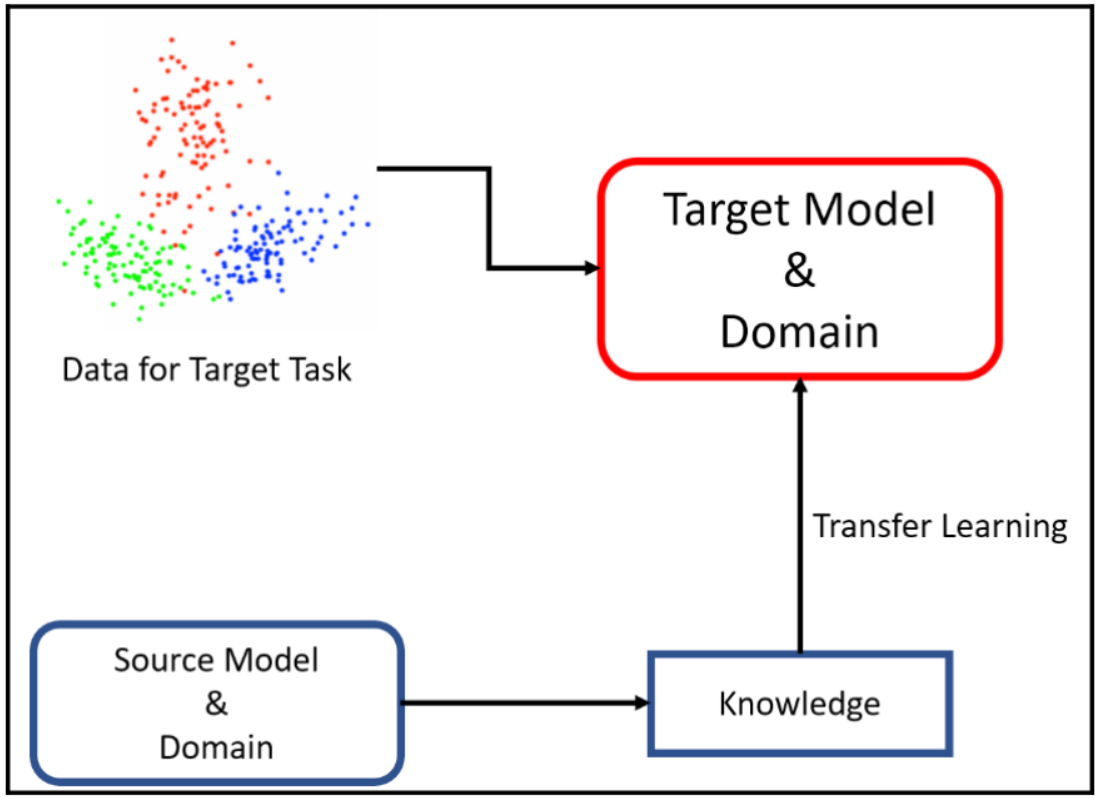
≒다른 분야의 학습 모델을 가져와 유사한 분야에서 적용하는 것

*Deep Learning, 이안 굿펠로,요슈아 벤지오,에런 쿠빌 공저/류광 역

