-NLP Study-

발표자: 박무재

AI명예학회

SKHU

목차

AI명예학회
SKHU

- 용어 정리
- GPT1 배경 설명
- GPT1 구조 및 훈련방식
- GPT1 결론
- GPT 2, 3, 4
- 코드

용어

- **Transfer learning: (특정 task에 대해) pre-trained model을 (다른 task에) 재사용**

- **Fine-tuning: 미세 조정**

- **Upstream (task): pretrained model**
  - **다음 단어 맞히기(GPT), 빈칸 채우기(BERT)**
  - **upstream task를 수행한 모델을 LM(언어 모델)이라고 함.**

- **Downstream (task): 최종적으로 만들고자 하는 모델**
  - **문서 분류, 자연어 추론, 문장 생성, QnA**
  - **학습 방식 fine-tuning**

용어

- Downstream Task에서
  - Fine-Tuning
  - Prompt Tuning: 모델을 일부 업데이트
  - In-context Learining: 모델 업데이트x
    - Zero-shot: 다른 정보 없이 바로 downstream task 실행
    - One-shot: 참고할만한 데이터 or form 하나 주고 task
    - Few-shot: 몇 가지 예시를 줌.

좋은 설명글: **https://velog.io/@dongyoungkim/GPT-fine-tuning-5.-in-context-learning,** https://ds-jungsoo.tistory.com/20

GPT란?

Generative(생성의) Pre-trained(사전 학습된) Transformer(트랜스포머)

Output	나는	오늘	그							
Input	나는	오늘								

GPT란?

Generative(생성의) Pre-trained(사전 학습된) Transformer(트랜스포머)

Output	나는	오늘	그							
Input	나는	오늘	그							

GPT란?

Generative(생성의) Pre-trained(사전 학습된) Transformer(트랜스포머)

Output	나는	오늘	그	곳에						
Input	나는	오늘	그							

GPT란?

Generative(생성의) Pre-trained(사전 학습된) Transformer(트랜스포머)

Output	나는	오늘	그	곳에						
Input	나는	오늘	그	곳에						

GPT란?

Generative(생성의) Pre-trained(사전 학습된) Transformer(트랜스포머)

Output	나는	오늘	그	곳에	간다					
Input	나는	오늘	그	곳에						

GPT1 배경

기존 한계

- **task 별로 모델을 학습**
- **unlabeled text가 많아 unsupervise를 갈기고 싶으나 transfer에 유용한 text representation을 학습하기 위한 목적함수의 불분명성, 학습된 representation들을 target task에서 쓰기 위해 가장 효율적인 방법에 대한 협의점 X ->semi-supervised 어렵게 함**

GPT

- **Semi-Supervised approach (unsupervised +supervised)**
- **적은 조정으로도 다양한 task에 쓰일 수 있는Universal representation 학습이 목표**

학습전략

1. Language Modeling Objective와 unlabeled data로 초기 학습
2. Supervised objective로 파라미터들을 target task에 맞게 조정

GPT1 학습 전략

데이터 라벨링하기 귀찮은데..
unlabeled text 왕창 때려 넣으면 언어 자체(text representation)
를 이해하면 인간처럼 여러 task를 할 수 있지 않을까?

GPT1 학습 전략 step1

pre-training(unsupervised)

Given an unsupervised corpus of tokens $\mathcal{U} = \{u_1, \ldots, u_n\}$, we use a standard language modeling objective to maximize the following likelihood:

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

where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters Θ. These parameters are trained using stochastic gradient descent [51].

MLE(maximum likelihood estimation)이 목적 함수
SGD로 파라미터 업데이트(backpropagation)

*MLE는 현재 주어진 데이터만 고려 MAP는 사전 분포를 고려

GPT1 학습 전략 step1

pre-training(unsupervised)

In our experiments, we use a multi-layer *Transformer decoder* [34] for the language model, which is a variant of the transformer [62]. This model applies a multi-headed self-attention operation over the input context tokens followed by position-wise feedforward layers to produce an output distribution over target tokens:

$$\begin{aligned} h_0 &= UW_e + W_p \\ h_l &= \texttt{transformer\_block}(h_{l-1}) \forall i \in [1, n] \\ P(u) &= \texttt{softmax}(h_n W_e^T) \end{aligned} \tag{2}$$

where $U = (u_{-k}, \ldots, u_{-1})$ is the context vector of tokens, n is the number of layers, W_e is the token embedding matrix, and W_p is the position embedding matrix.

