

Transformer-based Pre-trained Models for Natural Language Processing

Team member: Eric Wang, Jacky Chen, Ziyang Guo, Zirong Huang

Image by [NLP-image-scaled.jpg](#)

Table Of Content

Intro

Pre-Training

Models

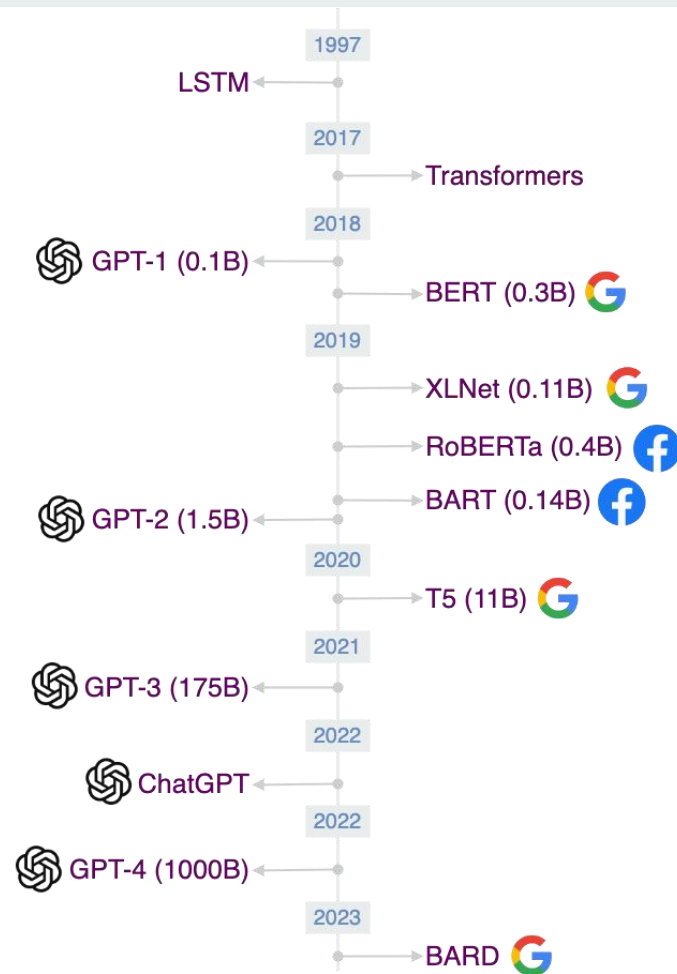
Fine tuning

Result

Overview (DL in NLP)

With the development of deep learning, various neural networks have been widely used to solve natural language processing tasks, including CNNs, RNNs and Attention mechanisms.

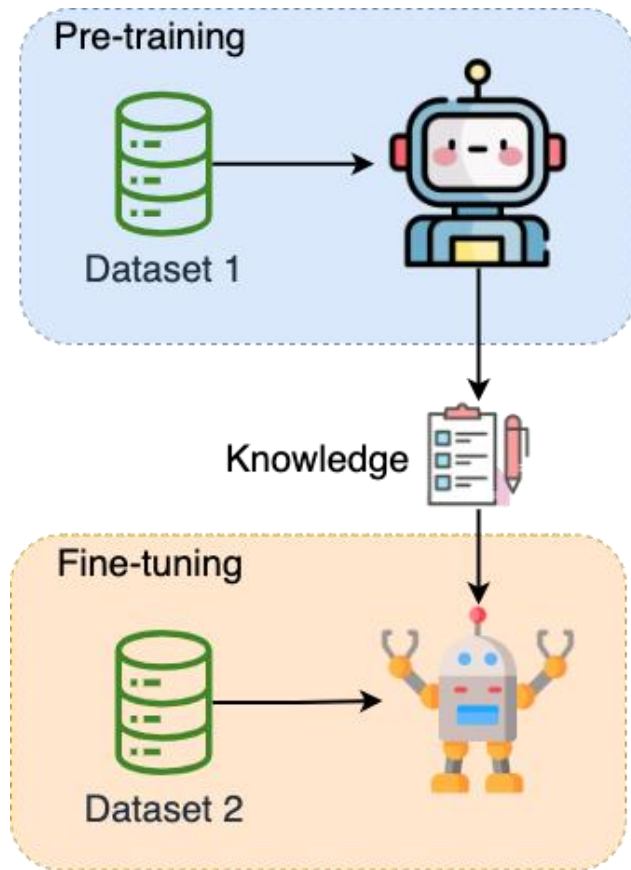
Recent substantial work shows that pre-trained models (PTMs), trained on large-scale unlabeled data, are able to learn universal language representations.



