Text generation with GPT-2

Fine-tuned model on “Song of Ice and Fire”

Nurzhan Sarzhan
Applied Mathematics
17.12.2020

Previous work

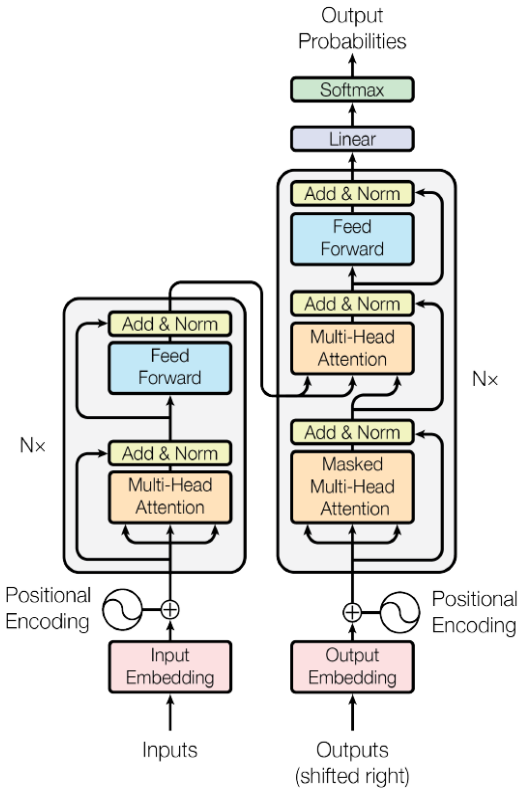
Disaster detection with Twitter and BERT

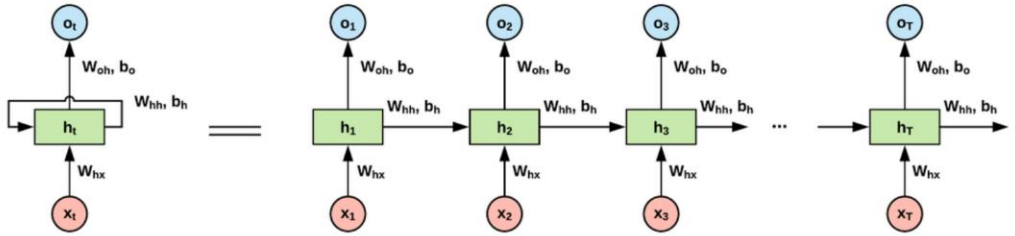
Created a text classifier:

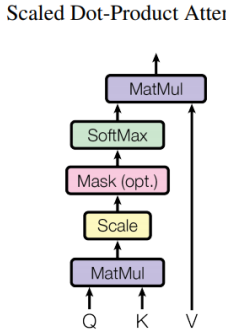
- Detects tweets about disaster (classifies disaster and non-disaster tweet)
- Based on BERT (Bidirectional Encoder Representations from Transformers) by Google team
- Trained on Disaster tweets dataset from Kaggle

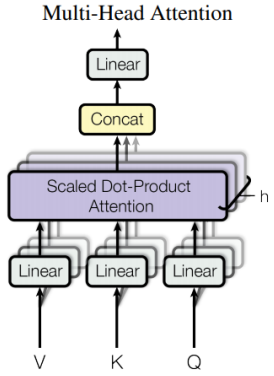
Historical background in Language Modelling

A statistical language model is a probability distribution over sequences of words









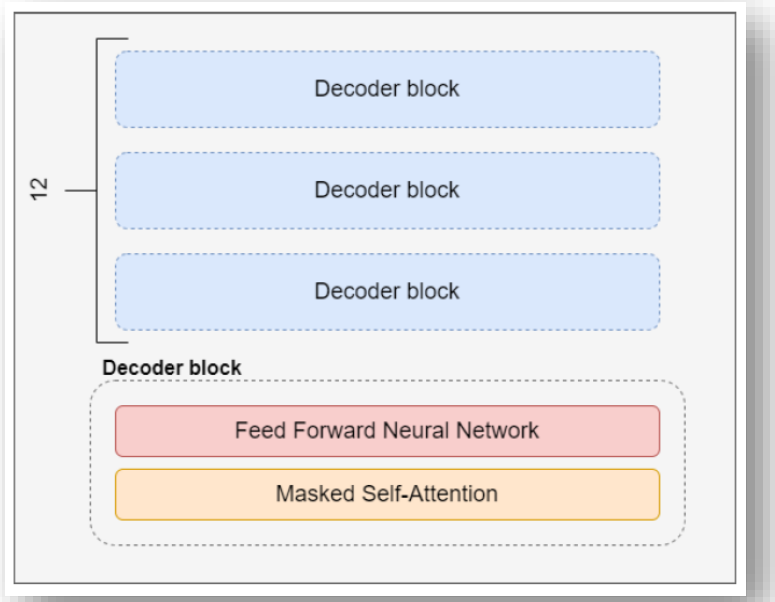
Classical approach (BOW, TFIDF, etc.)

