OpenAI

GPT1
논문 리뷰

한채은
한양대학교 정보공학전공

Index

1 chapter : Introduction

- Background
- Motivation

2 chapter : Framework

- Unsupervised pre - training
- Supervised Fine tuning
- Task-specific input transformations

3 chapter : Experiments

- Datasets & tasks
- Results

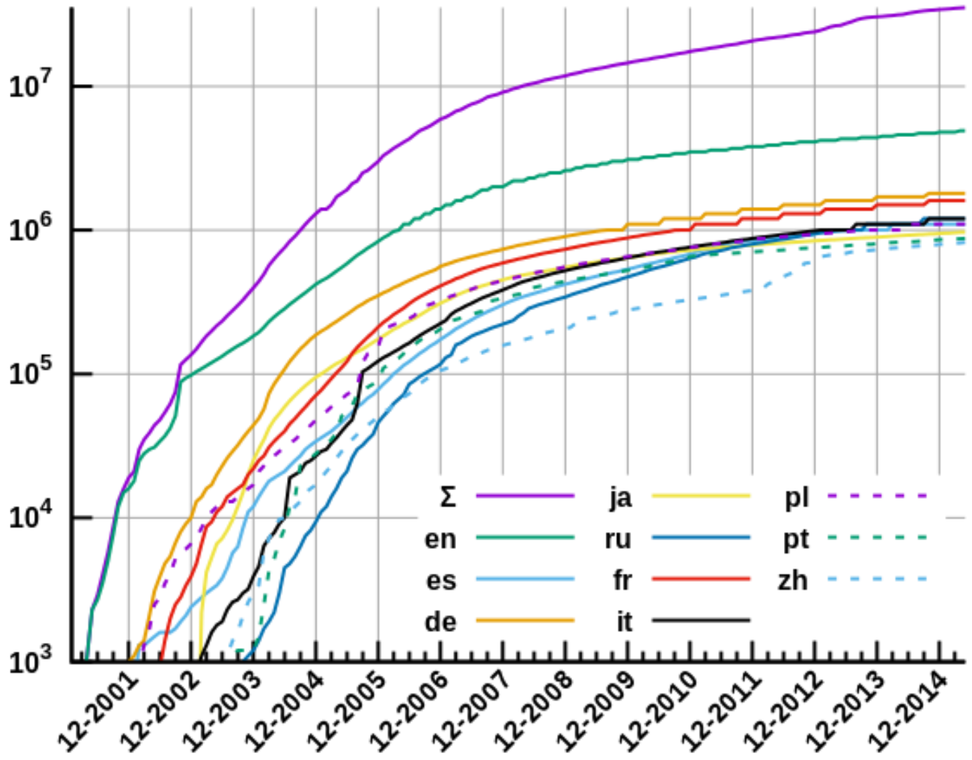
4 chapter : Analysis

- Impact of Number of Layers Transferred & Zero Shot Behaviors
- Ablation studies

ChatGPT

Background

Unlabeled dataset

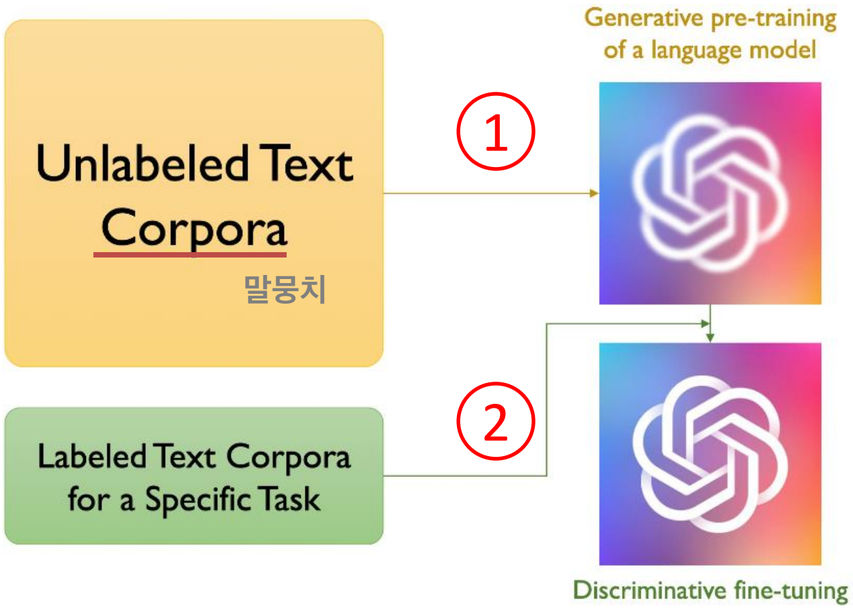


- As of 24 February 2020, Three are 6,020,081 article in the English Wikipedia containing over 3.5 billion words