Transformer decoder
멀티헤드어텐션 -> self-attention 여러 개 -> 각각 다른 걸 학습(단어위주, 문법)

GPT1 학습 전략 step2

fine-tuning(supervised)

After training the model with the objective in Eq. 1, we adapt the parameters to the supervised target task. We assume a labeled dataset $\mathcal{C}$, where each instance consists of a sequence of input tokens, $x^1, \ldots, x^m$, along with a label y. The inputs are passed through our pre-trained model to obtain the final transformer block's activation h_l^m, which is then fed into an added linear output layer with parameters W_y to predict y:

$$P(y|x^1, \ldots, x^m) = \texttt{softmax}(h_l^m W_y). \tag{3}$$

This gives us the following objective to maximize:

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

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):

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

Overall, the only extra parameters we require during fine-tuning are W_y, and embeddings for delimiter tokens (described below in Section 3.3).

GPT1 학습 전략 step2

fine-tuning(supervised)

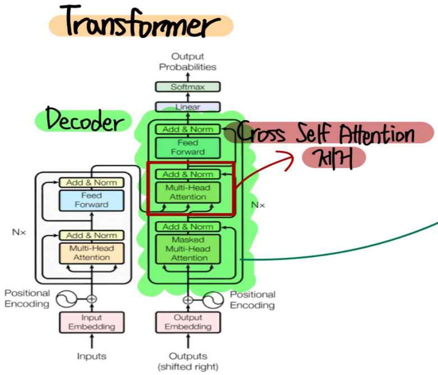
We additionally found that including language modeling as an auxiliary objective to the fine-tuning helped learning by (a) improving generalization of the supervised model, and (b) accelerating convergence. This is in line with prior work [50, 43], who also observed improved performance with such an auxiliary objective. Specifically, we optimize the following objective (with weight λ):

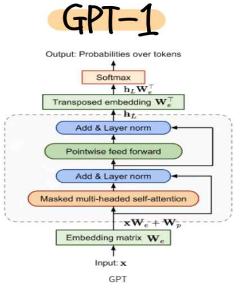
$$L_3(\mathcal{C}) = L_2(\mathcal{C}) + \lambda * L_1(\mathcal{C}) \tag{5}$$

Overall, the only extra parameters we require during fine-tuning are W_y, and embeddings for delimiter tokens (described below in Section 3.3).

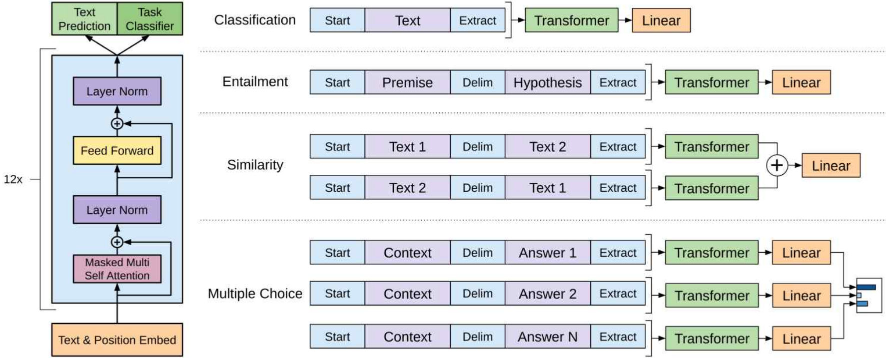
Auxiliary objective -> 지금 fine-tune하는게 classifier일 때 alignment 같은 다른 task도 함께 학습시키는 것 -> 일반화 + 수렴 good

GPT1 구조


Transformer
Decoder
Cross Self Attention
제거
Output
Probabilities
Softmax
Linear
Add & Norm
Feed
Forward
Add & Norm
Multi-Head
Attention
Add & Norm
Masked
Multi-Head
Attention
Nx
Add & Norm
Feed
Forward
Add & Norm
Multi-Head
Attention
Nx
Positional
Encoding
Positional
Encoding
Input
Embedding
Output
Embedding
Inputs
Outputs
(shifted right)


