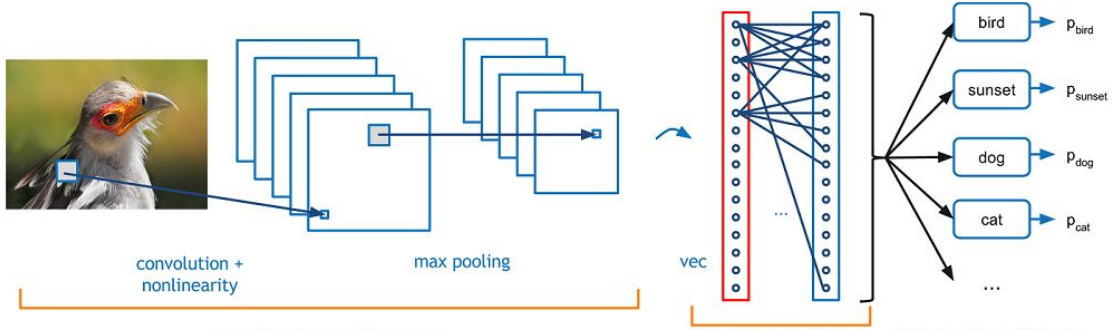
BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Outline

- Feature Extractor History in NLP
  - One-hot and word embedding
  - ELMo
  - BERT and GPT-2
  - How to use pretrained models
- Transformer in detail
  - Self Attention
  - Layer Normalization and Feed Forward
- Pretrain Tasks
- Questions

Recall: Use of ImageNet in Training a General Purpose Feature Extractor

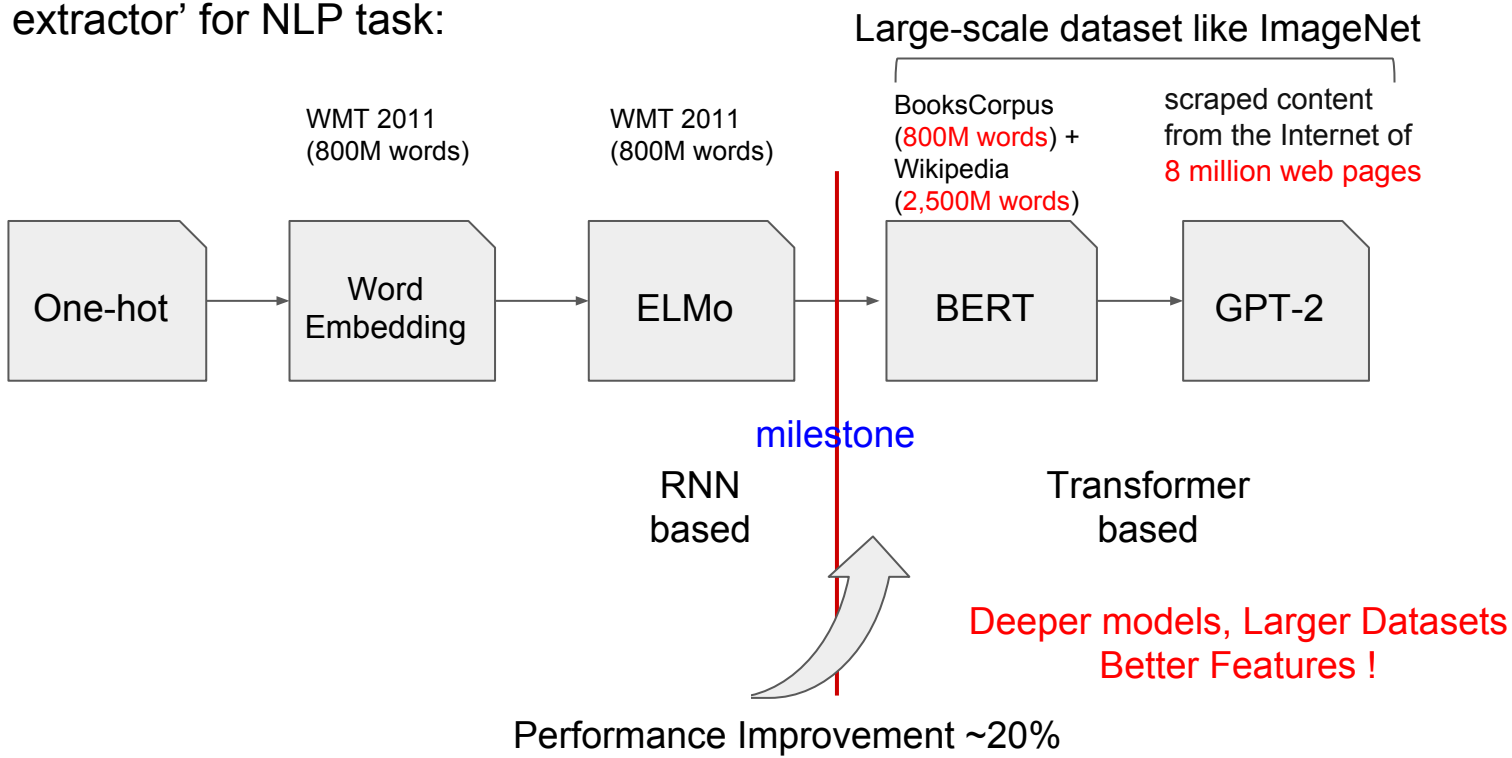
- ImageNet is a large vision dataset which has 1.2 billion training images and 100k testing images in 1,000 object classes. The success of ImageNet highlighted that in the era of deep learning, big data was at least as important as algorithms for performance.
- As a large scale dataset, ImageNet can be used to train a base model which can then be transferred to various tasks via fine-tuning. We call such a model *Feature Extractor*.



Recap: ImageNet-moment for NLP

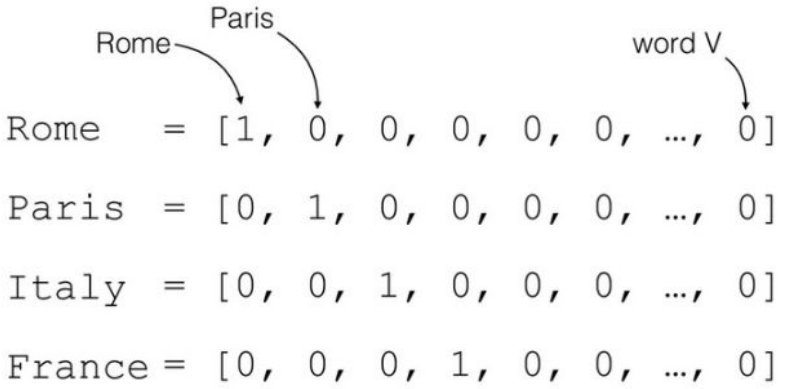
- Can we build such a feature extractor to power various NLP tasks?
- For NLP, features are hidden vectors for each word.