Labeled dataset

- STS Benchmark for sentence similarity:8,628 sentences
- Quora question pairs:404,290 question pairs
- CoLA dataset:10,657 sentences

Motivation



개념정리

	1 Generative model	2 Discriminative model
개념	데이터를 생성하기 위한 모델 데이터가 생성될 확률인 P(X,Y)	데이터를 구별하기 위한 모델 데이터가 주어졌을 때 특정 클래스에 속할 확률인 P(Y|X)
라벨 필요 여부	라벨 정보가 없어도 모델 구축 가능	라벨 정보 필요 -> supervised learning
예시	강아지, 고양이 이미지 각각의 특징을 학습한 후 강아지/고양이 분류	강아지와 고양이 이미지의 차이를 학습한 후 강아지/고양이 분류
GPT	Unlabeled data를 사용해 보편적인 표현을 학습	Labeled data를 활용해 특정 task에 fine-tuning하는 과정

Motivation

GPT에서 제시하는 문제점

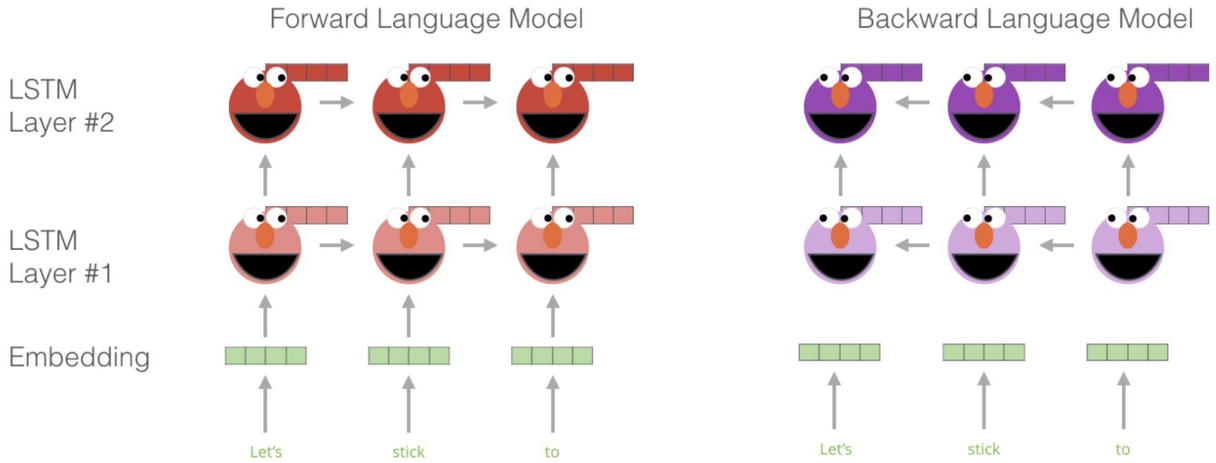
Leveraging more than word-level information from unlabeled text, however, is challenging for two main reasons. First, it is unclear what type of optimization objectives are most effective at learning text representations that are useful for transfer. Recent research has looked at various objectives such as language modeling [44], machine translation [38], and discourse coherence [22], with each method outperforming the others on different tasks.[1] Second, there is no consensus on the most effective way to transfer these learned representations to the target task. Existing techniques involve a combination of making task-specific changes to the model architecture [43, 44], using intricate learning schemes [21] and adding auxiliary learning objectives [50]. These uncertainties have made it difficult to develop effective semi-supervised learning approaches for language processing.

1. 레이블이 되지 않은 텍스트 데이터로부터 단어 레벨 이상의 정보를 사용하는 것은 어렵다. Why ? 어떤 목적 함수가 가장 효과적인지 모른다.

2. 실제로 학습이 되었다고 할지라도, target task 각각에 어떤 Transfer 방식이 가장 효과적인지 정리된 바가 없다.

