OpenAI

Language Models + NLP Tutorial

Rewon Child, SEA MLS 2019

Overview

- **15 mins**: Why LMs? Why LMs for NLP?
- **~1 hour**: Implement a Transformer LM from scratch in TensorFlow and get it training

LMs are (intrinsically) useful for the objective they optimize

- Large scale text generation (e.g., GPT-2)
- Can generate images, raw audio (PixelCNN, WaveNet, Sparse Transformer)
- Music generation (midi) (MuseNet, Music Transformer, and more)
- Speech recognition, machine translation, many conditional generation tasks

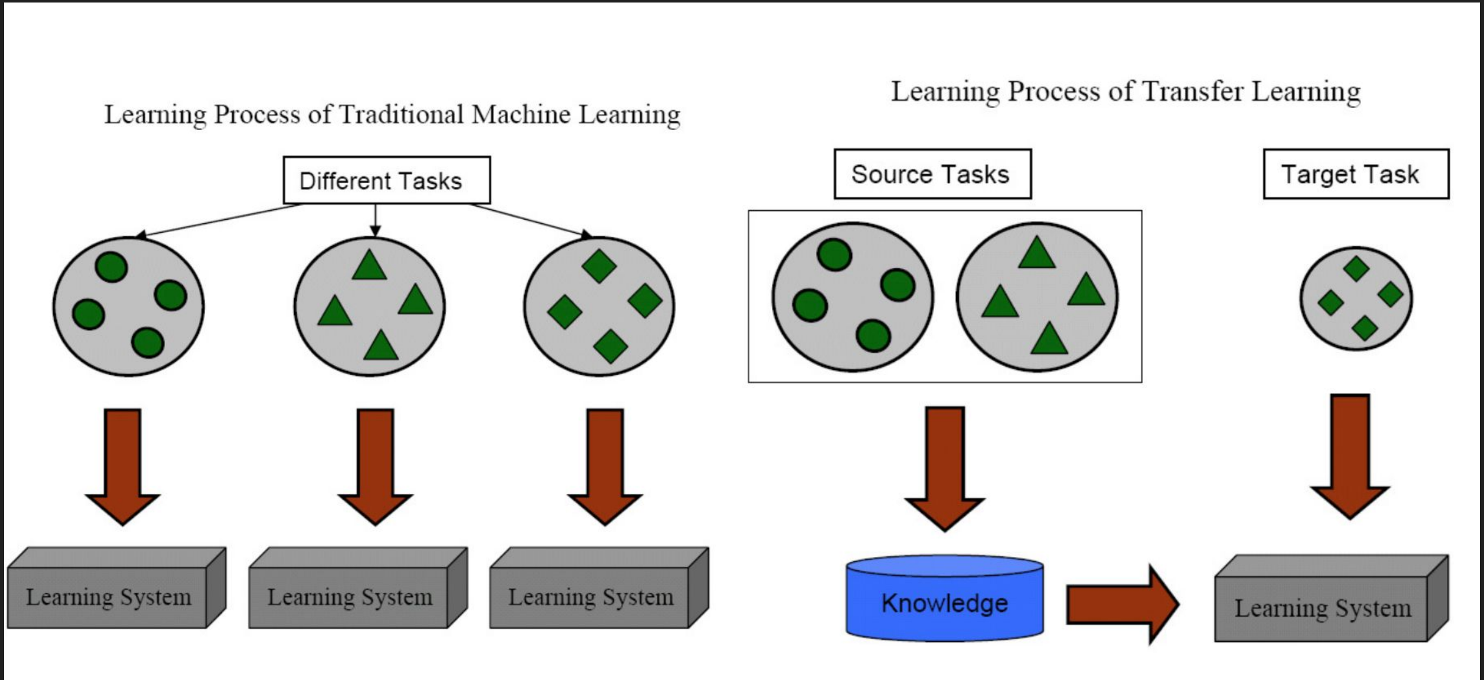
LMs are (intrinsically) useful for the objective they optimize

- Large scale text generation (e.g., GPT-2)
- Can generate images, raw audio (PixelCNN, WaveNet, Sparse Transformer)
- Music generation (midi) (MuseNet, Music Transformer, and more)
- Speech recognition, machine translation, many conditional generation tasks

But what if you care about other, non-generation specific tasks?

Examples: text classification, named entity recognition, i.e. most NLP. Why train an LM, instead of just directly training a network on that objective?