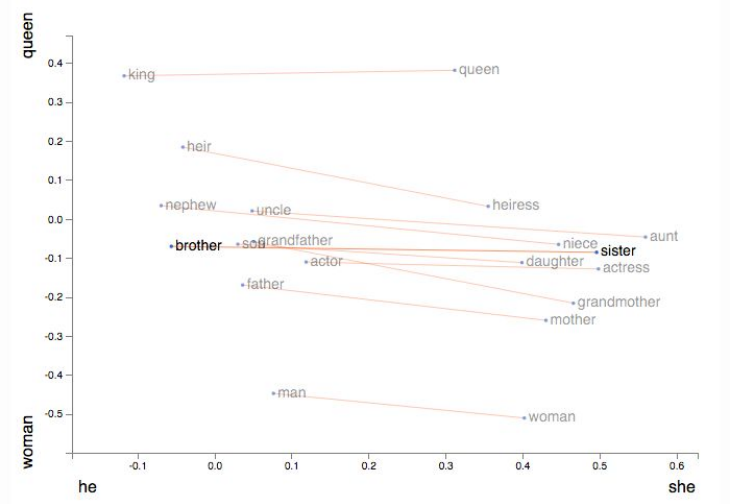
Development of pretrained
'feature extractor' for NLP task:



One-hot and Word Embedding

- One-Hot Encoding:
  - A one hot encoding is a representation of categorical variables as binary vectors.
- Word Embedding:
  - Word2Vec shows that we can use a vector (a list of numbers) to properly represent words in a way that captures *semantic* or meaning-related relationships. (e.g. king - man + woman = queen)

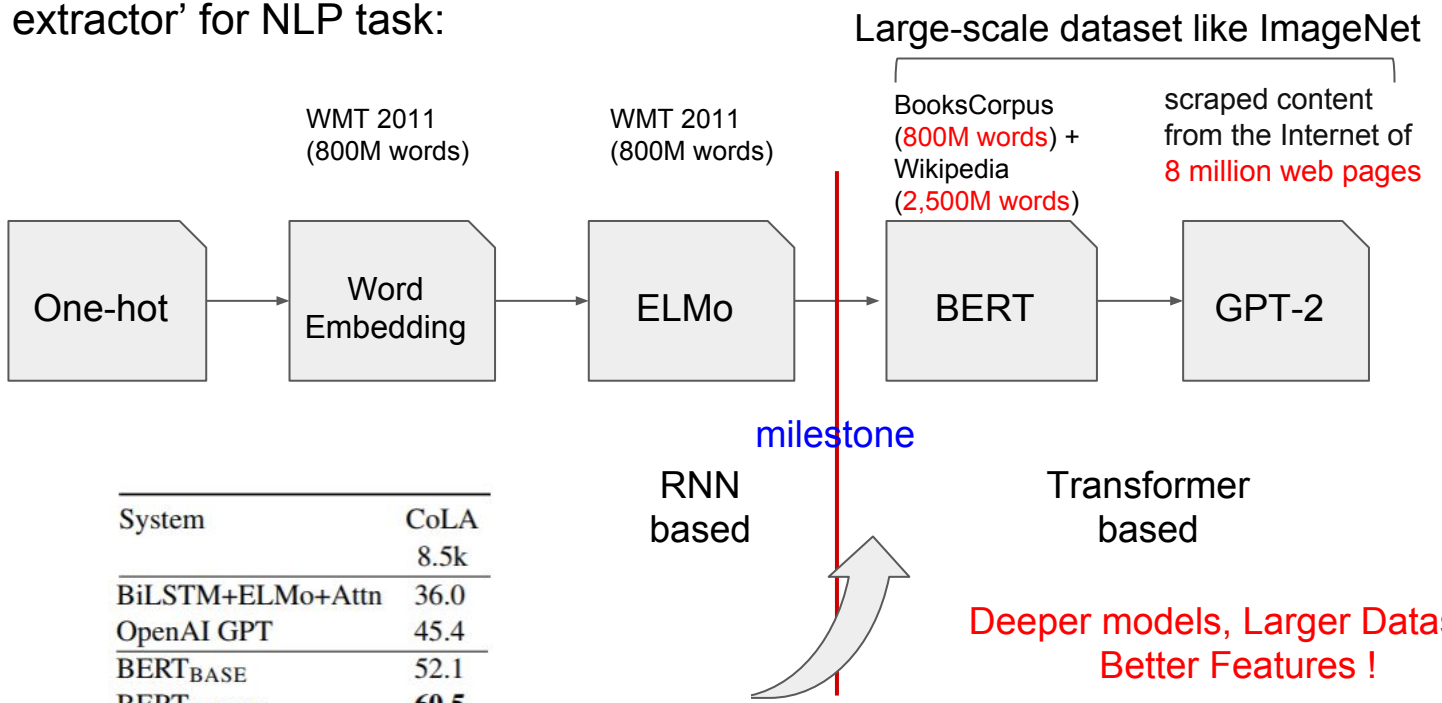




ImageNet-moment for NLP

- Can we build such a feature extractor to power various NLP tasks?
- For NLP, features are hidden vectors for each word.

Development of pretrained
'feature extractor' for NLP task:

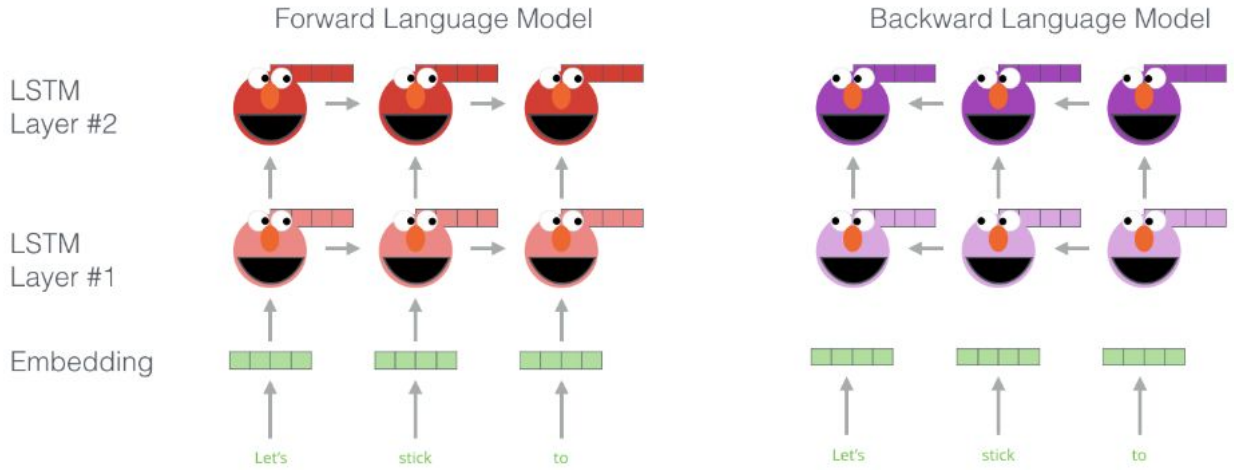


System	CoLA
	8.5k
BiLSTM+ELMo+Attn	36.0
OpenAI GPT	45.4
$\text{BERT}_{\text{BASE}}$	52.1
$\text{BERT}_{\text{LARGE}}$	**60.5**