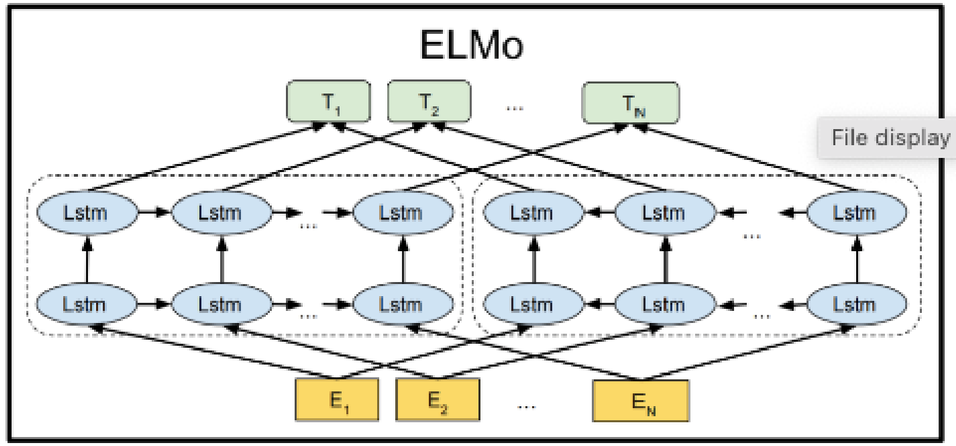
들어가기 전에,

ELMo(Embeddings from Language Model)

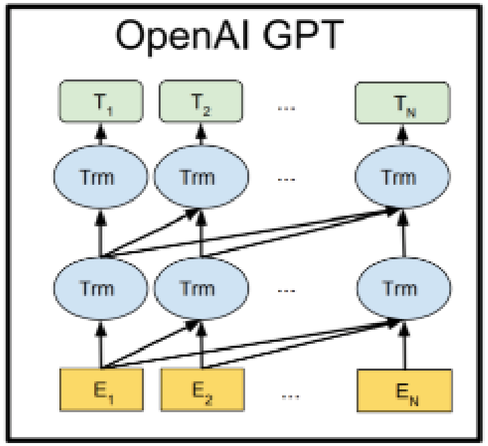


- 문맥을 반영한 새로운 임베딩 방식
- 다의어 임베딩 가능 (한국어 : 눈 , 뜻 : eye, snow 등)
- Forward, Backward 두 방향으로 문장을 읽는 biLSTM 사용

ELMO와 GPT의 구조 차이



VS

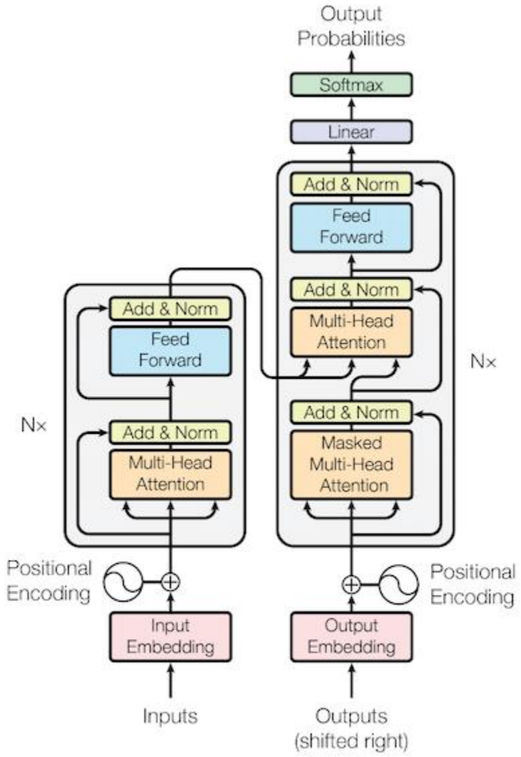


- 문맥을 반영한 새로운 임베딩 방식
- 다의어 임베딩 가능 (한국어 : 눈 , 뜻 : eye, snow 등)
- Forward, Backward 두 방향으로 문장을 읽는 biLSTM 사용