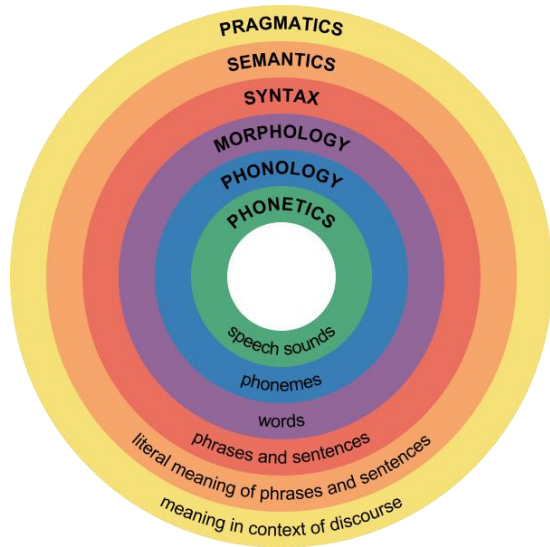
Pre-Trained Models (PTMs) & Transfer Learning

Advantages:

- Learns universal representations
- Better model initialization and faster convergence
- Helps to avoid overfitting on small data

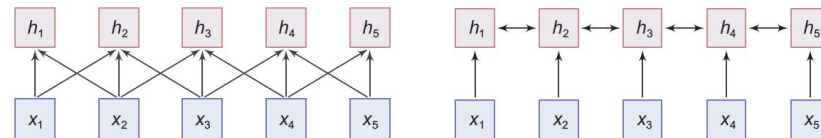


Contextual Embeddings

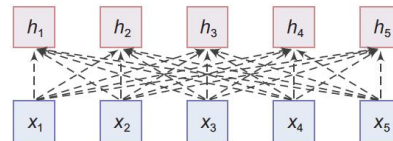


Linguistic Stack

- Non-contextual word embedding
 - Word2vec
 - Fail to model polysemous words
- Contextual word embedding
 - Sequential models: CNNs, RNNs

