Deep Learning paper review

Improving Language Understanding by Generative Pre-Training

인천대학교 컴퓨터공학부 DILAB 강병하

논문 소개

Improving Language Understanding by Generative Pre-Training

Alec Radford OpenAI alec@openai.com

Karthik Narasimhan OpenAI karthikn@openai.com

Tim Salimans OpenAI tim@openai.com

Ilya Sutskever OpenAI ilyasu@openai.com

- Open AI가 2018년 6월 발표
- GPT-1 논문 (Generative Pre-Training → GPT)
- 특정 task에만 특화된 모델이 아니라 광범위한 task를 처리할 수 있는 **다재다능**한 모델을 만들 수 없을까? 의 출발점
- 이후 GPT-2(2019)와 GPT-3(2020)가 공개

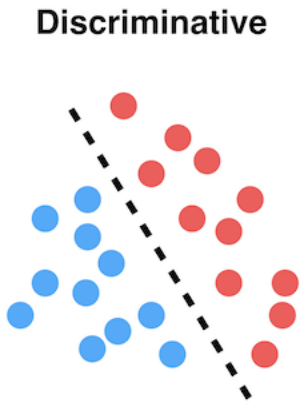
Background : 왜 Generative Model이 적합한가?

✓ Discriminative Model

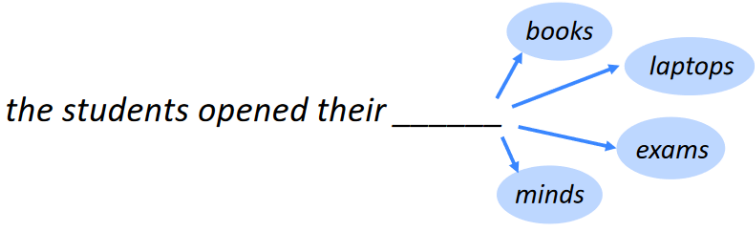
titanic dataset

PassengerId	Survived	Pclass	Name	Sex	Age
1	0	3	Braund, Mr. Owen Harris	male	22
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38
3	1	3	Heikkinen, Miss. Laina	female	26
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35
5	0	3	Allen, Mr. William Henry	male	35

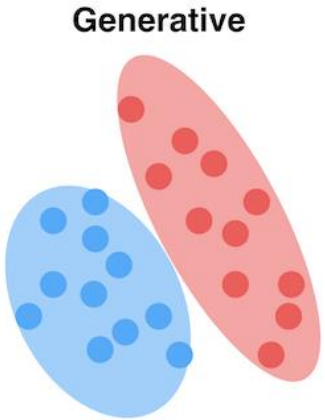
- 클래스간 차이에 주목(boundaries)
- 과적합 되기 쉬움
- Labeled data에 적합



✓ Generative Model



- 클래스별 분포에 주목
- Unlabeled data에 적합
- Data가 **충분히 많으면** 현실 분포와 매우 유사한 확률 분포 학습



GPT 등장 배경

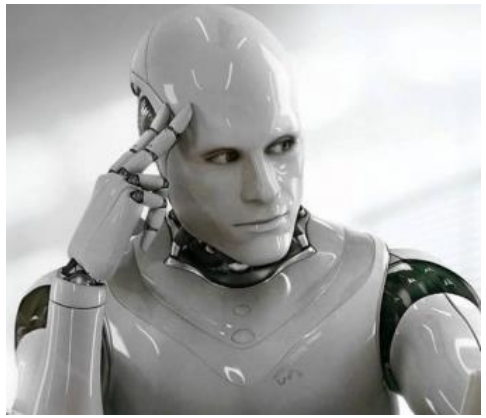
(기존)태스크별로 별도의 데이터셋으로 학습 → 태스크별 labeled data 필요 → 시간・비용↑

반면 디지털 세상에 **unlabeled data**는 **매우 풍부**

다양하고 방대한 raw text**로** **g**enerative **p**re-**t**raining → 태스크마다 개별적으로 fine-tuning

소규모의 labeled 데이터만 있으면 됨.