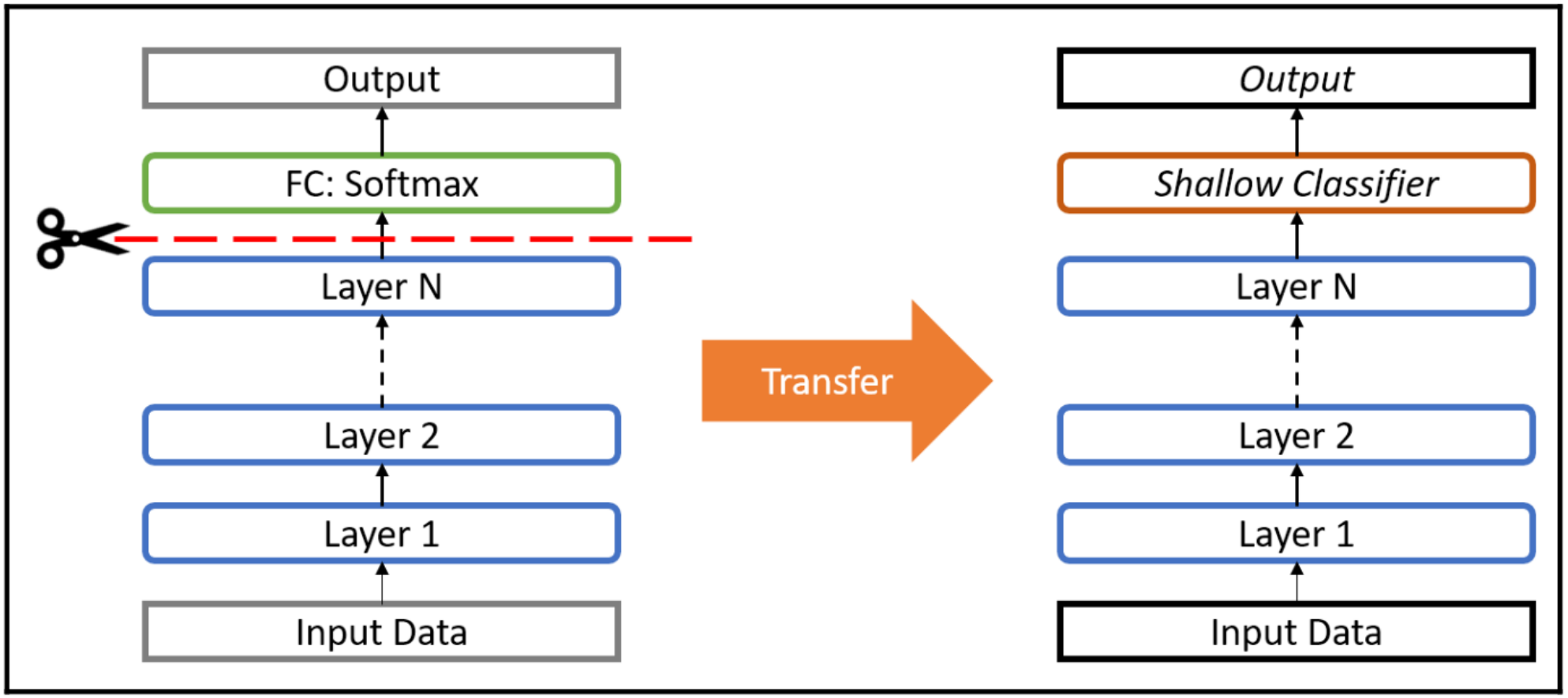
기존 머신 러닝



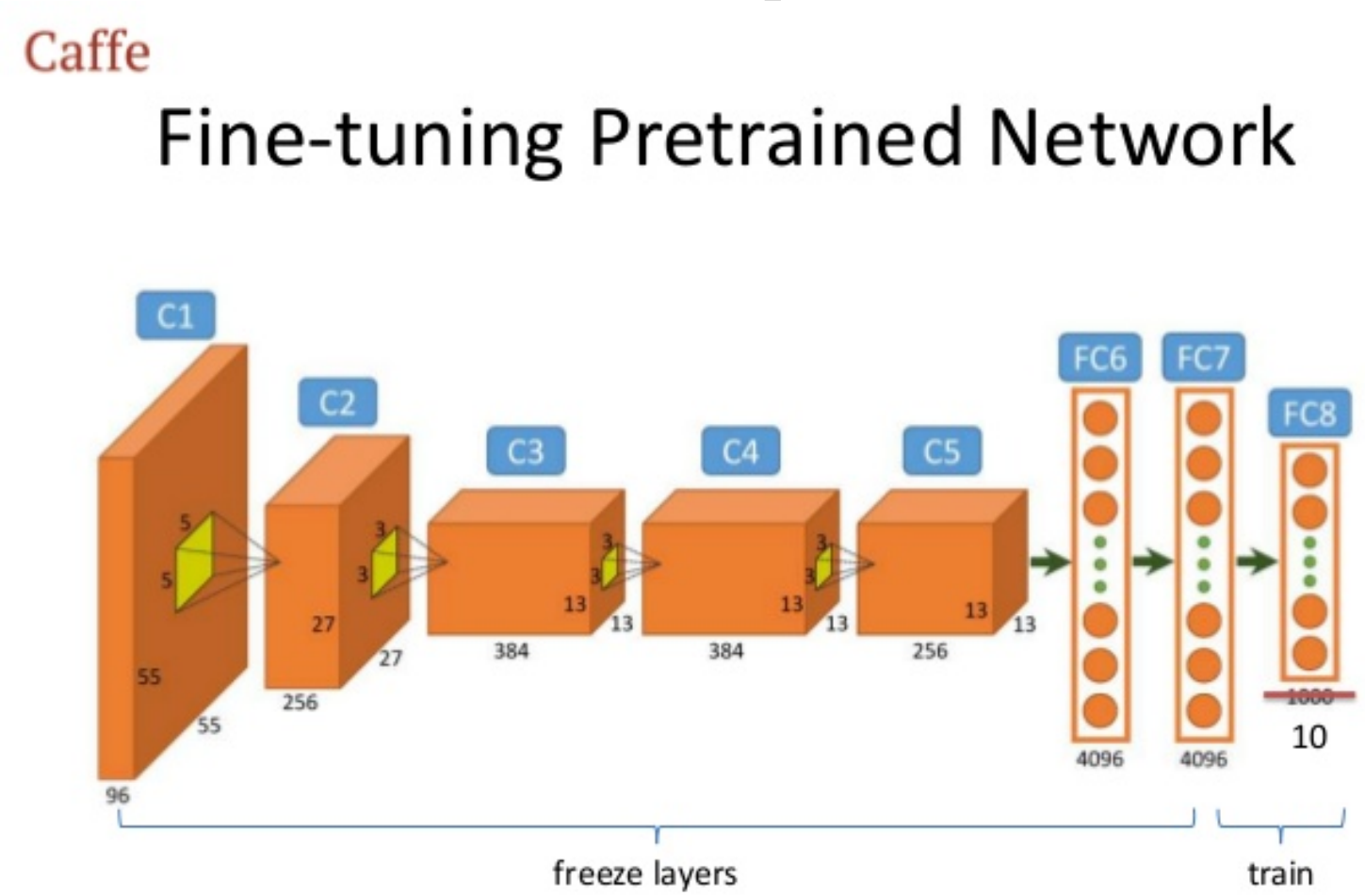
트랜스퍼 러닝



트랜스퍼 러닝(Feature-extraction)



트랜스퍼 러닝(Fine-tuning)



트랜스퍼 러닝을 사용하는 이유?

1. 베이스라인 성능 향상
2. 모델 개발/학습 시간 단축
3. 최종 성능 향상

영상(CV) 분야의 트랜스퍼 러닝?