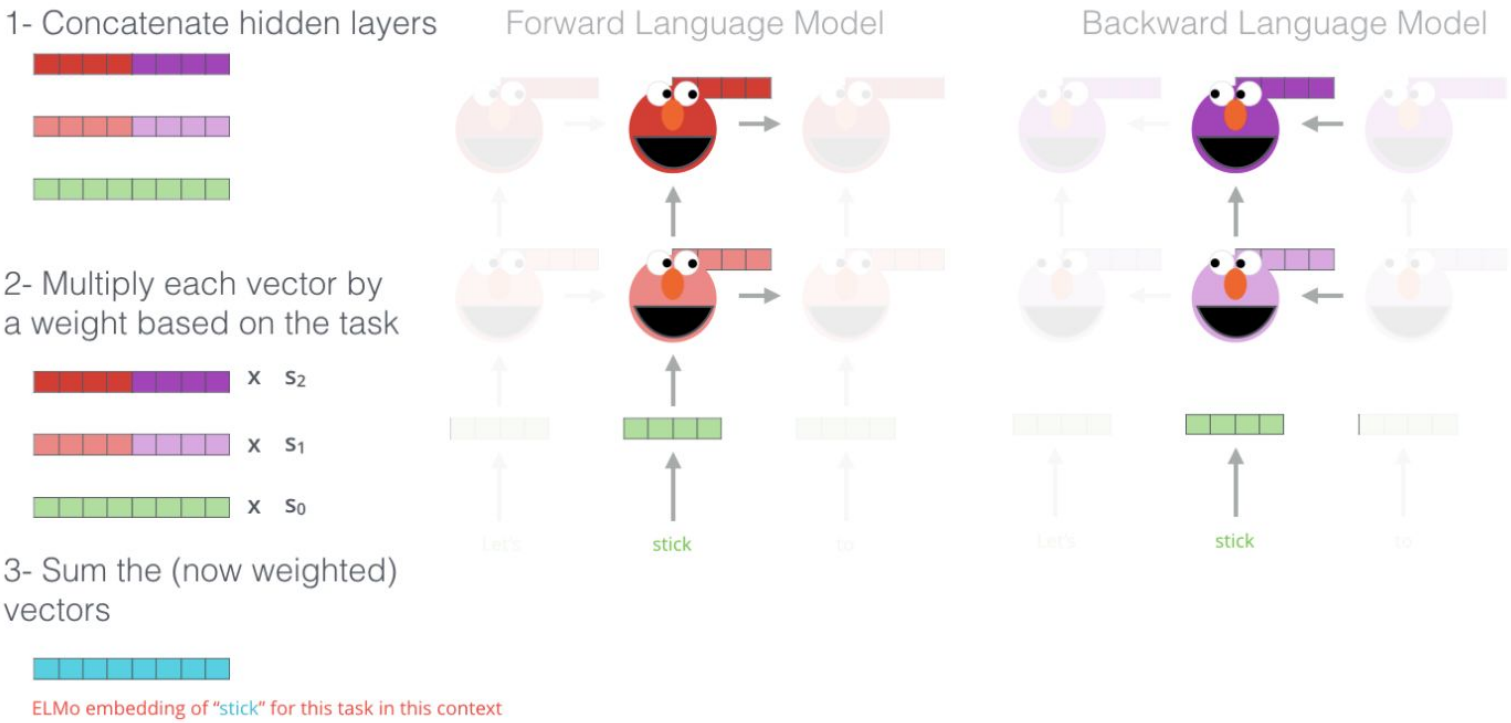
ELMo (Embeddings from Language Models)

- Word Embedding: Each word corresponds to a vector.
- In reality, a word may have different meanings depending on where it's used.
  - The "bank" on the other end of the street was robbed.
  - We had a picnic on the "bank" of the river.
- Instead of using a fixed embedding for each word, ELMo looks at the entire sentence before assigning each word an embedding. e.g. "The cat sat on the mat."

Train a model to take a part of sentence (say, the first n words) and predict the next word.



ELMo (Embeddings from Language Models)



Recap: Limitation of RNN

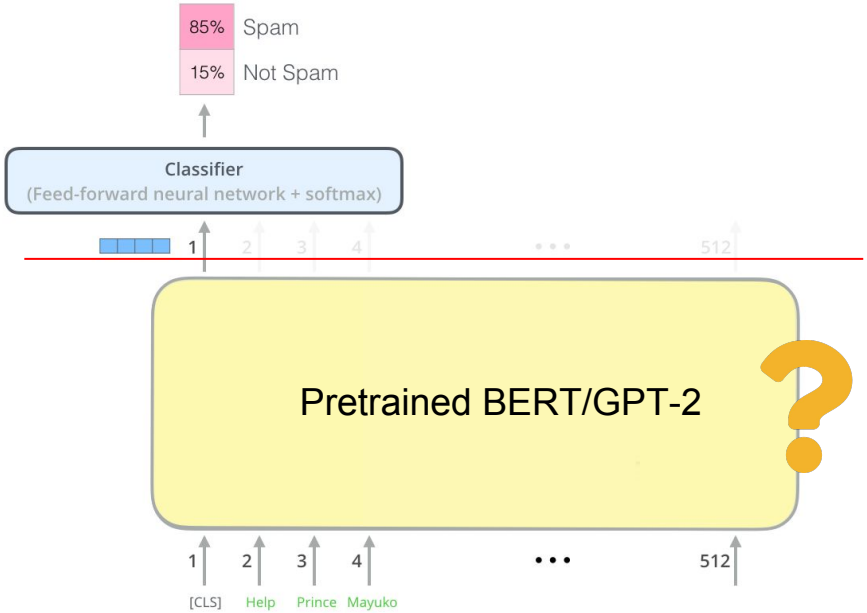
- RNN encodes relationship among words with hidden states in a recurrent way.
  - Hard to encode long distance relationship, saying the relationship between the first word and the last word in a long sentence because of gradient vanish.
    - e.g. “He” doesn't have very much confidence in “himself”.
    - e.g. “She” doesn't have very much confidence in “herself”.

- Transformer encodes relationship between words via matrix multiplication.

Use BERT and GPT as Feature Extractor

- Both BERT (from Google) and GPT (OpenAI) are general purpose pretrained Feature Extractors trained on enormous amounts of text data.
- They can then be fine-tuned on small-data NLP tasks (like question answering), resulting in substantial accuracy improvements compared to training on these datasets from scratch.