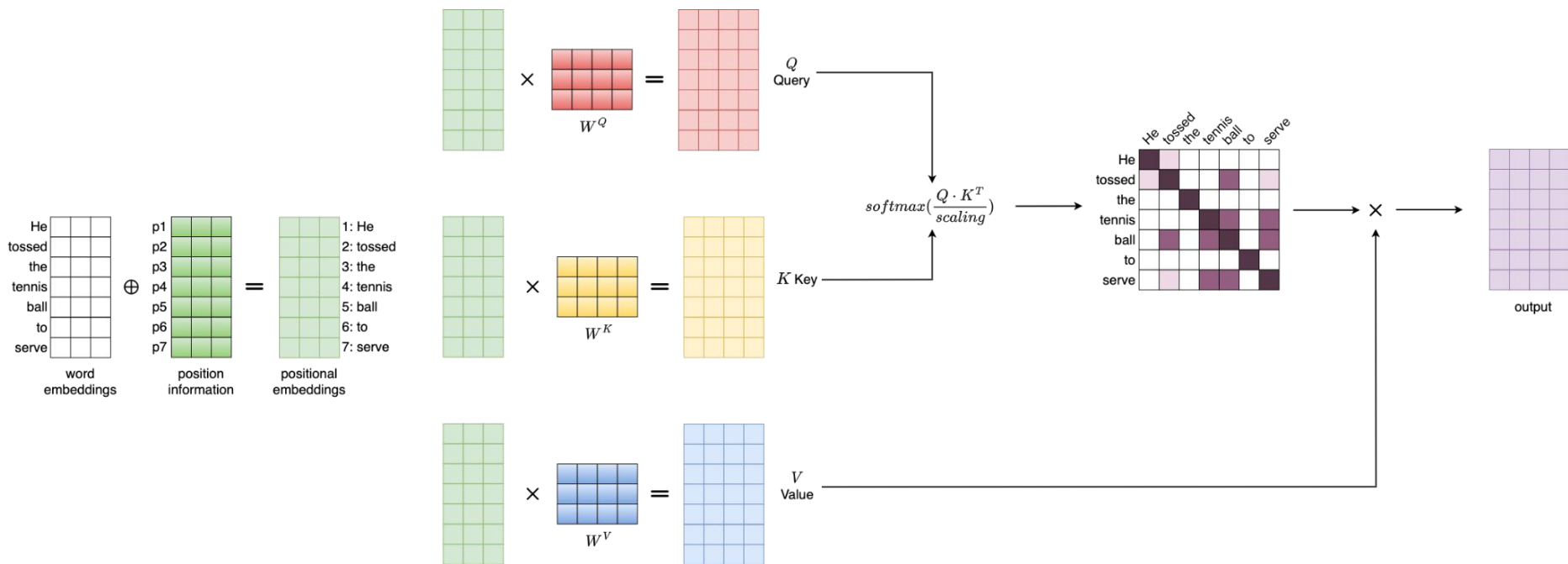


- Non-sequential models: Self-attention models



Scaled Dot-Product Attention

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



Scaled Dot-Product Attention

- Allows the network to “attend” to the parts of the input sequence that are relevant for predicting the next word
- Error can propagate more directly to the part of the network that produced those hidden states

Transformers

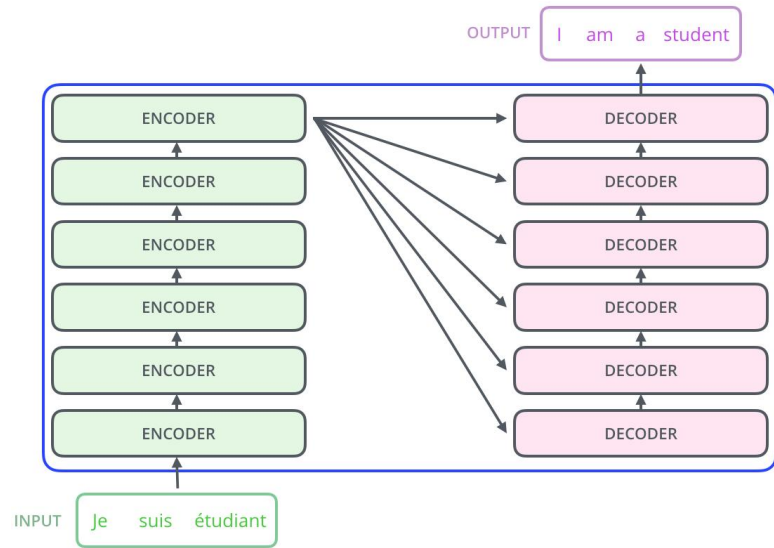
Advantages:

- More parallelizable than RNNs
- Requires less time to train
- Can directly model dependency of distant words

Disadvantages:

- Requires large training corpus and prone to overfitting on small datasets

Currently the Transformer has become the mainstream architecture of pre-trained models due to its power capacity