사전학습
(언어 이해)



미세조정
(태스크별 훈련)

- textual entailment
- question answering
- semantic similarity assessment
- document classification
- ⋮

Pre-train의 문제

Pre-trained **Word embedding**

- Word2Vec
- Glove

다양한 task에서의 성능 향상

(-) 단어 레벨 이상의 의미를 파악하기에는 어려움

✓ Unlabeled Text에서 **단어 레벨 이상**(구, 문장, 스토리)의 정보를 추출하자!

- 어떤 목적함수(optimization objective)를 사용해야 하는가?
- 학습된 표현을 transfer하는 가장 효과적인 방법은 무엇인가?

GPT의 기본 아이디어

목표 : 다양한 Task에 적용할 수 있는 **universal representation**을 학습 ← 언어 자체에 대한 이해

2단계

1. Unlabeled Text로 **언어 모델링** 학습 진행 with Transformer 디코더
2. Task별 labeled dataset으로 지도학습 진행 ← 각 태스크별 input transformations
    - Natural Language Inference
    - Semantic Similarity
    - Question Answering and Commonsense Reasoning
    - Classification

Unsupervised pre-training

일반적인 언어 모델링 objective 사용

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

L1의 likelihood를 **최대화** 하는 것이 목적 in **pre-training**

$\mathcal{U}$: 말뭉치 토큰(unlabeled)