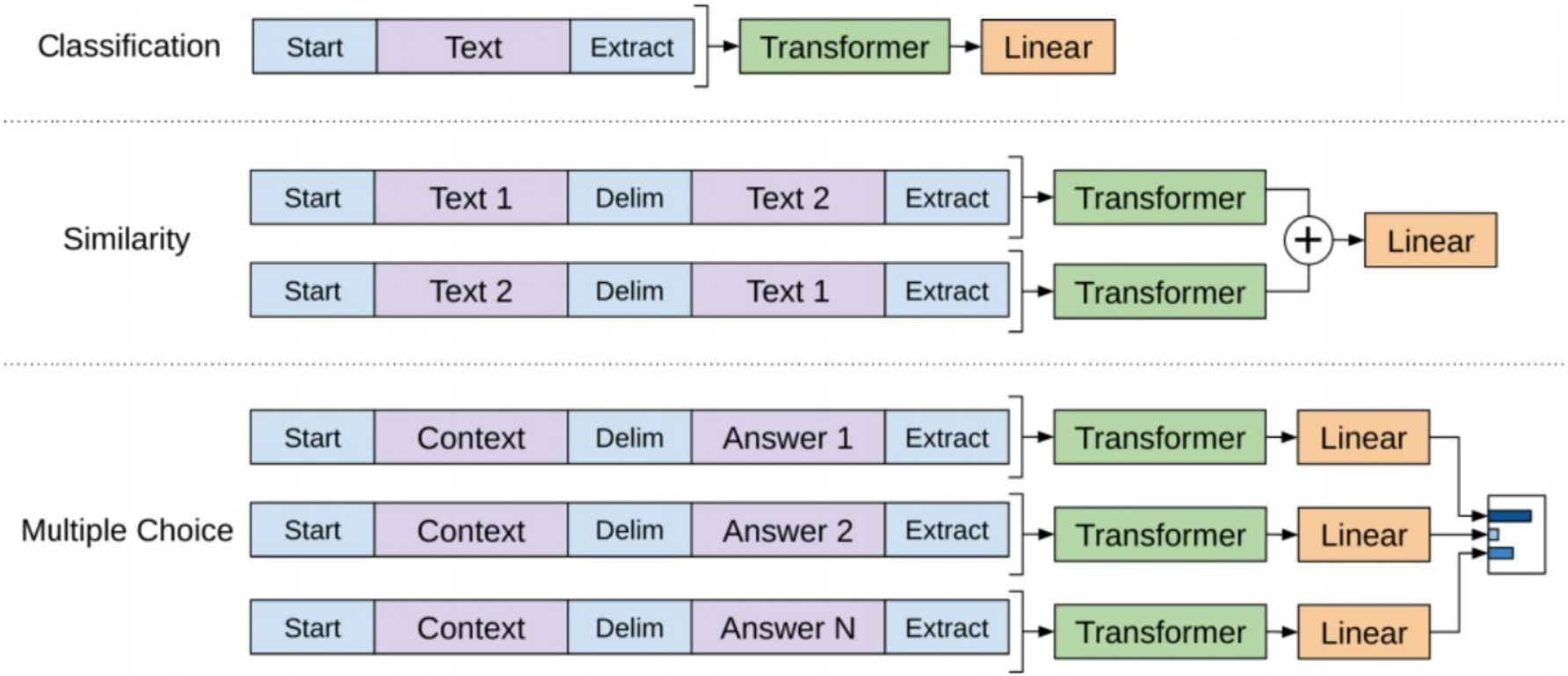
Additional classifier for your own tasks



The pretrained feature extractor

Based on
Transformer

More Use-cases

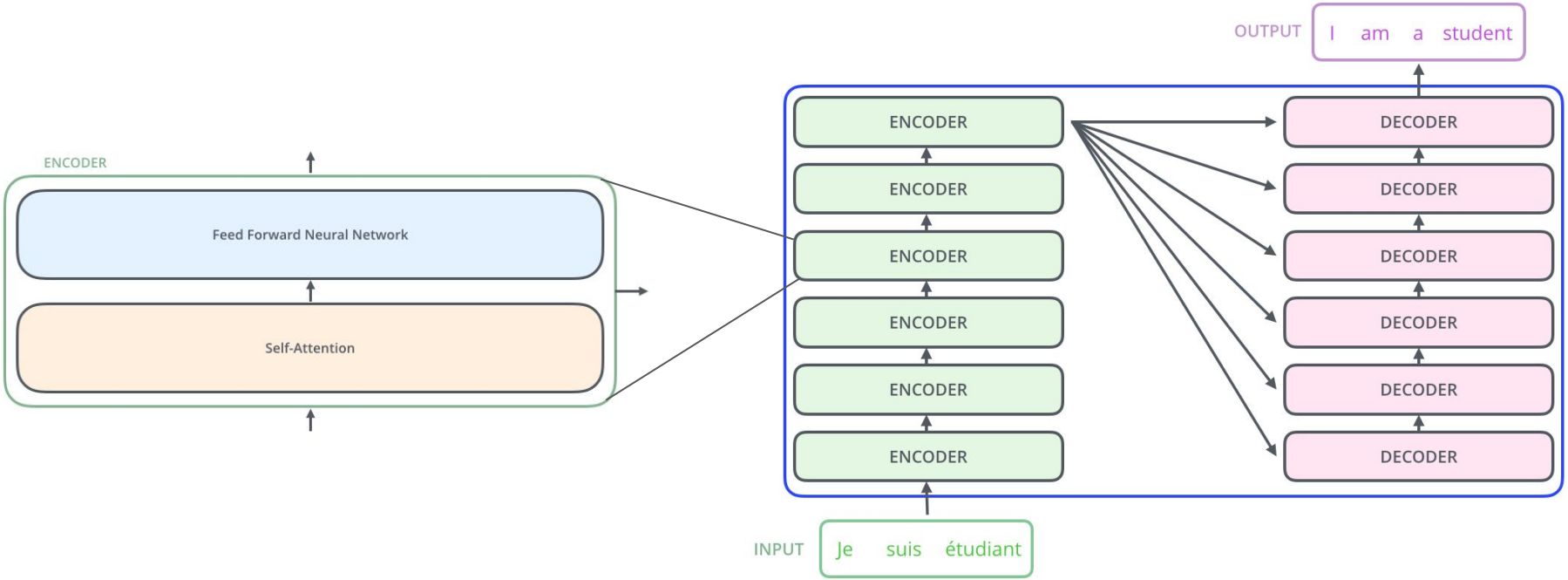


Example of Paragraph Generation from GPT-2

SYSTEM PROMPT (HUMAN-WRITTEN)	*A train carriage containing controlled nuclear materials was stolen in Cincinnati today. Its whereabouts are unknown.*
MODEL COMPLETION (MACHINE-WRITTEN, FIRST TRY)	The incident occurred on the downtown train line, which runs from Covington and Ashland stations. In an email to Ohio news outlets, the U.S. Department of Energy said it is working with the Federal Railroad Administration to find the thief. “The theft of this nuclear material will have significant...