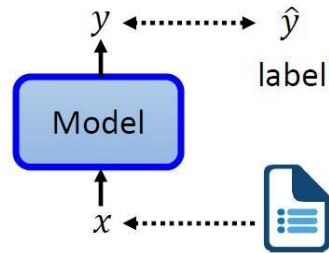




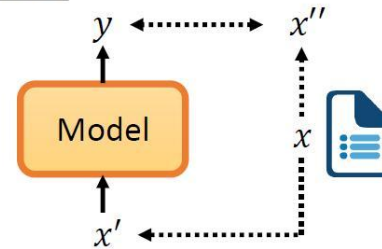
Pre-Training tasks

Self-supervised Learning

Supervised



Self-supervised



Pre-training Tasks

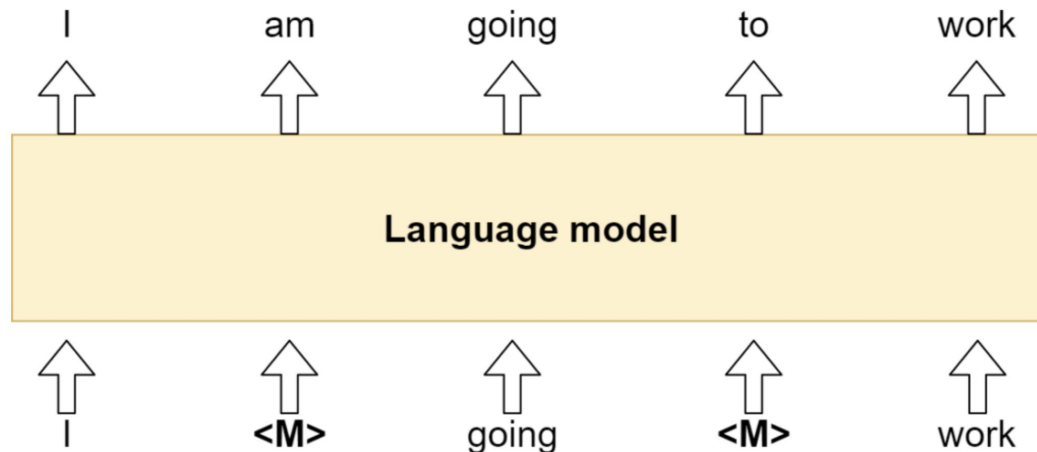
- Masked Language Model (MLM)
- Permuted Language Model (PLM)
- Denoising Autoencoder (DAE)
- Next Sentence Prediction (NSP)
- Others



Masked Language Model (MLM)

Cloze task.

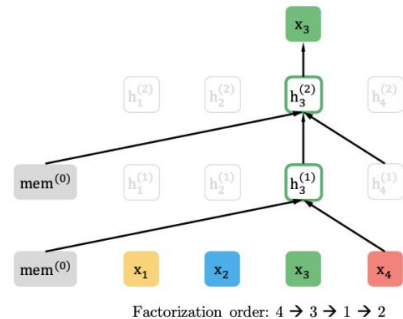
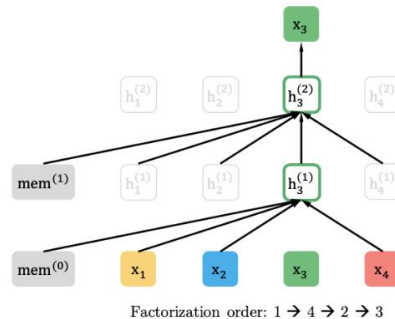
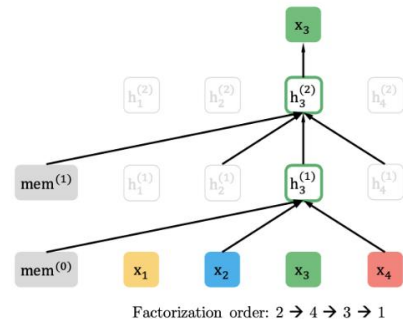
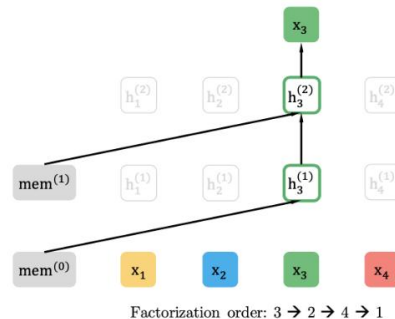
Predict the missing words in a sentence based on the context provided by the surrounding words.