k : Context window size

Θ : 파라미터

Unsupervised pre-training : Transformer Decoder

multi-layer(12개) Transformer 디코더 사용

토큰들의 context vector

$$h_0 = UW_e + W_p$$

토큰 임베딩 위치 임베딩

$$h_l = \texttt{transformer\_block}(h_{l-1}) \forall i \in [1, n]$$

$$P(u) = \texttt{softmax}(h_n W_e^T)$$

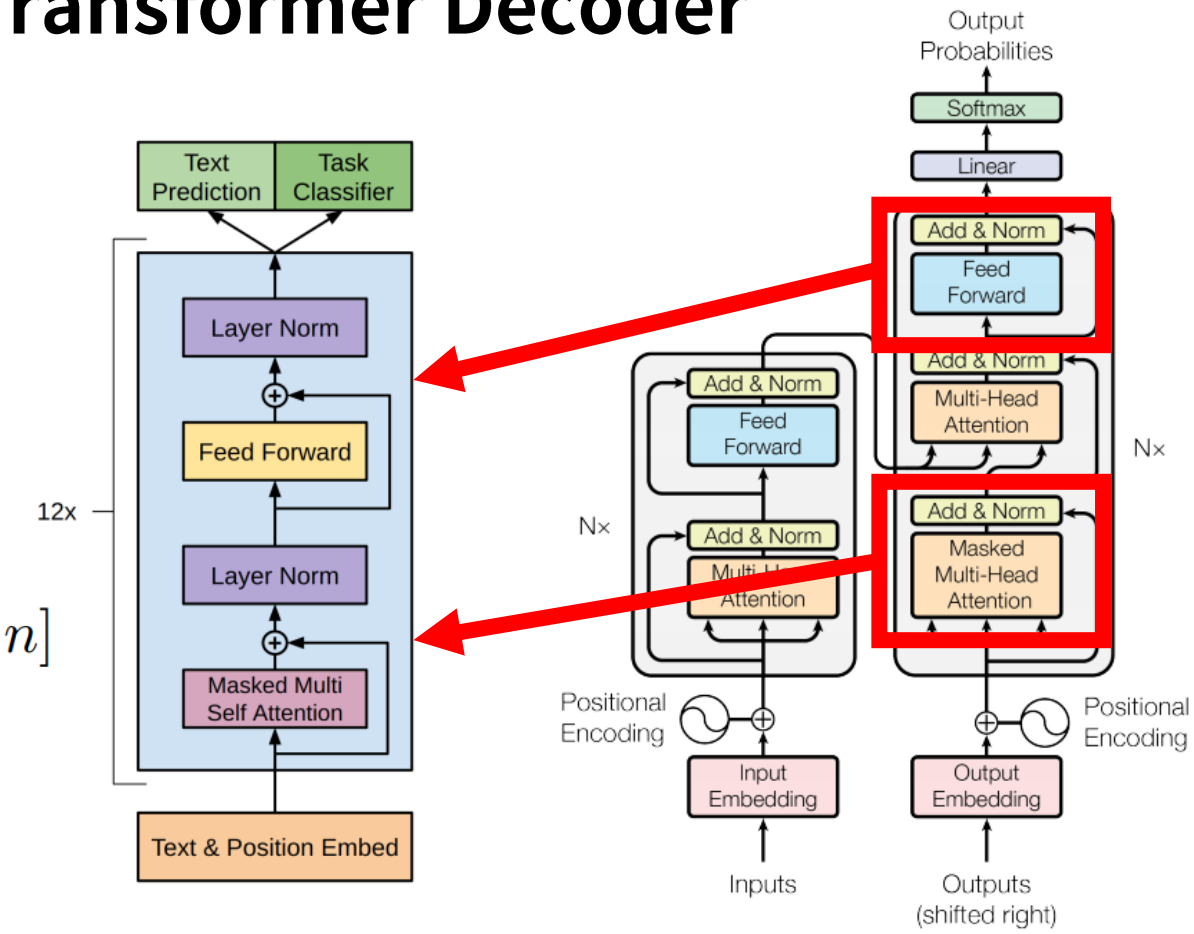


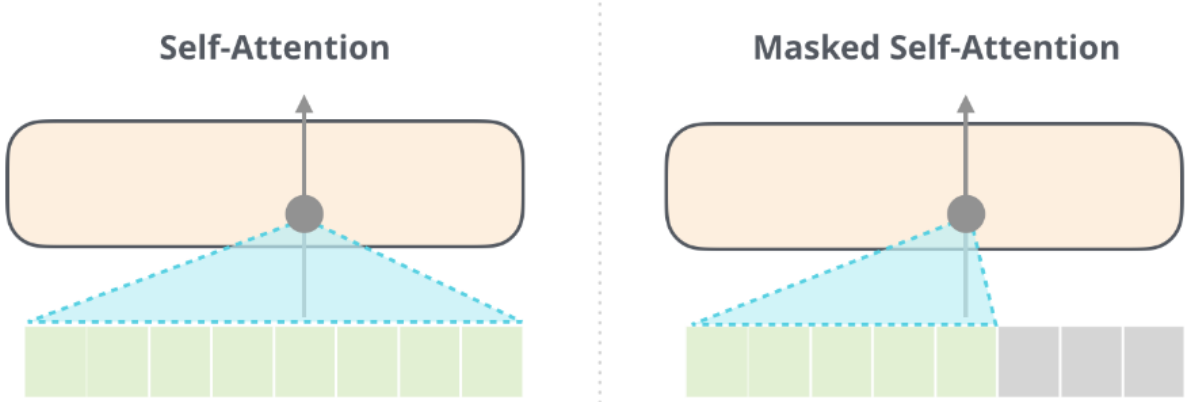
Figure 1: The Transformer - model architecture.