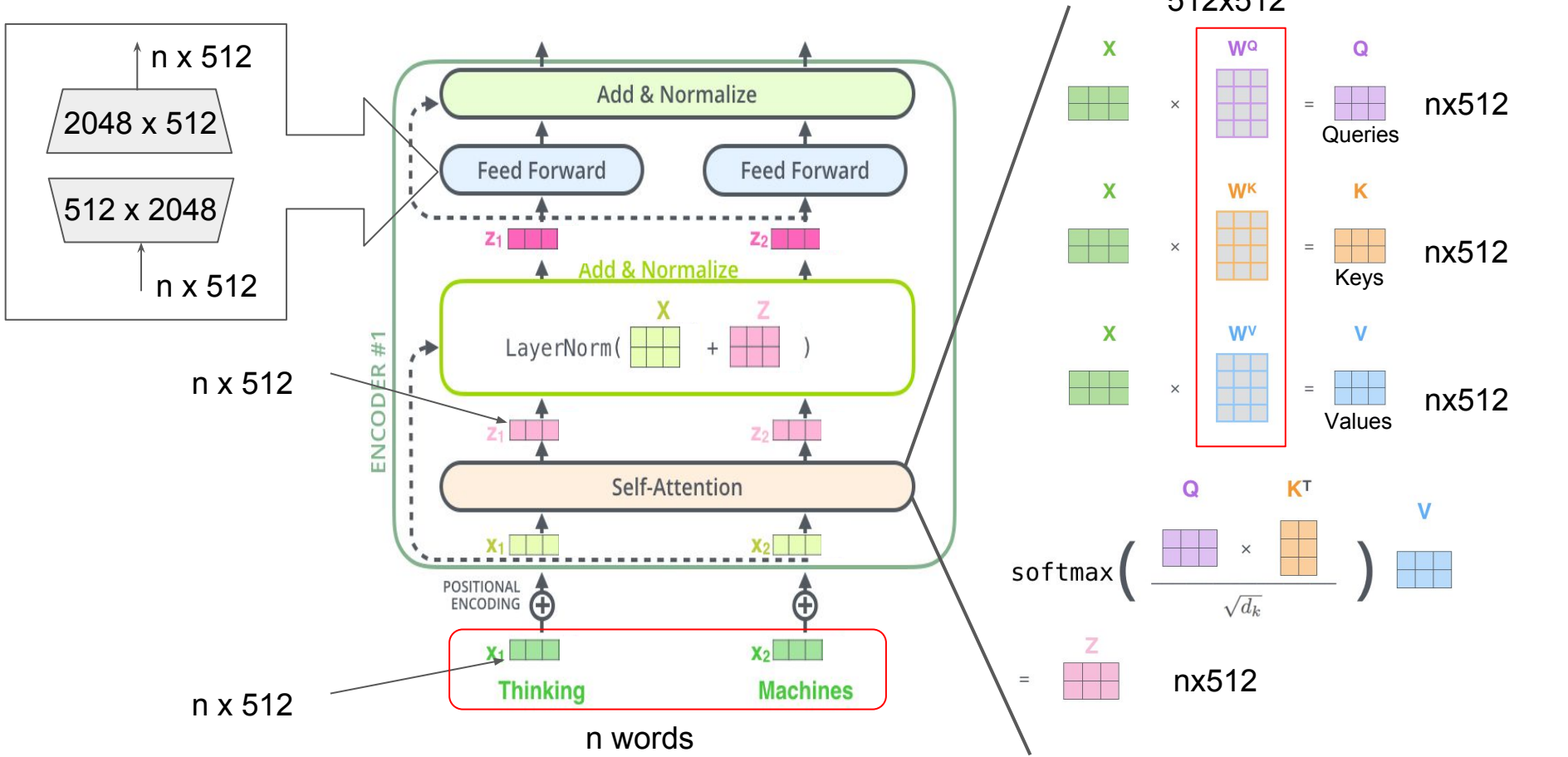
Transformer

- A transformer has an encoding component, a decoding component, and connections between them.
- Both the encoding and decoding components are the stack of encoder units and decoder units.
- Encoder units have the same architecture with different weights.
- Decoder units have a similar architecture to encoder units.

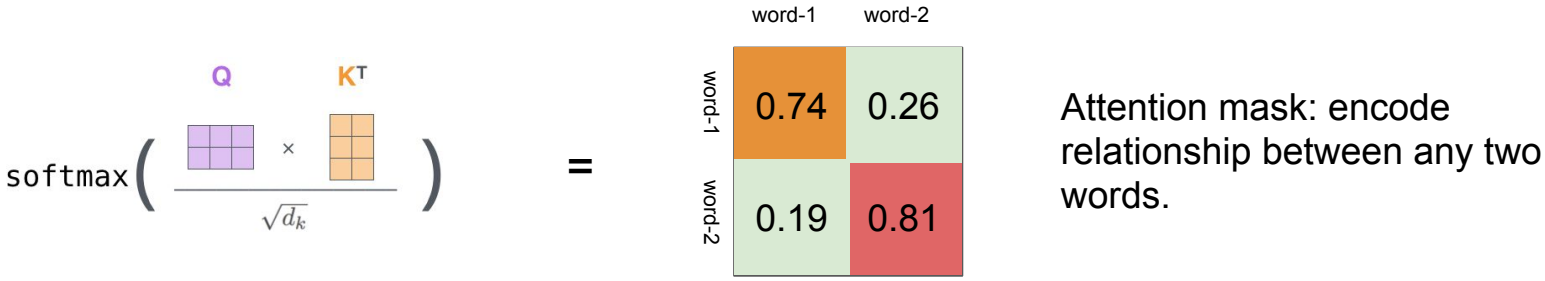


Configuration of Transformer (Encoder)

Here n is 2 for illustration



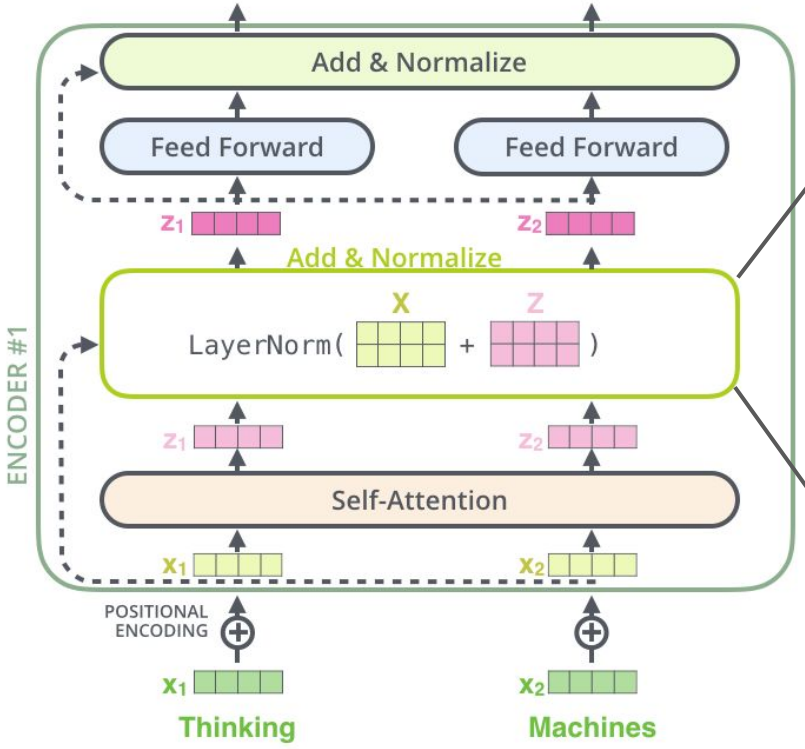
Self-Attention Layer



Attention mask: encode relationship between any two words.

- RNN encodes relationship among words with hidden states in a recurrent way.
  - Hard to parallel computation.
  - Hard to encode long distance relationship, saying the relationship between the first word and the last word in a long sentence because of gradient vanish.
- The transformer encodes relationship between any two words via matrix multiplication.
  - Core computation is just matrix multiplication which can be paralleled easily.
  - Distance between any two words is always 1 no matter how long your sentence is. It is easier to encoder long distance relationship.