Permuted Language Model (PLM)

A language modeling task on a random permutation of input sequences.

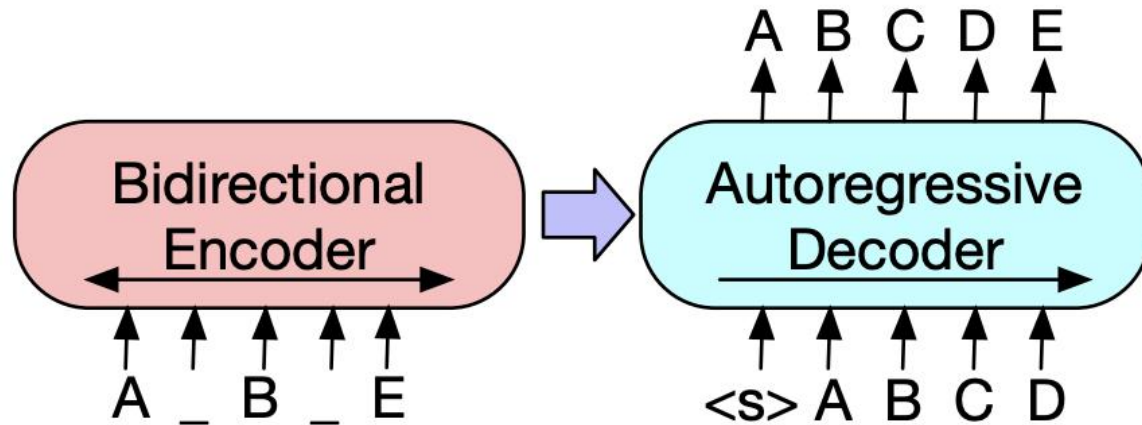
Randomly shuffling the input sequence order can make the context token of the predicted token appear before the predicted token.





Denoising Autoencoder (DAE)

Takes a partially corrupted input and aims to recover the original undistorted input.



Next Sentence Prediction (NSP)

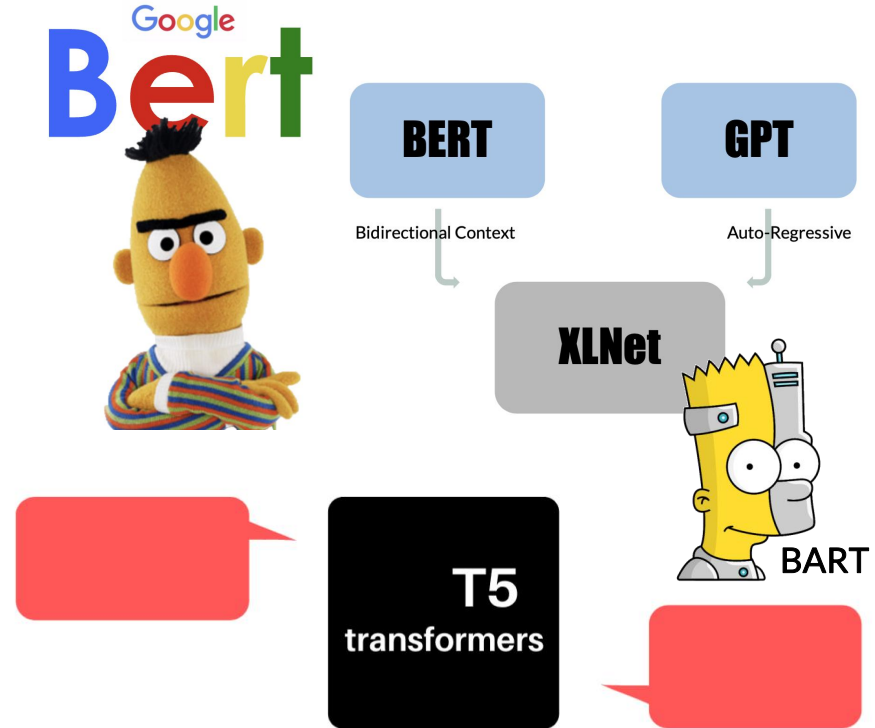
A "next sentence prediction" binary classification task.