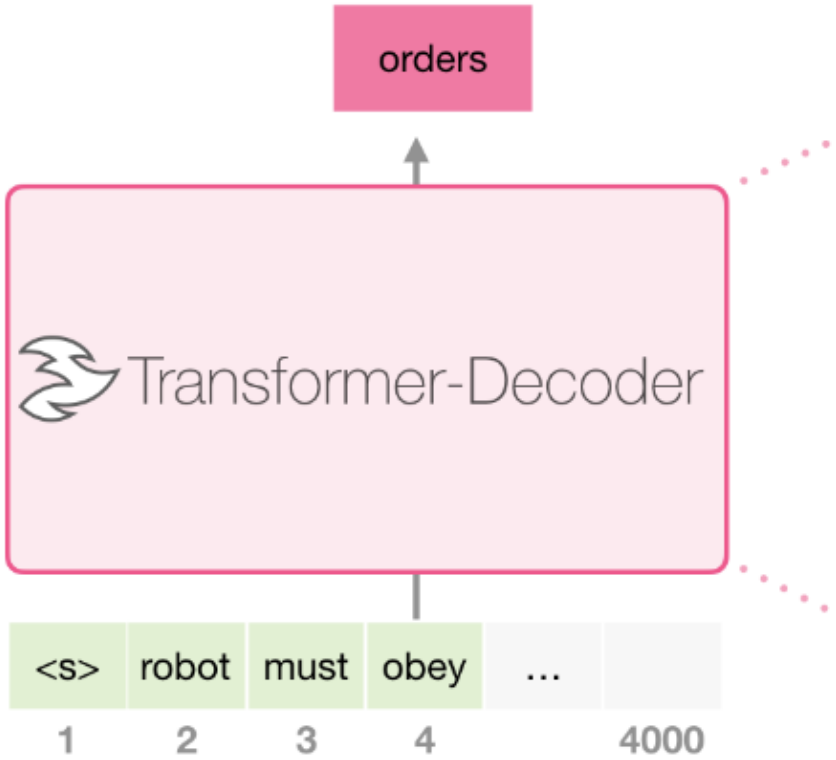
Unsupervised pre-training : Masked Self-Attention