Add and Normalization Layer



In this case, M=4.

$$Y = X + Z$$

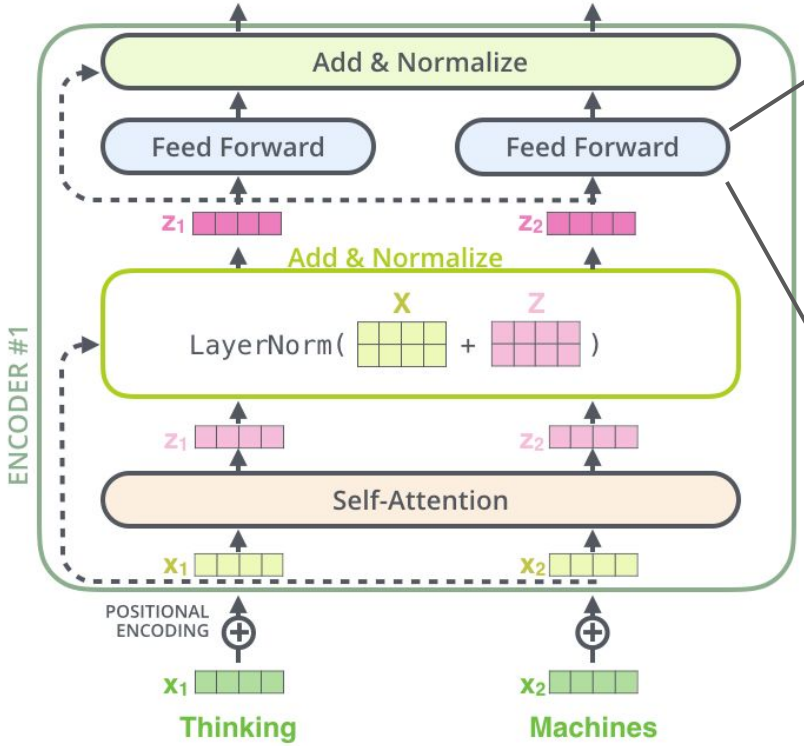
$$\mu = \frac{1}{M}\left(Y_1 + Y_2 + Y_3 + Y_4\right)$$

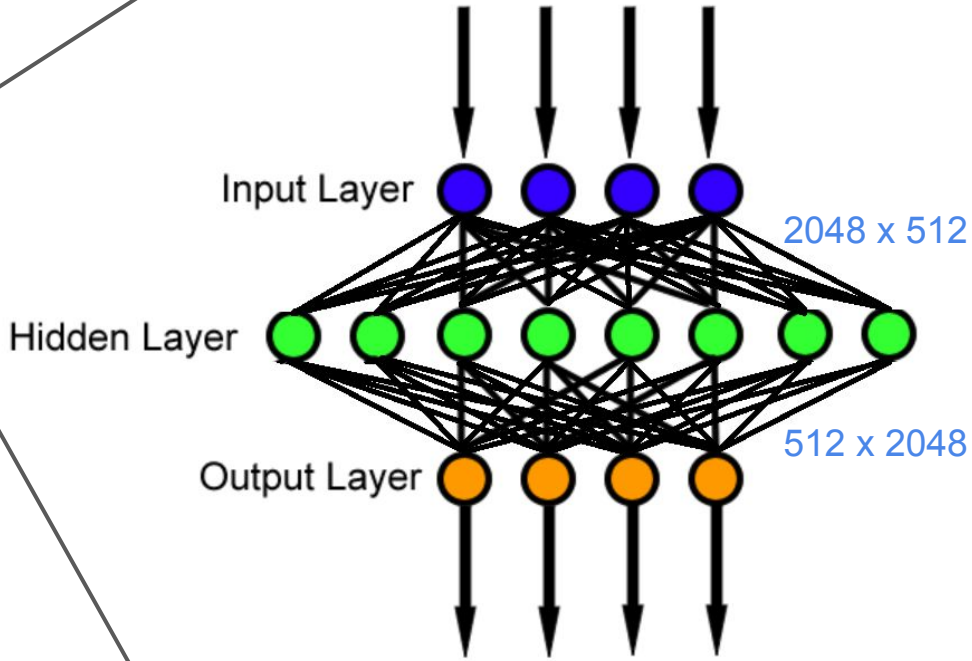
$$\sigma = \sqrt{\frac{1}{M}\sum_{i}^{M}\left(Y_i - \mu\right)^2}$$