Recurrent Neural Networks (RNN, LSTM, GRU)

Transformers (Attention based NNs, GPT-2)

GPT-2

OpenAI



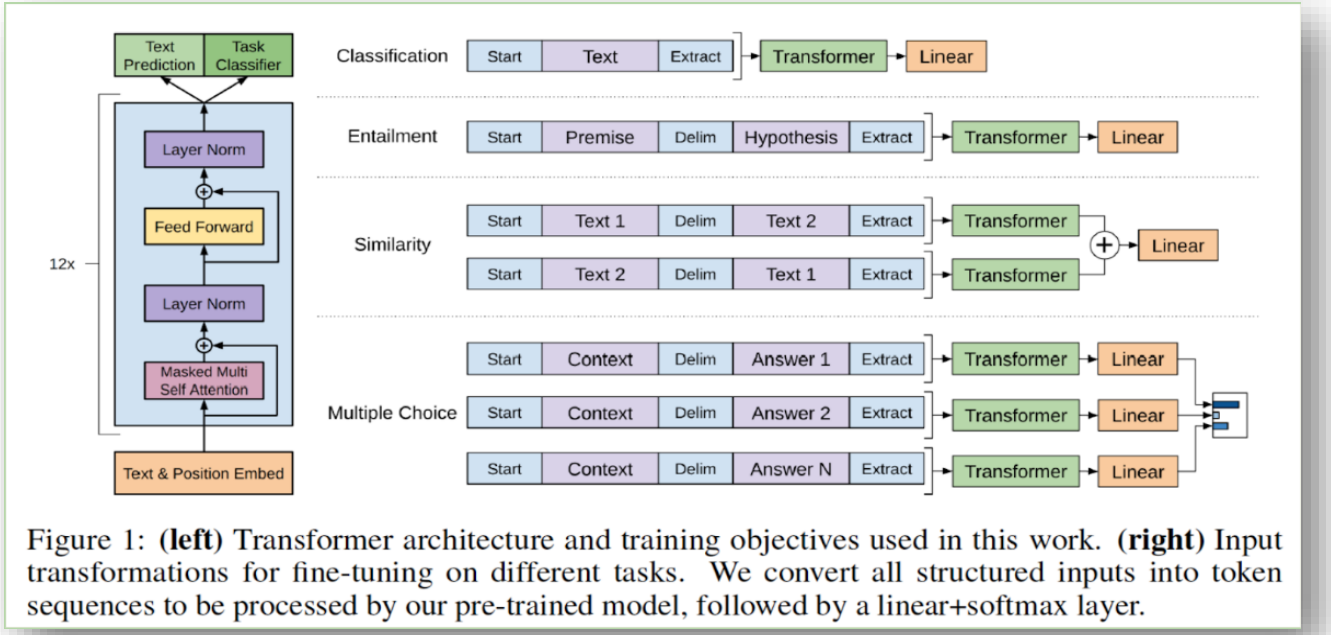