Transformer 디코더의 Masked Self-Attention

: 앞의 단어만 가지고 다음 단어 예측 → LM

➔ 이후 GPT의 Text Generation 능력의 이유





Supervised fine-tuning

Pre-train을 통해 초기화된 파라미터를 각 Task별로 fine-tuning

$$P(y|x^1,\ldots,x^m) = \texttt{softmax}(h_l^m W_y)$$

h_l^m : 마지막 Transformer 블록의 출력

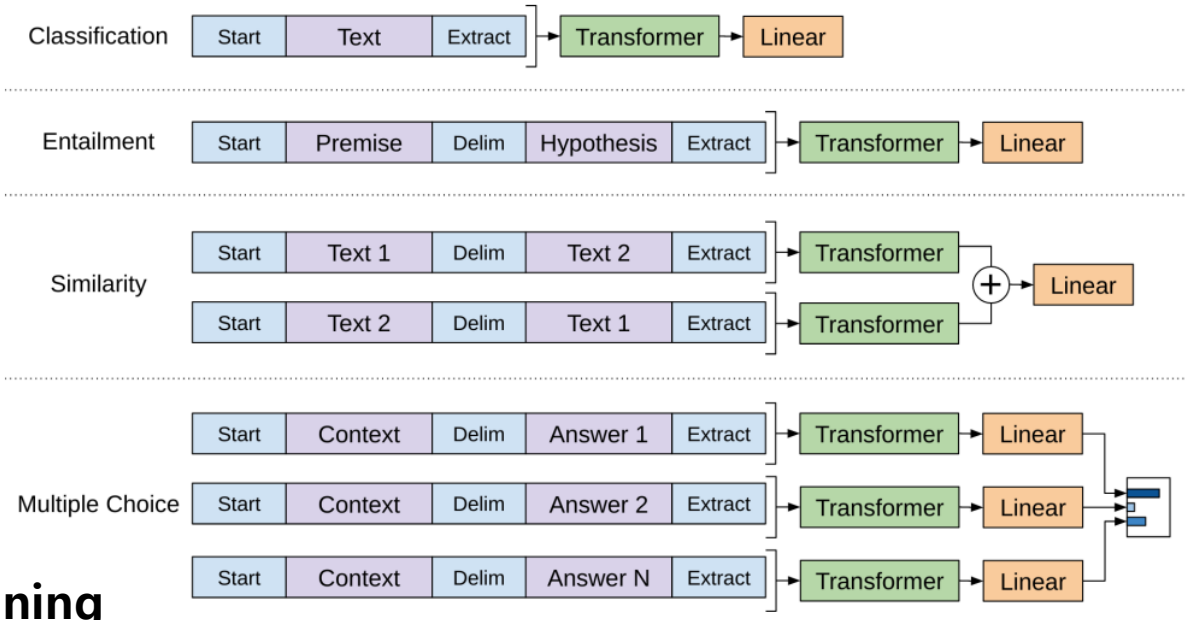
W_y : linear layer의 가중치

Only extra parameter

$$L_2(\mathcal{C}) = \sum_{(x,y)} \log P(y|x^1,\ldots,x^m)$$

L2의 likelihood를 **최대화** 하는 것이 목적 in **fine-tuning**

$\mathcal{C}$: labeled dataset



Supervised fine-tuning : auxiliary objective

Fine-tuning에서 L1을 함께 사용 → 학습에 도움

$$L_3(\mathcal{C}) = L_2(\mathcal{C}) + \lambda * L_1(\mathcal{C})$$

가중치

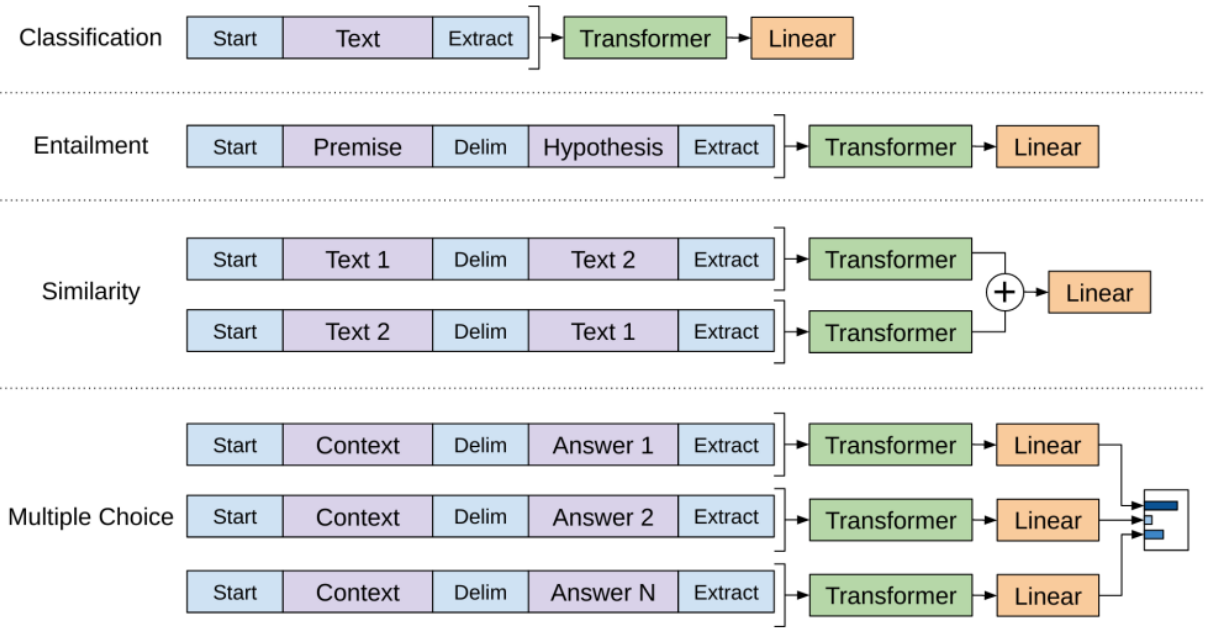
Labeled Corpus에 대한 LM도 함께 업데이트

helped learning by (a) improving generalization of the supervised model, and (b) accelerating convergence. This is in line with prior work [50, 43], who also observed improved performance with

- 다운 스트림 Task에 대한 일반적 성능 향상
- 학습 속도 향상

Task-specific input transformations

모델의 수정을 최소화하기 위해 구조화된 입력(Task마다 다름)을 사용



- Concat → 구분자 $
- 앞뒤로 순서가 바뀐 두 문장의 출력을 Element-wise
- 각 출력에 대해 softmax

데이터셋

| Pre-train(Unlabeled)

✓ **BooksCorpus dataset**

7,000 unique unpublished books from a variety of genres (어드벤처, 판타지, 로맨스)

Fantasy, and Romance. Crucially, it contains long stretches of contiguous text, which allows the generative model to learn to condition on long-range information. An alternative dataset, the 1B

✓ 1B Word Benchmark (alternative, ELMo에서 사용됨)

| Fine-tuning(labeled)

Task	Datasets
Natural language inference	SNLI [5], MultiNLI [66], Question NLI [64], RTE [4], SciTail [25]
Question Answering	RACE [30], Story Cloze [40]
Sentence similarity	MSR Paraphrase Corpus [14], Quora Question Pairs [9], STS Benchmark [6]
Classification	Stanford Sentiment Treebank-2 [54], CoLA [65]

실험 결과 : Natural language Inference

두 쌍의 문장의 관계(**entailment** or **contradiction** or **neutral**)

- The president was **assassinated**.
- The president is **dead**.