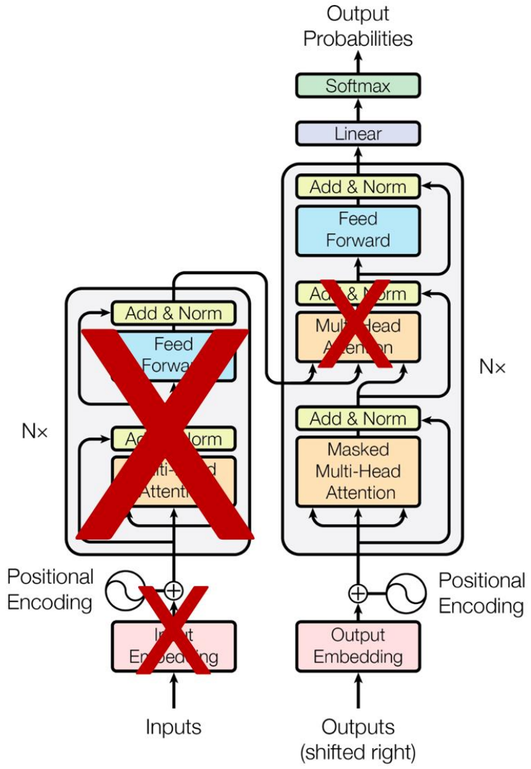
- Transformer decoder block
- Backward 모델 따로 존재 X

Unsupervised pre-training 구조

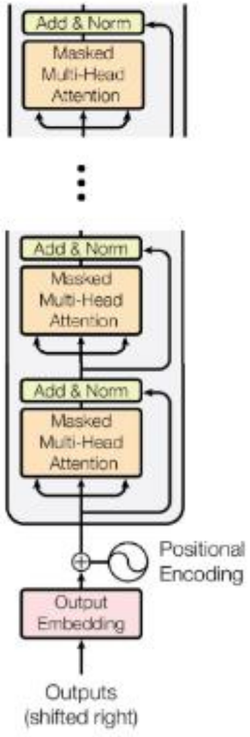
Transformer



only decoder



GPT



Unsupervised pre-training

구조

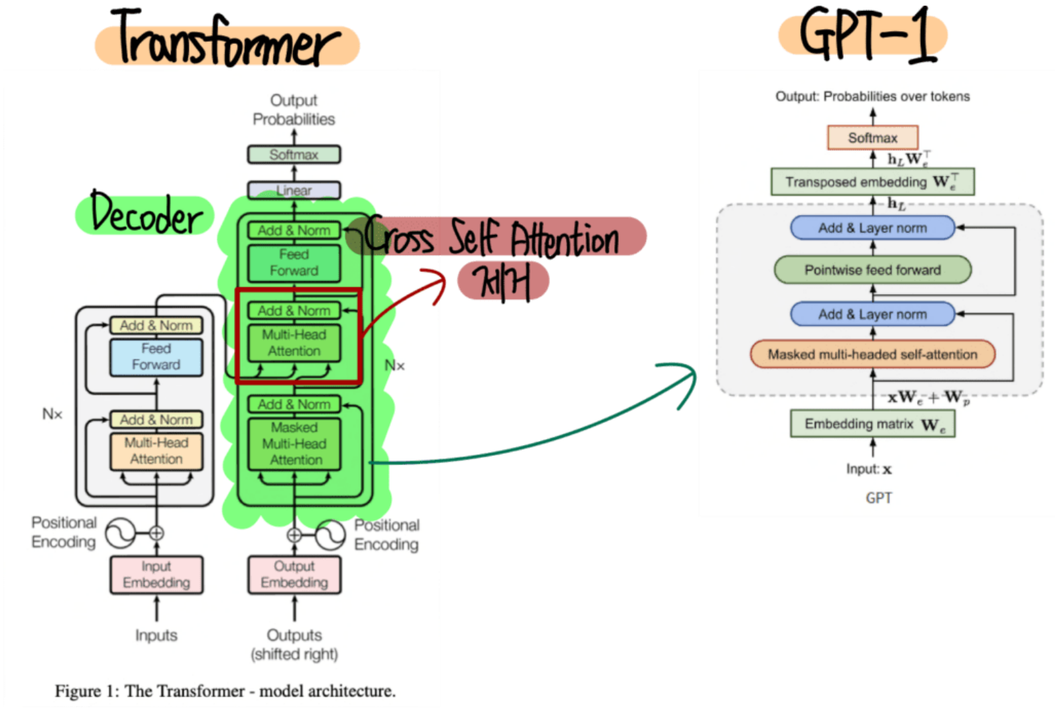


Figure 1: The Transformer - model architecture.