Sentence 1	Sentence 2	Next Sentence?
I have a class	I will be back by 6	<input checked="" type="checkbox"/>
I have a class	Zebra is a animal	<input type="checkbox"/>

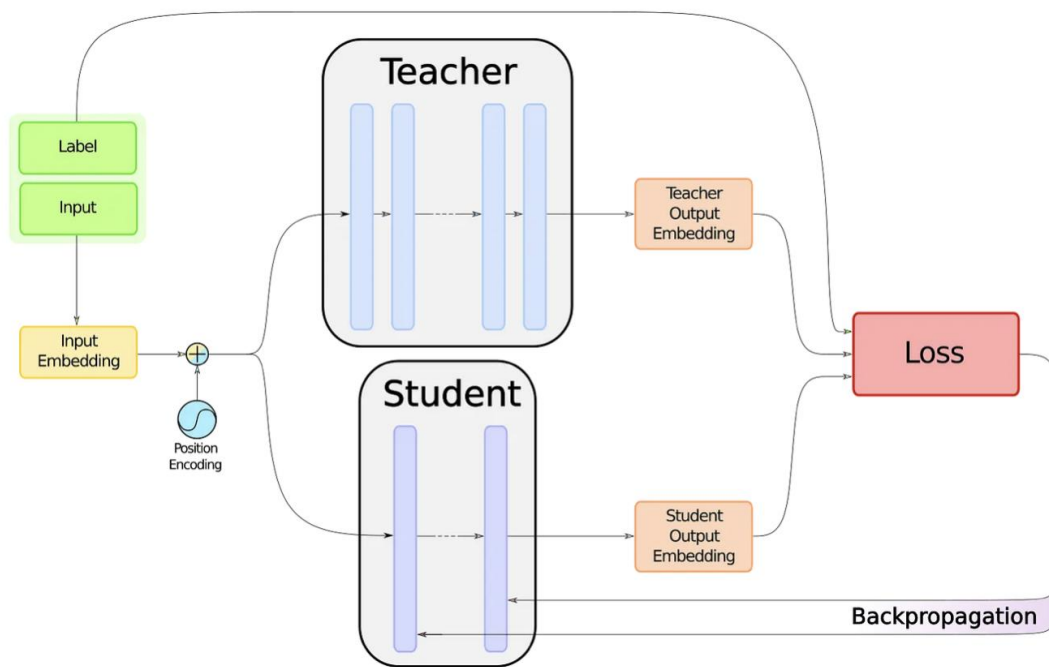


Used Models

- 01 | DistilBERT (Distilled Bidirectional Encoder Representations from Transformers)
- 02 | XLNet (eXtreme Language understanding Network)
- 03 | T5 (Text-to-Text Transfer Transformer)
- 04 | BART (Bidirectional and Auto-Regressive Transformers)