대량의 이미지 데이터 셋으로

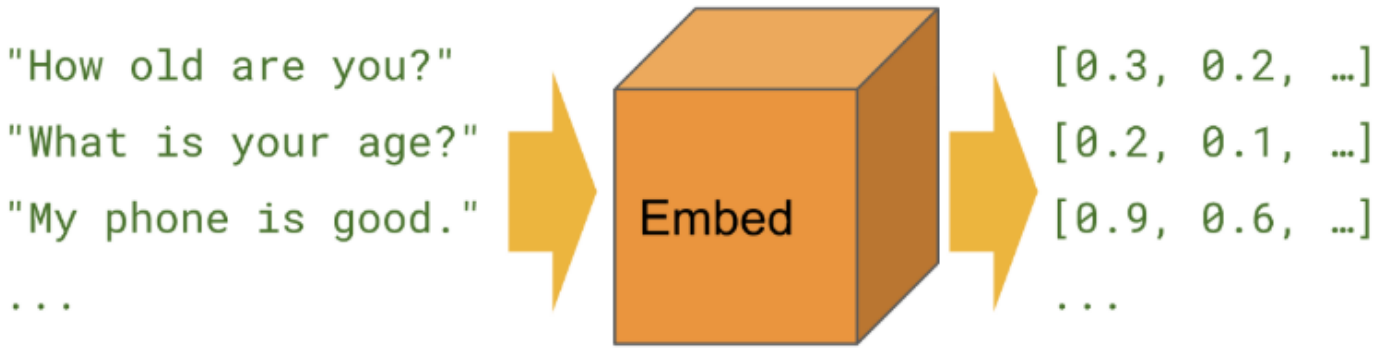
학습시킨 모델을 사용하여

구체적인 문제들을 해결

자연어 처리의 트랜스퍼 러닝은?

- 워드 임베딩을 중심으로 논의
- 최근 ELMO, BERT 등의 사전 학습 모델 등장

임베딩?



- 워드 임베딩 : 단어를 실수 벡터 값으로 맵핑시키는 것

- 어떻게 맵핑?

워드 임베딩 모델 : Word2vec, Glove

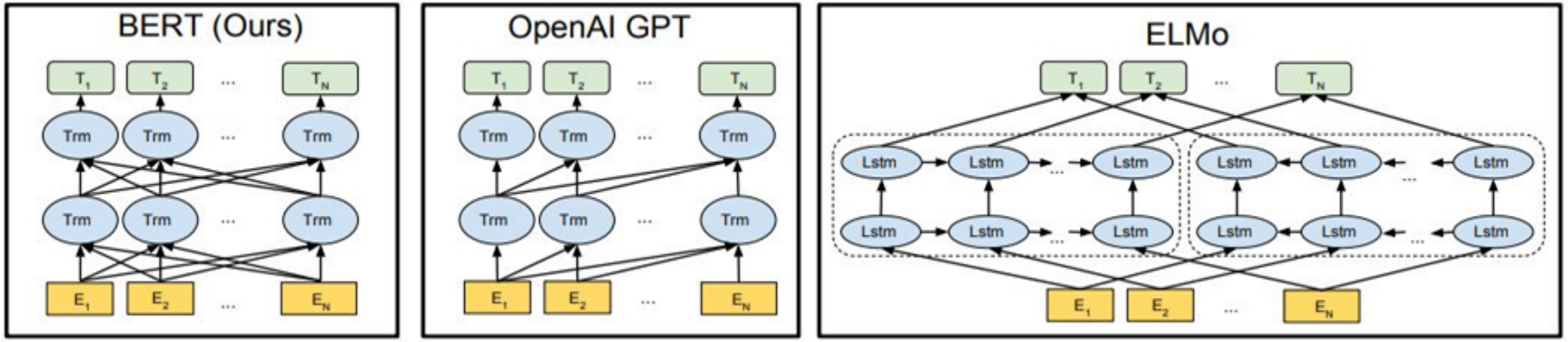
- Word2vec : 문장 내 단어들의 위치를 기반으로 학습
- Glove : 전체 단어들의 통계 정보(동시출현확률)를 사용

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

워드 임베딩 모델 : ELMO, BERT

- 문맥에 따라 같은 단어라도 다른 벡터로 표현 (Word2vec 에서의 다의어, 동음이의어 문제)