방법

- Next Word Prediction

$$L_1(u) = \sum_i logP(u_i|u_{i-k}, \ldots, u_{i-1}; \theta)$$

Ex)

(나는 오늘 학교에 가서 수업을 듣는다)

GPT1에 입력 > 나는 오늘 학교에

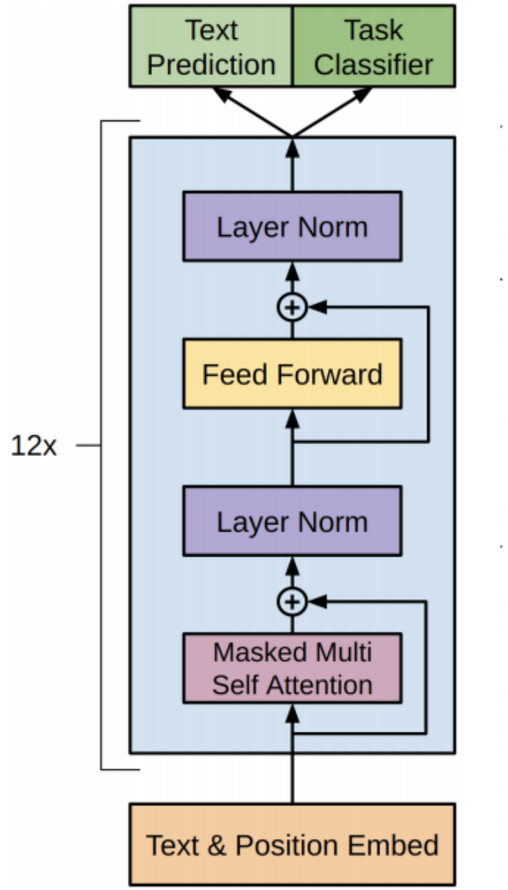
Supervised Fine tuning

This gives us the following objective to maximize:

$$L_2(\mathcal{C}) = \sum_{(x,y)} \log P(y|x^1, \ldots, x^m). \quad (4)$$

We additionally found that including language modeling as an auxiliary objective to the fine-tuning 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 such an auxiliary objective. Specifically, we optimize the following objective (with weight λ):

$$L_3(\mathcal{C}) = L_2(\mathcal{C}) + \lambda * L_1(\mathcal{C}) \quad (5)$$



해석

Pre-trained model에 linear output layer 추가

Pre-training에 사용된 LM의 최적화 함수를 함께 사용하여 a) 일반화 능력 향상 b) 빠른 수렴(학습 속도 향상)