 **entailment**

Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	89.3	-	-	-
CAFE [58] (5x)	80.2	79.0	89.3	-	-	-
Stochastic Answer Network [35] (3x)	80.6	80.1	-	-	-	-
CAFE [58]	78.7	77.9	88.5	83.3		
GenSen [64]	71.4	71.3	-	-	82.3	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	**61.7**
Finetuned Transformer LM (ours)	**82.1**	**81.4**	**89.9**	**88.3**	**88.1**	56.0

평가지표로 accuracy 사용

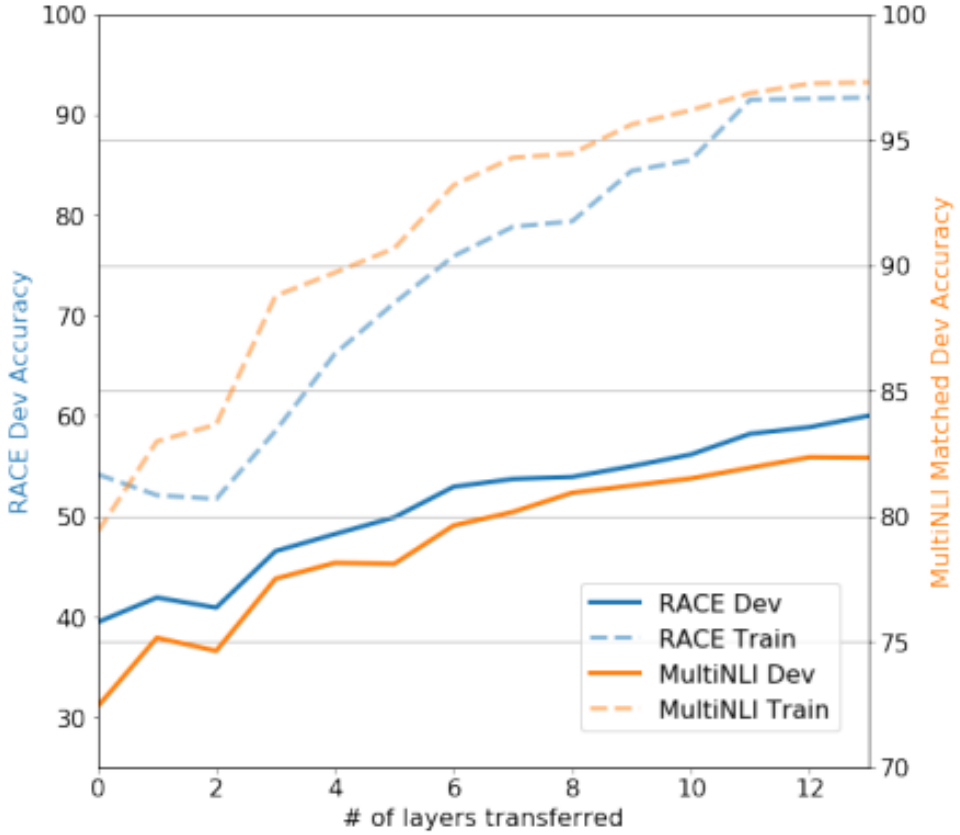
실험 결과 – Question Answering and Commonsense Reasoning

Method	Story Cloze	RACE-m	RACE-h	RACE
val-LS-skip [55]	76.5	-	-	-
Hidden Coherence Model [7]	77.6	-	-	-
Dynamic Fusion Net [67] (9x)	-	55.6	49.4	51.2
BiAttention MRU [59] (9x)	-	60.2	50.3	53.3
Finetuned Transformer LM (ours)	**86.5**	**62.9**	**57.4**	**59.0**