- 대량의 텍스트 데이터를 미리 학습하는 모델



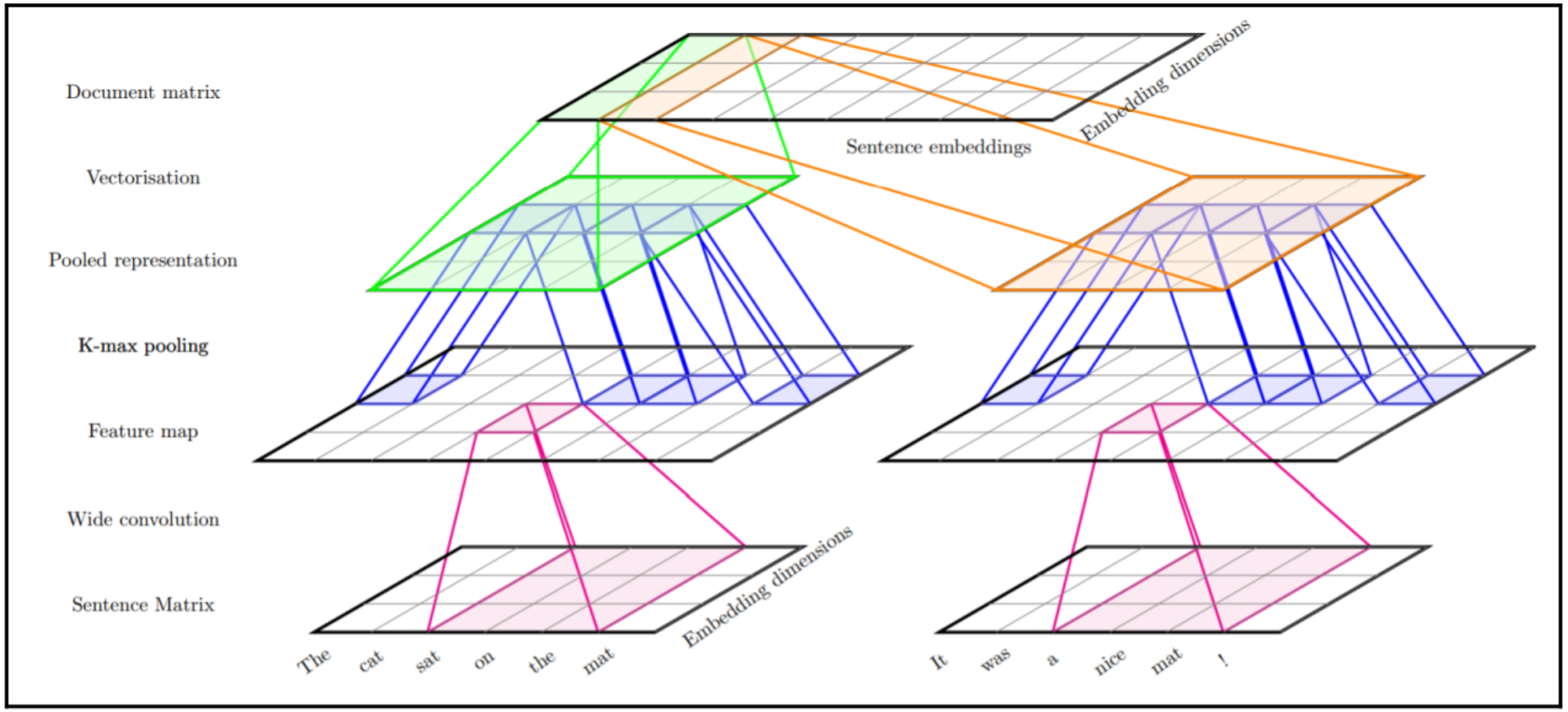
* Jacob Devlin, et all, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

CNN 문서 모델*

- 워드 임베딩 ->문장 임베딩 -> 문서 임베딩
- 인풋 레이어 : 워드 임베딩
- 문장과 문서의 길이가 다를 수 있으므로,
    - K-max 풀링 / 0으로 패딩

* Misha Denil, et all, Modelling, Visualising and Summarising Documents with a Single Convolutional Neural Network, 2014

CNN 문서 모델 *



* Misha Denil, et all, Modelling, Visualising and Summarising Documents with a Single Convolutional Neural Network, 2014

IMDB 영화 리뷰 – 긍정/부정 분류

- 트레이닝 데이터 25,000개, 테스트 데이터 25,000개
- 사전 학습된 Glove 벡터(Wikipedia 2014 + Gigaword 5)
- 약 83.7%

```
Epoch 00018: val_loss did not improve from 0.36422
Epoch 19/20
 - 25s - loss: 0.3785 - acc: 0.8316 - val_loss: 0.3753 - val_acc: 0.8304

Epoch 00019: val_loss did not improve from 0.36422
Epoch 20/20
 - 24s - loss: 0.3763 - acc: 0.8350 - val_loss: 0.3730 - val_acc: 0.8440

Epoch 00020: val_loss did not improve from 0.36422
[0.36754346494674683, 0.8375999972343445]
```

만약 IMDB 데이터가 1,250개만 있다면? (5%)

- 트레이닝 데이터 1,250개만 있다면 …
- 이 경우 트랜스퍼 러닝을 고려해 볼 수 있음
- 영화평과 유사한 상품 구매 평가!