Task-specific input transformations

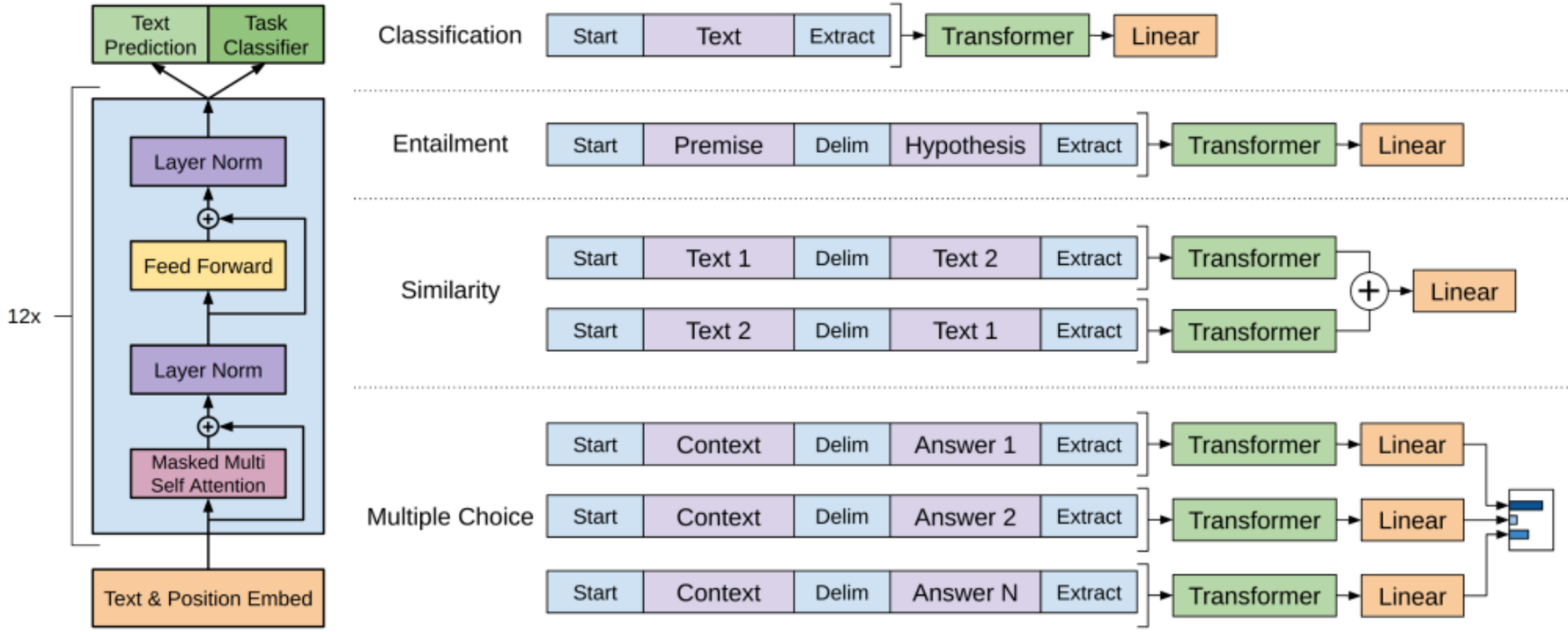


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

Datasets & tasks

Pre-training

Dataset : BooksCorpus(7천개 이상의 다양한 장르의 책 내용을 포함) , 1 Billion Word Language Model Benchmark(used by ELMO)

Fine-tuning

Dataset : 4개의 Task 별로 Dataset 사용 (아래 테이블 참고)

Table 1: A list of the different tasks and datasets used in our experiments.

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]