실험 결과 – Semantic similarity

Method	Classification		Semantic Similarity			GLUE
	CoLA (mc)	SST2 (acc)	MRPC (F1)	STSB (pc)	QQP (F1)	
Sparse byte mLSTM [16]	-	**93.2**	-	-	-	-
TF-KLD [23]	-	-	**86.0**	-	-	-
ECNU (mixed ensemble) [60]	-	-	-	81.0	-	-
Single-task BiLSTM + ELMo + Attn [64]	35.0	90.2	80.2	55.5	66.1	64.8
Multi-task BiLSTM + ELMo + Attn [64]	18.9	91.6	83.5	72.8	63.3	68.9
Finetuned Transformer LM (ours)	**45.4**	91.3	82.3	**82.0**	**70.3**	**72.8**

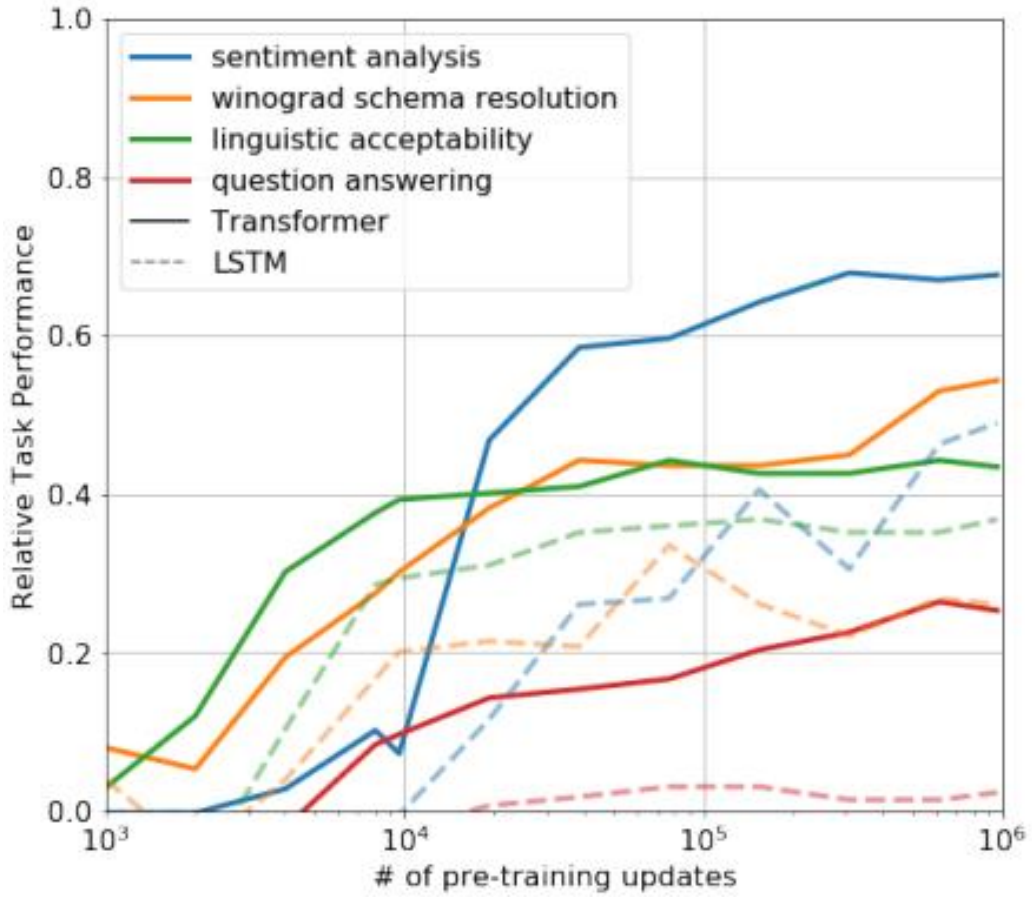
분석 : Impact of number of layers transferred