LMs can be used for transfer learning to NLP tasks



Pan and Yang, 2009

Main idea behind ELMo, GPT & BERT

Main idea behind ELMo, GPT & BERT:

- Embeddings from Language Models (ELMo):
  - Pretrain an bidirectional autoregressive LM on a large unsupervised text dataset
  - Then, finetune the model on an array of downstream NLP tasks

Peters et al 2018

Main idea behind ELMo, GPT & BERT:

- Embeddings from Language Models (ELMo):
  - Pretrain an bidirectional autoregressive LM on a large unsupervised text dataset
  - Then, finetune the model on an array of downstream NLP tasks
- Language Understanding via Generative Pretraining (GPT):
  - Use a (unidirectional) Transformer instead

Peters et al 2018, Radford et al 2018

Main idea behind ELMo, GPT & BERT:

- Embeddings from Language Models (ELMo):
  - Pretrain an bidirectional autoregressive LM on a large unsupervised text dataset
  - Then, finetune the model on an array of downstream NLP tasks
- Language Understanding via Generative Pretraining (GPT):
  - Use a (unidirectional) Transformer instead
- Bidirectional Encoding Representations from Transformers (BERT):
  - Use a “masked self prediction” objective, which allows the model to use bidirectional context
  - Not an autoregressive model, but still implicitly defines a generative process (Mansimov et al, 2019)

Peters et al 2018, Radford et al 2018, Devlin et al 2018

Empirically, these methods perform well

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Table 1: GLUE Test results, scored by the evaluation server (`https://gluebenchmark.com/leaderboard`). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set.[8] BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks. We exclude entries that use BERT as one of their components.

Figure from Devlin et al, 2018
General Language Understanding Evaluation (GLUE) benchmark, Wang et al 2019

Generatively pretrained LMs can do zero-shot NLP

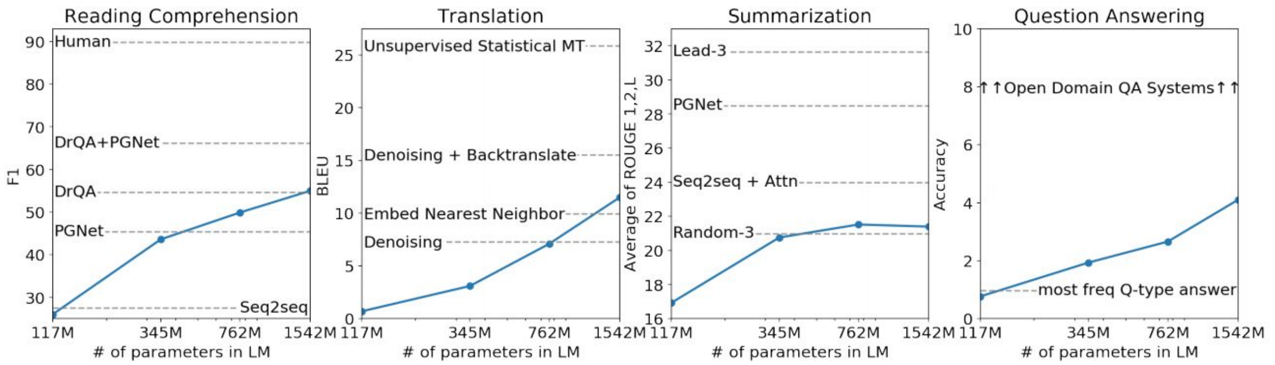


Figure 1. Zero-shot task performance of WebText LMs as a function of model size on many NLP tasks. Reading Comprehension results are on CoQA (Reddy et al., 2018), translation on WMT-14 Fr-En (Artetxe et al., 2017), summarization on CNN and Daily Mail (See et al., 2017), and Question Answering on Natural Questions (Kwiatkowski et al., 2019). Section 3 contains detailed descriptions of each result.

GPT-2: Radford et al 2019

Summary

- LM pretraining + transfer is an simple + easy approach, and does quite well.
- Will get better with larger models and better datasets
- Generating data is also intrinsically useful and exciting in many domains.

Summary

- LM pretraining + transfer is an simple + easy approach, and does quite well.
- Will get better with larger models and better datasets
- Generating data is also intrinsically useful and exciting in many domains.

So let’s implement one!

- Implement a simple transformer language model that is roughly the same architecture as GPT-2.

Colab link: https://bit.ly/32ilBco

Interested in more?

Great tutorial from Ruder et al (2019) at NAACL on Transfer learning in NLP

https://colab.research.google.com/drive/1iDHCYIrWswIKp-n-pOg69xLoZO09MEgf#scrollTo=E2Z8CC-IW1Nq

Unified Pytorch implementation of BERT, GPT, and more: https://github.com/huggingface/pytorch-pretrained-BERT

References

- Pan, Sinno Jialin, and Qiang Yang. "A survey on transfer learning." IEEE Transactions on knowledge and data engineering 22.10 (2009): 1345-1359.
- Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).
- Peters, Matthew E., et al. "Deep contextualized word representations." *arXiv preprint arXiv:1802.05365* (2018)
- Radford, Alec, et al. "Improving language understanding by generative pre-training." *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf* (2018).
- Radford, Alec, et al. "Language models are unsupervised multitask learners." *OpenAI Blog* 1.8 (2019)
- Mansimov, Elman, Alex Wang, and Kyunghyun Cho. "A Generalized Framework of Sequence Generation with Application to Undirected Sequence Models." *arXiv preprint arXiv:1905.12790* (2019).
- Wang, Alex, et al. "Glue: A multi-task benchmark and analysis platform for natural language understanding." *arXiv preprint arXiv:1804.07461* (2018).