$$Z_i = \frac{Y_i - \mu}{\sigma}$$

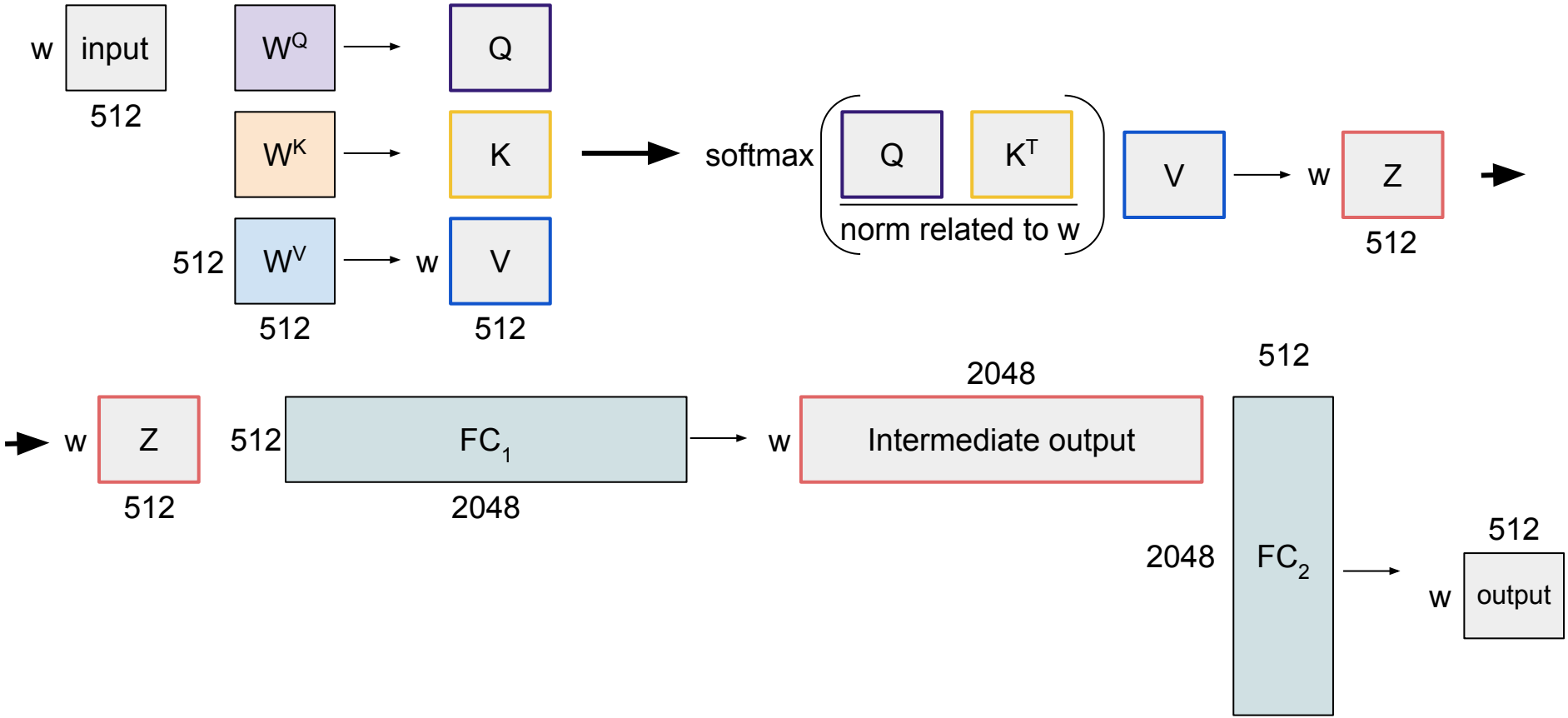
$$Z = \text{Concat}(Z_i, \ldots, Z_i)$$

Feed-Forward Layer

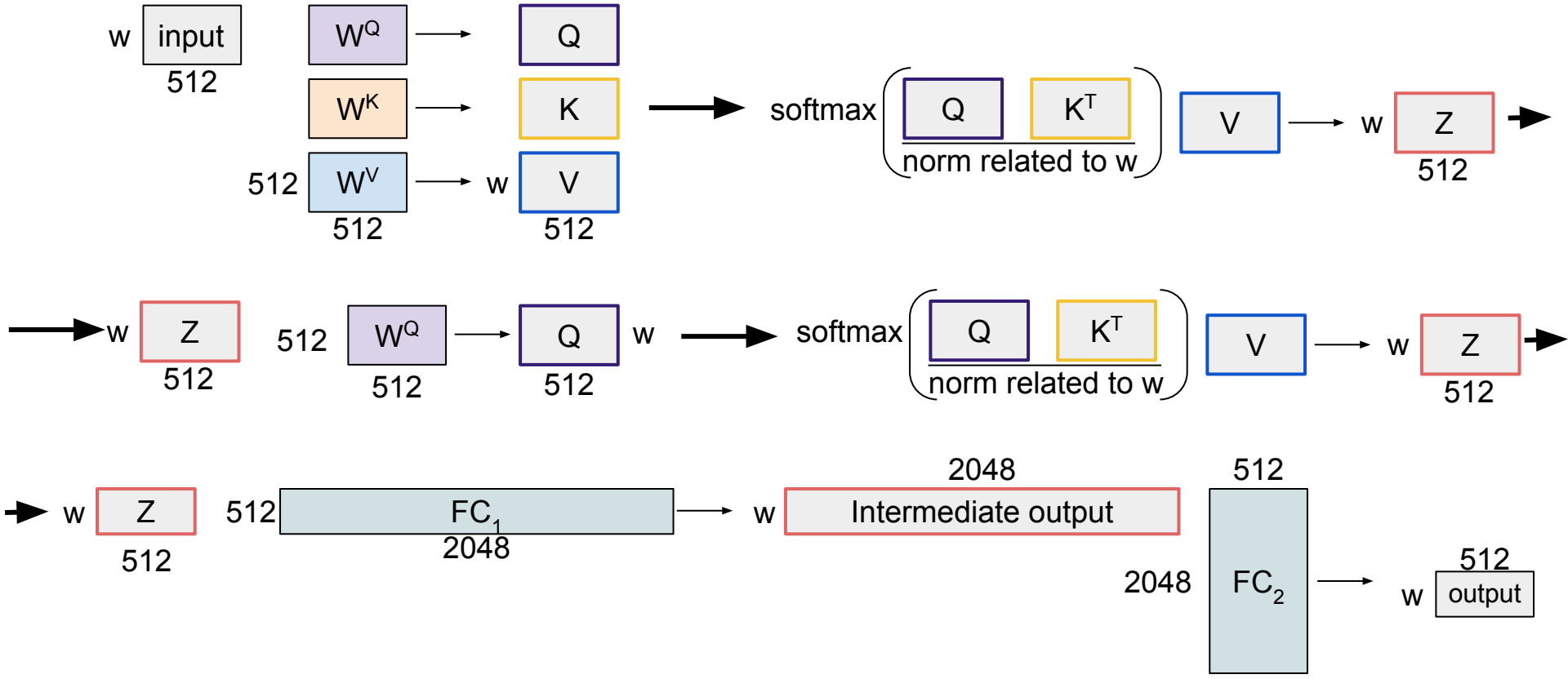




Computational Diagram: Transformer Encoder



Computational Diagram: Transformer Decoder



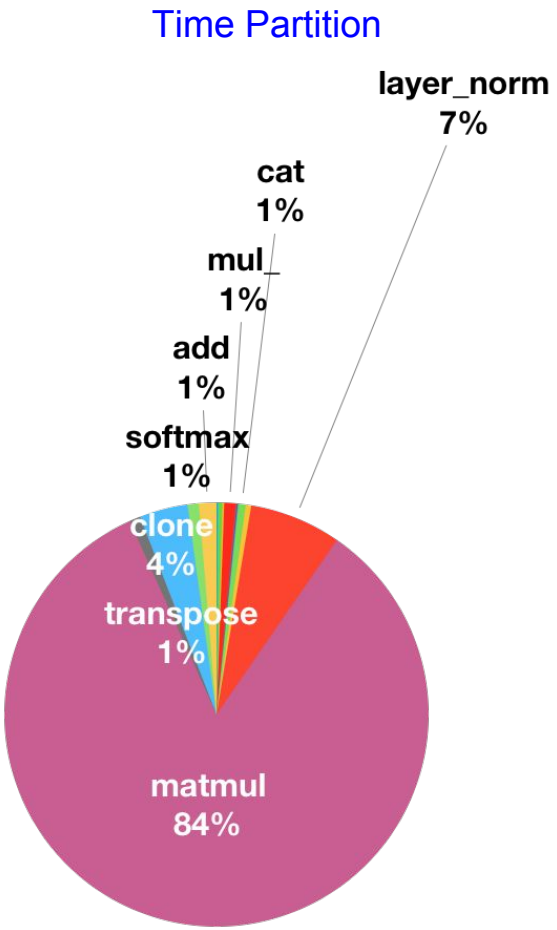
Latency of Transformer

- Translate a sentence which has 55 words.
- Elapsed Time: ~ 0.97 sec
- Test on one Nvidia 1080 Ti with official TensorFlow implementation

- Efficient computation for Transformer is crucial
  - TPU from Google
  - Meastro from Prof. Kung's lab

Q: For complicated language model presented in this paper (340M), if we implement it to the federated learning, what are the strategies to make the learning possible on resources constrained device like cellphone or embedded devices?

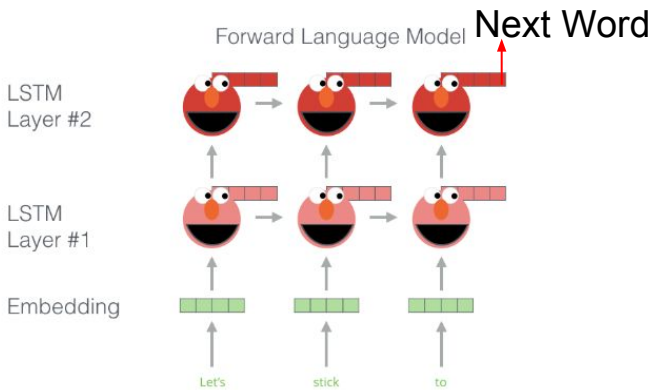
A: It is an open problem for now. Hope we can solve this with our Meastro design (hardware) and network slimming (software) .

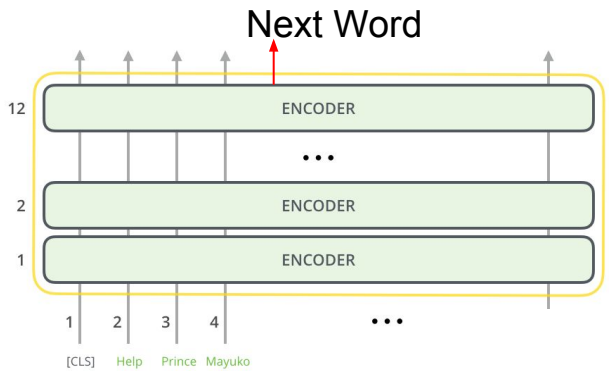


Recap: How to Train BERT? Language Model ?