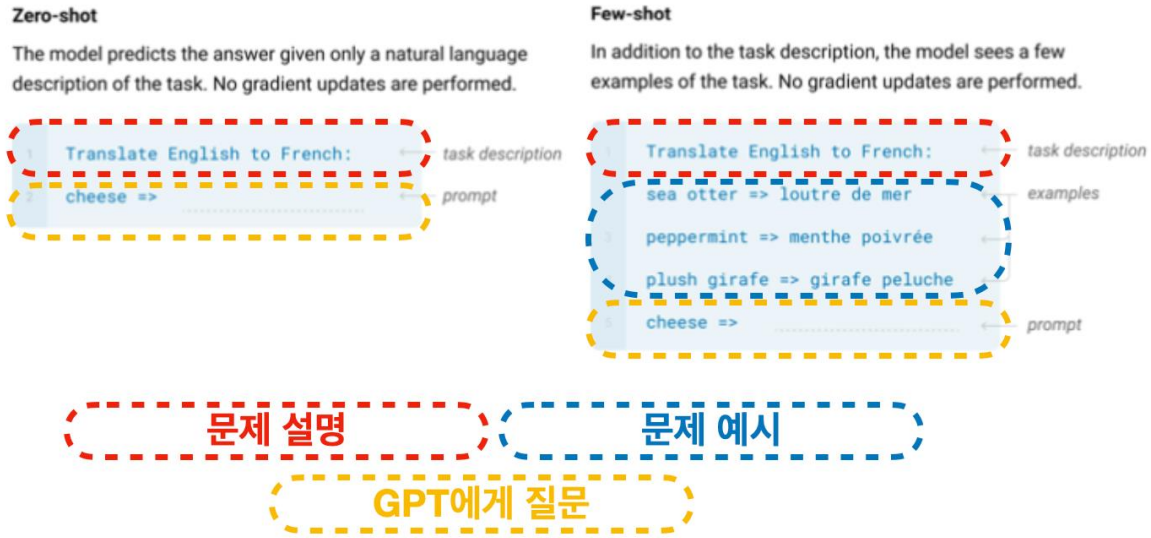
Pre-trained 모델의 디코더 블록의 수↑ ➔ Accuracy↑

12개부터 성능 향상이 완만해짐 → **12개** 사용

분석 : Zero-shot Behaviors



Zero-shot (without supervised fine-tuning)



Generative Pre-training이 광범위한 작업에 유용

분석 : Various Model Ablations

Method	Avg. Score	CoLA (mc)	SST2 (acc)	MRPC (F1)	STSB (pc)	QQP (F1)	MNLI (acc)	QNLI (acc)	RTE (acc)
Transformer w/ aux LM (full)	74.7	45.4	91.3	82.3	82.0	**70.3**	**81.8**	**88.1**	**56.0**
Transformer w/o pre-training	59.9	18.9	84.0	79.4	30.9	65.5	75.7	71.2	53.8
Transformer w/o aux LM	**75.0**	**47.9**	**92.0**	**84.9**	**83.2**	69.8	81.1	86.9	54.4
LSTM w/ aux LM	69.1	30.3	90.5	83.2	71.8	68.1	73.7	81.1	54.6

데이터셋이 큰 경우 : Fine-tuning 중, LM을 목적함수로 함께 사용 → 성능 향상 $L_3(C) = L_2(C) + \lambda * L_1(C)$

데이터셋이 작은 경우 : Fine-tuning 중, LM을 목적함수로 함께 사용 X → 성능 향상

데이터셋이 작다면 LM을 Labeled Corpus로 fine-tuning하지 않는 것이 좋다 $L_2(C) = \sum_{(x,y)} \log P(y|x^1, \ldots, x^m)$

결론

- generative pre-training과 개별적 fine-tuning을 통해 특정 task에 국한되지 않는 모델이 가능함을 보임
- Labeled 데이터를 구축하기 위한 시간, 비용↓
- 제로-샷 모델의 첫 걸음
- 이후 GPT-2, GPT-3 파라미터 수 증가(117M → 345M → 1,542M)

➔ 미세 조정 지분 감소(거의 혹은 전혀)