아마존 제품 구매 평가 - 긍정/부정 분류

- 학습용 데이터 360만개, 테스트용 데이터 40만개

- 샘플 20만개 학습

```
Epoch 00031: val_loss improved from 0.17975 to 0.17930, saving model to /home/lfm
/TL/Hands-On-Transfer-Learning-with-Python/Chapter07/model/amazonreviews/model_06
.hdf5
Epoch 32/35
 - 134s - loss: 0.1570 - acc: 0.9419 - val_loss: 0.1869 - val_acc: 0.9322

Epoch 00032: val_loss did not improve from 0.17930
Epoch 33/35
 - 136s - loss: 0.1570 - acc: 0.9417 - val_loss: 0.1846 - val_acc: 0.9313

Epoch 00033: val_loss did not improve from 0.17930
Epoch 34/35
 - 133s - loss: 0.1559 - acc: 0.9423 - val_loss: 0.1876 - val_acc: 0.9297

Epoch 00034: val_loss did not improve from 0.17930
Epoch 35/35
 - 131s - loss: 0.1552 - acc: 0.9426 - val_loss: 0.1865 - val_acc: 0.9328

Epoch 00035: val_loss did not improve from 0.17930
(elmoenv) lfm@lfm-System-Product-Name:~/TL/Hands-On-Transfer-Learning-with-Python
/Chapter07$
```

아마존->IMDB 트랜스퍼 러닝

- 구매평 모델 로드 + 1,250개 데이터 학습
- 86.3%!

```
Train on 1237 samples, validate on 13 samples
Epoch 1/30
 - 2s - loss: 1.7599 - acc: 0.8294 - val_loss: 1.5267 - val_acc: 0.8462

Epoch 00001: val_loss improved from inf to 1.52668, saving model to
/home/lfm/TL/Hands-On-Transfer-Learning-with-Python/Chapter07/model/imdb/transfer_model_10.hdf5
Epoch 2/30
 - 1s - loss: 1.6181 - acc: 0.8367 - val_loss: 1.4488 - val_acc: 0.7692
```

```
Epoch 00030: val_loss improved from 0.48925 to 0.46712, saving model to
/home/lfm/TL/Hands-On-Transfer-Learning-with-Python/Chapter07/model/imdb/transfer_model_10.hdf5
[0.5902624861717224, 0.8636800016403198]
```

아마존->IMDB 트랜스퍼 러닝

- 구매평 모델 로드 + 25,000개 데이터 학습
- 87.3%!!

```
23636 words are updated out of 28681
Vocab Size = 28683  and the index of vocabulary words passed has 28681 words
Train on 23750 samples, validate on 1250 samples
Epoch 1/30
 - 14s - loss: 1.1472 - acc: 0.8482 - val_loss: 0.7576 - val_acc: 0.8592

Epoch 00001: val_loss improved from inf to 0.75759, saving model to
/home/lfm/TL/Hands-On-Transfer-Learning-with-Python/Chapter07/model/imdb/transfer_model_10.hdf5
Epoch 2/30
```

```
Epoch 00030: val_loss did not improve from 0.35825
[0.3611379730224609, 0.8738400040626526]
```

트랜스퍼 러닝 결과

IMBD (GLOVE)	AMAZON ->IMBD(5%)	AMAZON ->IMBD (100%)	SVM
83%	**86.3%**	**87.3%**	**83%**

학습에 필요한 데이터가 부족하거나,

성능 향상이 필요할 때,

트랜스퍼러닝 고려해볼 수도 있다. 끝.