Results

Task 1 : Natural Language Inference

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

Task 2 : 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**

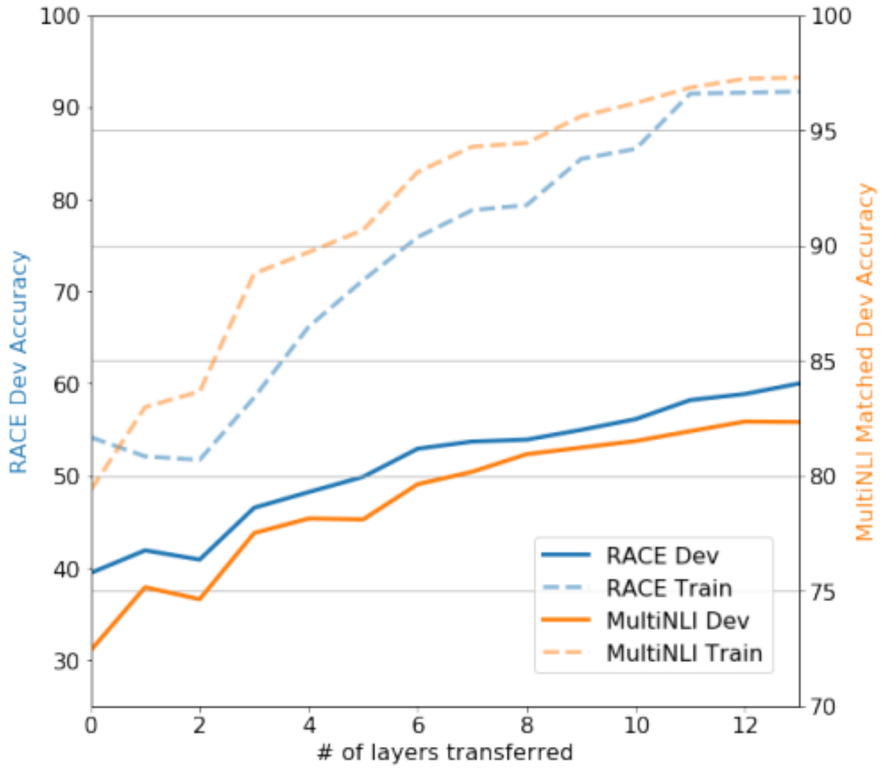
Results

Task 3,4 : Semantic Similarity, Classification

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**

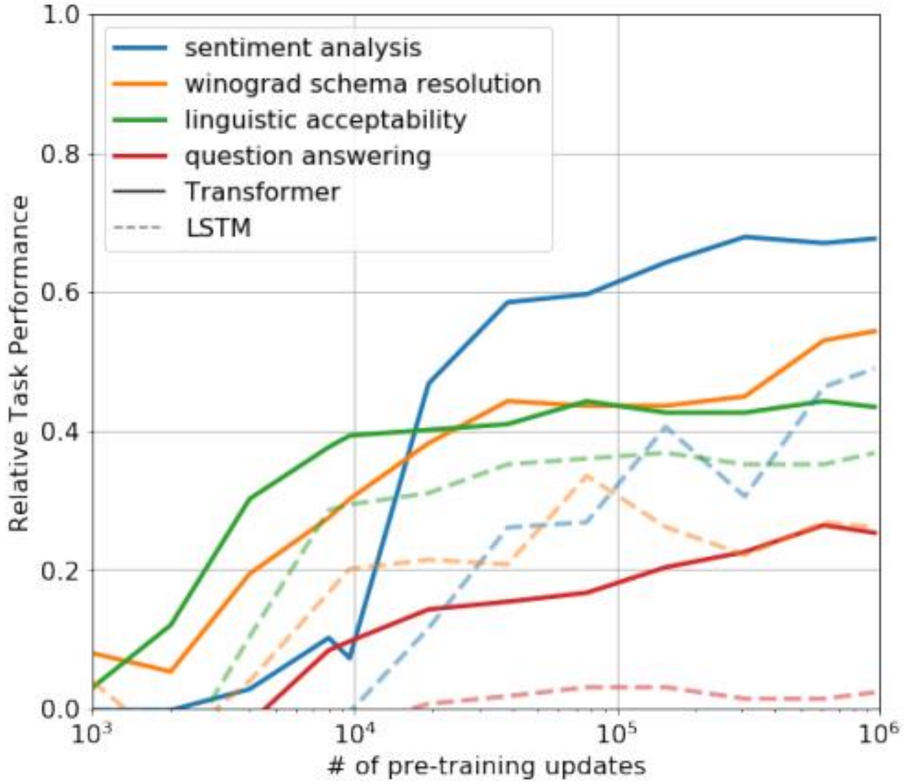
Impact of Number of Layers Transferred & Zero Shot Behaviors

Decoder block을 몇 개나 쌓아야 하는지에 대한 실험



: 쌓을수록 성능 향상

zero-shot / pre-training 파라미터의 업데이트 비교 실험



: pre-training 을 할수록 성능 향상

Ablation studies

기법 별 성능 점검 : (1) GPT1 (2) pre-training 없이 supervised learning만 (3) fine tuning 단계에서 aux LM 제외

(4) Transformer대신 LSTM으로 바꿔서 진행한 경우

Table 5: Analysis of various model ablations on different tasks. Avg. score is a unweighted average of all the results. (*mc*= Mathews correlation, *acc*=Accuracy, *pc*=Pearson correlation)

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

결과

1. L1 auxiliary objective의 효과는 NLI, QQP등 큰 데이터셋에서는 도움이 되고, 작은 데이터셋에서는 도움이 되지 않음
2. Transformer 구조가 LSTM 구조보다 우수함
3. GPT1에서 제안하는 pre-training 방식이 효과적이며 성능에 아주 큰 영향을 줌

Conclusion

1. Generative pre-training과 discriminative fine-tuning을 통해 자연어 이해가 뛰어난 프레임 워크를 소개

2. 문서 분류, 문장 간 유사성 평가, 질의 응답, 문맥적 함의 추론 등 다양한 분야에서 성공적으로 fine-tuning이 이루어졌으며, GPT-1의 성능은 대부분의 데이터셋에서 SOTA를 기록

3. Unsupervised learning에서 많이 학습할수록 성능이 올라가는 빅 언어 모델의 시작을 알림

OpenAI

Thank you