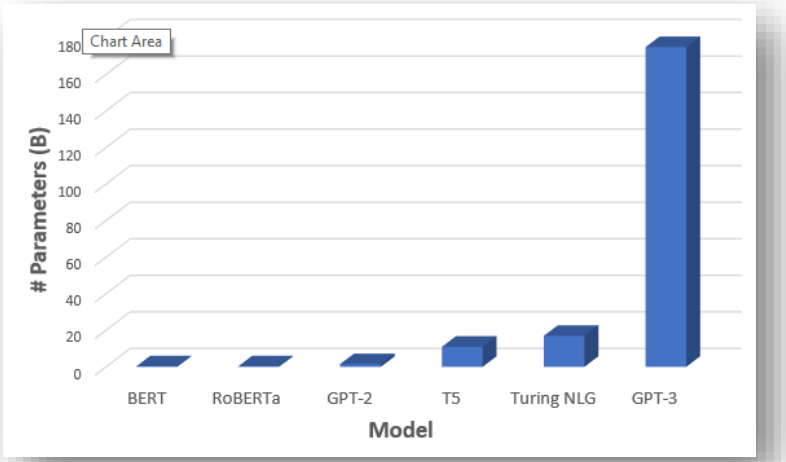
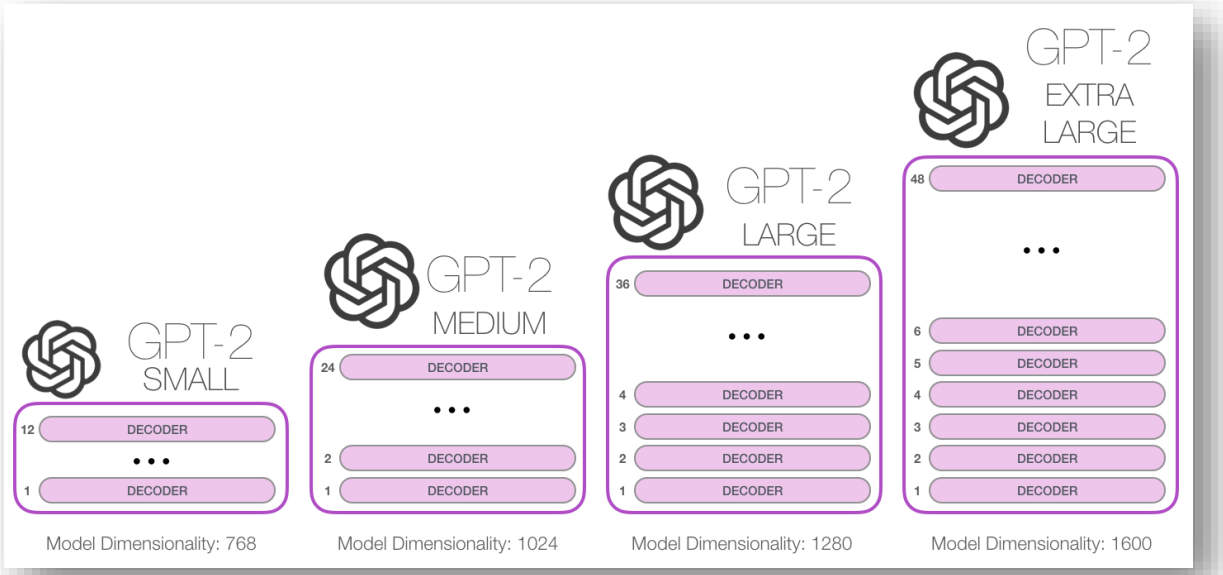
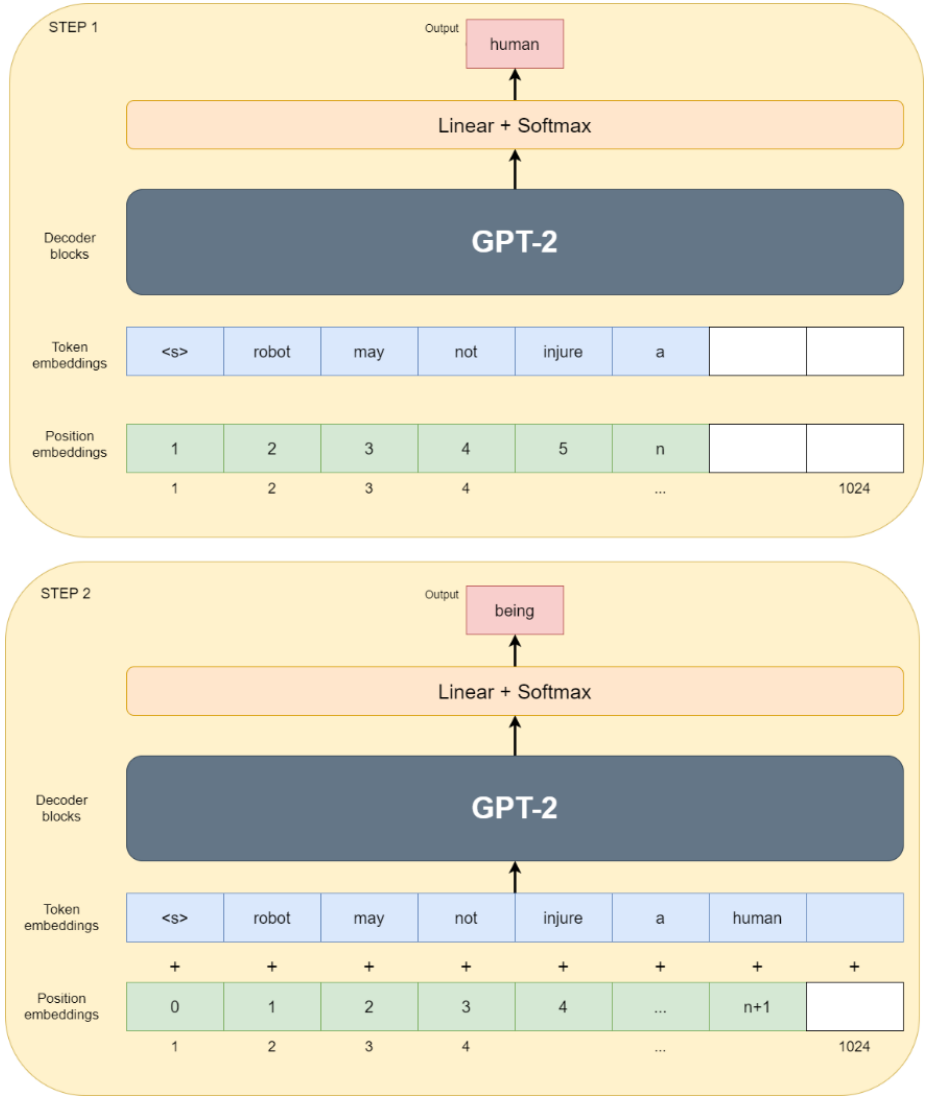


