Universität Zürich[UZH]

Dissecting GPT-1

Jan Bauer

Machine Learning for Natural Language Processing

Department of Computational Linguistics

January 16, 2022

1 INTRODUCTION

Generative Pre-trained Transformer (GPT) models by OpenAI use generative pre-training of a language model on a large corpus of unlabeled text, followed by discriminative fine-tuning on each specific downstream task. The final model achieves good performance on various tasks, like sentiment analysis, question answering, textual entailment, text summarisation. Prior to this work, most state-of-the-art NLP models were trained specifically on a particular task (e.g sentiment classification, textual entailment) using supervised learning and a dedicated architecture. However, this approach has two major limitations. First of all, a large amount of annotated training data for learning a particular task is often not available. Furthermore, they fail to generalize for tasks other than what they have been trained for. The ability to learn effectively from raw text is crucial to alleviating the dependence on supervised learning in NLP (Shree 2020).

2 ARCHITECTURE

GPT-1 uses the Transformer architecture, since it has been shown that it performs strongly on various tasks like machine translation, document generation, syntactic parsing. Furthermore, Transformers are better with handling long-term dependencies and higher-level semantics in text, compared to alternatives like RNNs and LSTMs. Like shown in Figure 1, the multi-layer Transformer decoder comprises 12 Transformer decoders stacked on top of each other.

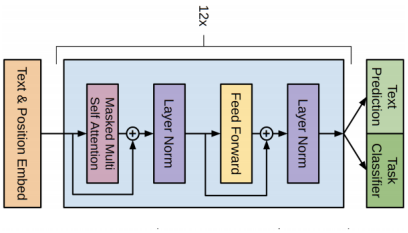

Figure 1: Multi-layer Transformer decoder

Transformers process sentences as a whole rather than word by word. Instead of considering the word order, they use a multi head self attention mechanism. This mechanism helps models to focus on only certain parts of the input and model the relationship between words. This considers the relationship between words, irrespective of where they are placed in a sentence. The first layer performs word and positional embeddings of the input. The positional embeddings are used to get rid of the recurrent structure used by predecessors like RNNs and LSTMs. Abandoning the sequential nature of RNNs makes it easier to take full advantage of modern fast computing devices such as TPUs and GPUs.

3 FRAMEWORK

The system operates in two stages. At first, a transformer model gets trained in an unsupervised manner on a very large corpus of data using language modeling as a training objective. This model is then fine-tuned on much smaller supervised datasets to help it solve specific tasks.

3.1 Unsupervised pre-training

According to Radford and Narasimhan, the name Generative Pre-training, stems from using unsupervised learning as a pre-training objective for subsequent supervised fine-tuning. The goal of this is to learn the initial parameters of a neural network model that acts as a good initialization point for the supervised learning objective. In this step an unlabeled corpus of words $U = \{u_1, \ldots, u_n\}$ is provided as input. This corpus can be gained from various sources like books, online resources (e.g Wikipedia) or newspaper articles. In the case of GPT-1, 7000 books from the BookCorpus dataset were used. A neural network with parameters Θ models the conditional distribution P, while trying to minimize the standard language modelling objective $L_1(U) = \Sigma_i \; log \; P(u_i|u_{i-k}, \ldots, u_{i-1}; \Theta)$ with fixed context size k.

3.2 Supervised fine-tuning

The second step is to adapt the parameters to the supervised target task. A labeled dataset C acts as input and each instance consists of a sequence of input tokens, $x^1, \ldots, x^m$, along with a label y. This part is aimed at maximising the likelihood $L_2(C) = \Sigma_{(x,y)} \; log \; P(y|x^1, \ldots, x^m; \Theta)$. To get the models prediction a linear and softmax layer are appended to the transformer model. The inputs are fed to the pre-trained model and result in the final transformer blocks output h_l^m. This result is then passed through a linear layer with parameters W_y in order to compute the predicted label $\hat{y}$ as $P(y|x^1, \ldots, x^m) = softmax(h_l^m W_y)$. Including language modeling as an auxiliary objective to the fine-tuning, improves generalization and accelerates convergence. This leads to the final objective $L_3(C) = L_2(C) + \lambda * L_1(C)$. In order to fine-tune effectively with minimal changes to the architecture of the pre-trained model, task-specific input transformations might be necessary. These input transformations allow to use the same architecture across different tasks.

4 PERFORMANCE

The models performance on natural language inference, question answering, semantic similarity, and text classification was assessed. In 9 out of the 12 studied tasks, GPT-1 outperformed specifically constructed and discriminatively trained models.

DATASET	TASK	SOTA	OURS
SNLI	Textual Entailment	89.3	**89.9**
MNLI Matched	Textual Entailment	80.6	**82.1**
MNLI Mismatched	Textual Entailment	80.1	**81.4**
SciTail	Textual Entailment	83.3	**88.3**
QNLI	Textual Entailment	82.3	**88.1**
RTE	Textual Entailment	**61.7**	56.0
STS-B	Semantic Similarity	81.0	**82.0**
QQP	Semantic Similarity	66.1	**70.3**
MRPC	Semantic Similarity	**86.0**	82.3
RACE	Reading Comprehension	53.3	**59.0**
ROCStories	Commonsense Reasoning	77.6	**86.5**
COPA	Commonsense Reasoning	71.2	**78.6**
SST-2	Sentiment Analysis	**93.2**	91.3
CoLA	Linguistic Acceptability	35.0	**45.4**
GLUE	Multi Task Benchmark	68.9	**72.8**

Figure 2: SOTA vs GPT-1 performance

Like shown in Figure 2 big absolute improvements of 8.9% were made on commonsense reasoning (Stories Cloze Test on ROCStories Corpora), 5.7% on question answering (RACE), 1.5% on textual entailment (MultiNLI) and 5.5% on the recently introduced GLUE multi-task benchmark. The zero-shot behaviors of the pre-trained model on four different settings also demonstrate that it acquires useful linguistic knowledge for downstream tasks.

5 KEY TAKEAWAYS

GPT-1 was the first scalable and task-agnostic system, provide a convincing example that pairing supervised learning methods with unsupervised pre-training can work very well on a diverse set of tasks. The authors hope to foster new research into unsupervised learning, for both natural language understanding and other domains further improving the understanding of how and when unsupervised learning works.

REFERENCES

OpenAI (2018) *The Comprehensive Tex Archive Network.*

Radford, Alec and Karthik Narasimhan (2018) “Improving Language Understanding by Generative Pre-Training”. In.

Shree, Priya (2020) *The Journey of Open AI GPT models.*