GPT-1
Output: Probabilities over tokens
Softmax
$\mathbf{h}_L \mathbf{W}_e^\top$
Transposed embedding $\mathbf{W}_e^\top$
$\mathbf{h}_L$
Add & Layer norm
Pointwise feed forward
Add & Layer norm
Masked multi-headed self-attention
$\mathbf{x}\mathbf{W}_e + \mathbf{W}_p$
Embedding matrix $\mathbf{W}_e$
Input: $\mathbf{x}$
GPT

GPT1 구조



task별로 모델을 만드는 것이 비효율적이니 input을 task별로 다르게
Start <s>, End <e>

GPT1 결론

Impact of number of layers transferred We observed the impact of transferring a variable number of layers from unsupervised pre-training to the supervised target task. Figure 2(left) illustrates the performance of our approach on MultiNLI and RACE as a function of the number of layers transferred. We observe the standard result that transferring embeddings improves performance and that each transformer layer provides further benefits up to 9% for full transfer on MultiNLI. This indicates that each layer in the pre-trained model contains useful functionality for solving target tasks.

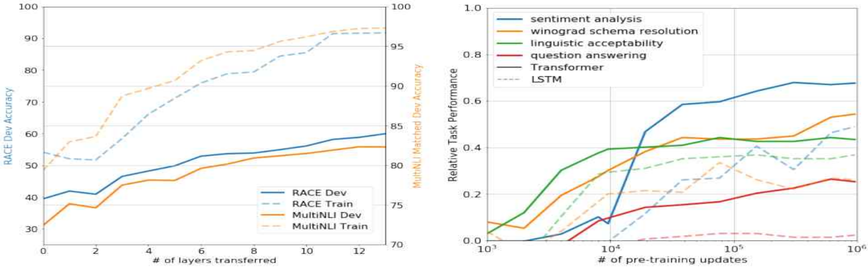


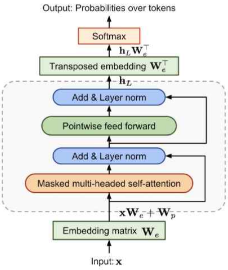
Figure 2: **(left)** Effect of transferring increasing number of layers from the pre-trained language model on RACE and MultiNLI. **(right)** Plot showing the evolution of zero-shot performance on different tasks as a function of LM pre-training updates. Performance per task is normalized between a random guess baseline and the current state-of-the-art with a single model.

GPT1 결론

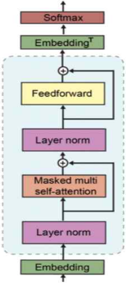
- 달성
  - 12개의 dataset 중 9개 SOTA 모델 달성
- 구조
  - Transformer decoder 사용 -> 구조화된 memory로 효율 up
- 학습방법
  - generative pre-training + discriminative fine-tuning
  - 위를 통해 world knowledge와 긴 문장 처리 및 여러 task 가능

결론: input structure 변환만으로 unsupervise에 대한 가능성을 알려주었다.

GPT2



GPT-1 decoder block



GPT-2 decoder block

GPT-2는 Layer normalization이 sub block의 input 부분으로 옮겨졌고 더하여 마지막 self-attention block 이후에는 추가적인 layer normalization이 존재

또 하나의 변경점은 residual layer의 누적에 따른 initialization의 변화
residual layer의 깊이
에 따라 * weights 를 사용하여 residual layer의 가중치를 설정

또한 vocabulary의 크기가 50,257개로 증가하였으며, 한번에 입력가능한 context size 또한 512에서 1024로 증가

결론: 살짝 바뀌고 크기만 커졌다. GPT-2의 가장 큰 목적은 Fine-tuning 없이 **unsupervised pre-training 만**을 통해 **Zero-shot으로 down-stream task를 진행할 수 있는 General language model을 개발하**는 것. -> 성능 부족

GPT3

https://ffighting.net/deep-learning-paper-review/language-model/gpt-3/

GPT3.5 및 4

https://velog.io/@easter423/GPT-3-vs-GPT-3.5-vs-ChatGPT

-NLP Study-

발표자: 박무재

AI명예학회

SKHU

감사합니다.