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.





How it works



Algorithm of text generation:

1. Input sequence of words is converted into tokens (with BPE tokenizers)
2. Input tokens are converted into vector space (word embeddings ELMO + position embeddings)
3. Input embedding then processed through GPT-2
    1. They go through 12 decoder blocks of Transformer
    2. Output vector then processed into Feed Forward Network (Classifier) + Softmax
    3. Then Top-k sampling method applied to output of the classifier (probabilities of tokens in dictionary)
4. Output token is chosen (next in the sequence) and added into the sequence as an input
5. Go to step 3 or exit if stopping condition is satisfied (end of sentence is generated or total length of sequence exceeds 1024)

Training results

Results:

- Training loss = 3.2998
- Validation loss = 3.0993
- Perplexity = 21.2652 (comparing to 37.4395 on non fine-tuned model)
- Model size = 498MB

Training Parameters:

- Number of epochs = 5
- Training batch size = 32
- Learning rate = 1e-5
- Optimizer = Adam with weight decay (AdamW)
- Sentence size in Dataloader = 128
- Tesla T4 GPU and 13GB of RAM

Dataset info:

- 5 books of “Song of Ice and Fire” series
- 4.5MB / 83.000 sentences
- 80 to 20 for training and test sets split respectively

$$PP(W) = P(w_1, w_2, ..., w_N)^{\frac{1}{N}}$$

$$= \sqrt[N]{\frac{1}{P(w_1, w_2, ..., w_N)}}$$

$$= \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_1, ..., w_{i-1})}}$$

colab

jupyter

Inference

Interesting examples:

- Tyrion Lanister's daughter came crying and the king asked if she had been frightened, but the Lannisters let her out.
- Tyrion's face had turned pale at the cold touch of the blood inside.
- Arya Stark sent his brother to inspect the remains of her husband's body.

Common not controlling examples:

- *It wasn't so big a fight as she feared, though. He had to be there, that was all."I fear the wolf will never come in sight of me. Or we'll be worse off now.*
- *By the time they arrived at the front door, she was feeling dizzy. The sun was shining down on them above the city walls, and with her back to them, they saw a dozen of her own people, half-dozen in j*
- *What's in there?"Dornishman-beyond-the-Wall is a great stone fortress, built after the Conquest, and under the protection of the Great Sept of Baelor and the First Men, who*

Examples with subsequence "King of the seven" as input:

- *King of the seven lands, the Wall will be a refuge for the men of the Seven Kingdoms.""We want them." The other wolf turned on his heels, and they exchanged a glance. "How many times do*
- *King of the seven thousand men now known as Lord Stark," said Jon Davenport. "Lord Walder, captain of the King's Guards and Hand of the King, commander of the Trident.*
- *King of the seven great powers, in their wisdom, to rise up against an enemy more formidable than King Robert or the Others. They had the power to fight and win, and they learned the truth of that truth from the gold in your hands.*

Conclusion and Future work

- Good base model for further fine-tuning
- Hardly controls context
- Stops on the middle of the sentence
- Enrichment of the dataset and more training process

[Github link to the project]