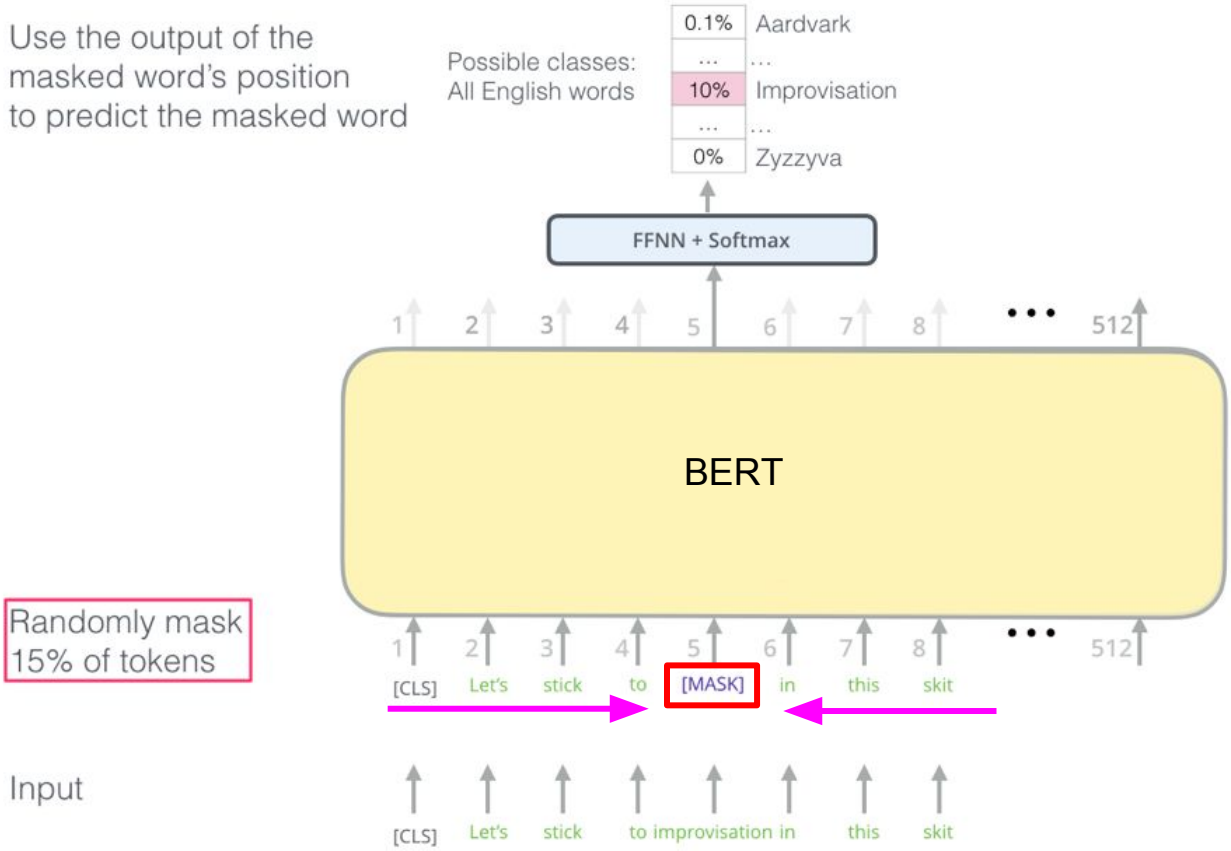
- Two keywords:
  - Large size model
  - Large scale training data
- Large size model
  - BERT uses the Transformer as base architecture
  - BERT(base) has 12 transformer units stacked. The hidden size is 768, and the size of feed forward layer is 768-3078-768. Total number of parameters is 65 million.
- Large scale training data
  - BooksCorpus (800M words) + Wikipedia (2,500M words)

Language Model



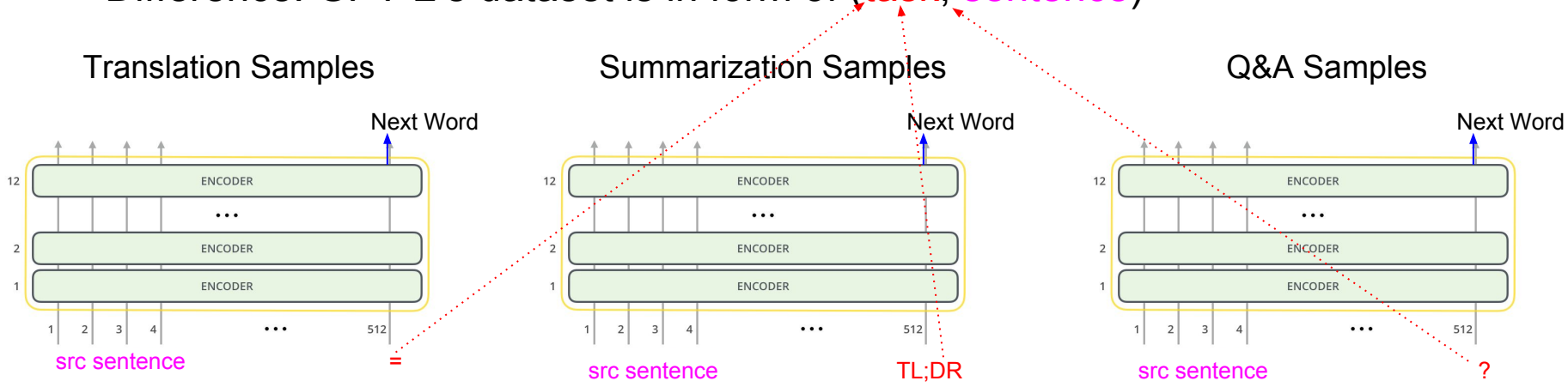


Recap: How to Train BERT? Bi-Direction



GPT-2

- It seems like that OpenAI doesn't believe in the Bi-Directional stuff from Google.
- OpenAI insists on using Language Model (predict the next word, so it isn't bidirectional) to train their GPT-2
- GPT-2 has 10x parameters and about 10x larger dataset than BERT.
- Difference: GPT-2's dataset is in form of (task, sentence)



Pass both your specific task and src sentence to GPT-2 without any fine tuning!

Questions

1. From your understanding of the Transformer, do you see any further use cases where the network architecture would lend itself particularly useful such as image processing?
    a. Yes! Please check Wang, Xiaolong, et al. "Non-local neural networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.
2. Finally, this is only tangentially related, but do you know whether people have found success with large-scale unsupervised representation of speech? It seems as if some of the developments driving recent successes in NLP could be applicable to speech.
    a. Both Transformer and RNN are designed to process sequential data including text, speech and so on, so you can use Transformer to process speech data for sure.
    b. When you process text data, you may represent each word by a 1x512 vector and then input them into Transformer. Similarly, you can also represent each speech segment (you can do this in both time-domain and frequency-domain) by a 1x512 vector.