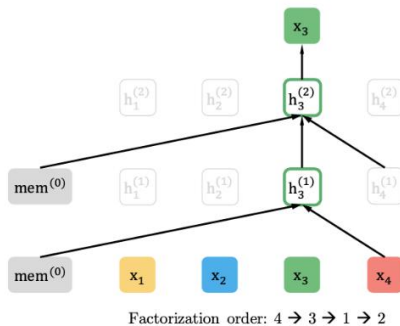
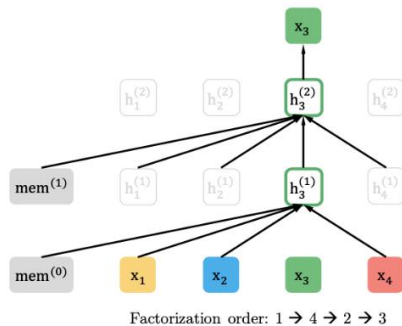
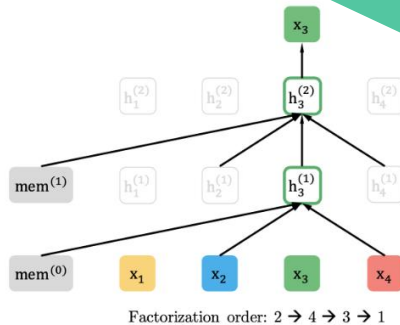
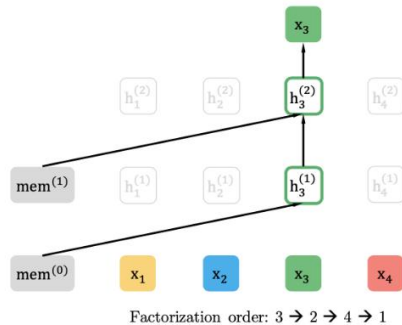
- DistilBERT (a compressed version of the BERT)



- Encoder only
- Pre-trained :
 - Masked Language Model
 - Next Sentence Prediction

- XLNet (eXtreme Language understanding Network)

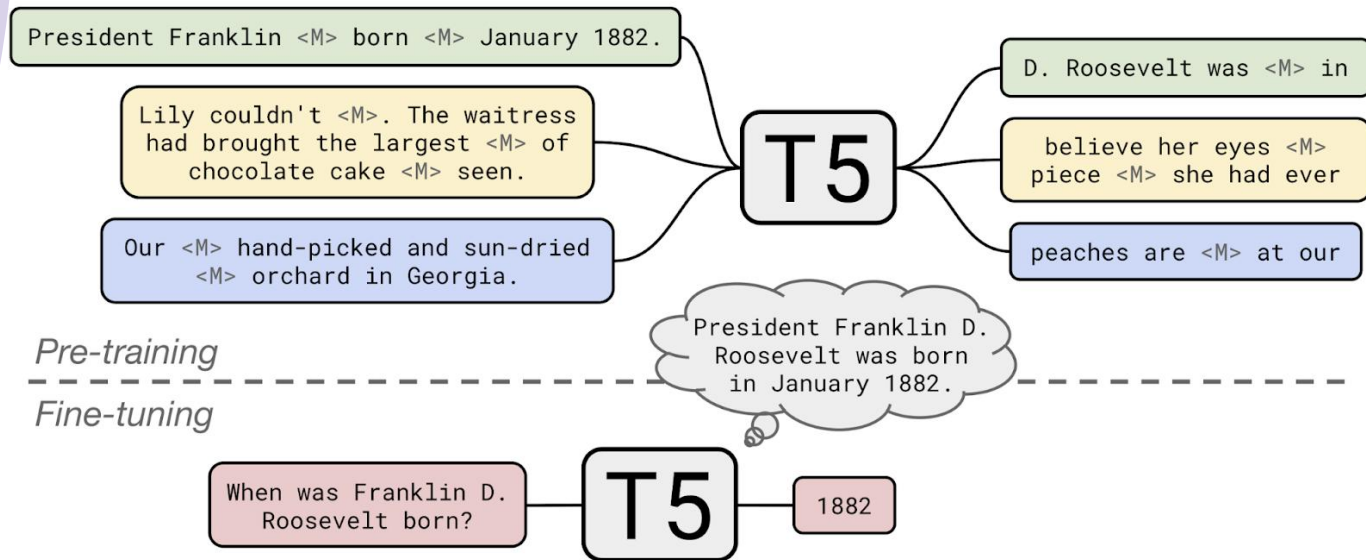
- Encoder only
- Pre-trained :
 - Permuted Language Model



- T5 (Text-to-Text Transfer Transformer)

Encoder - decoder

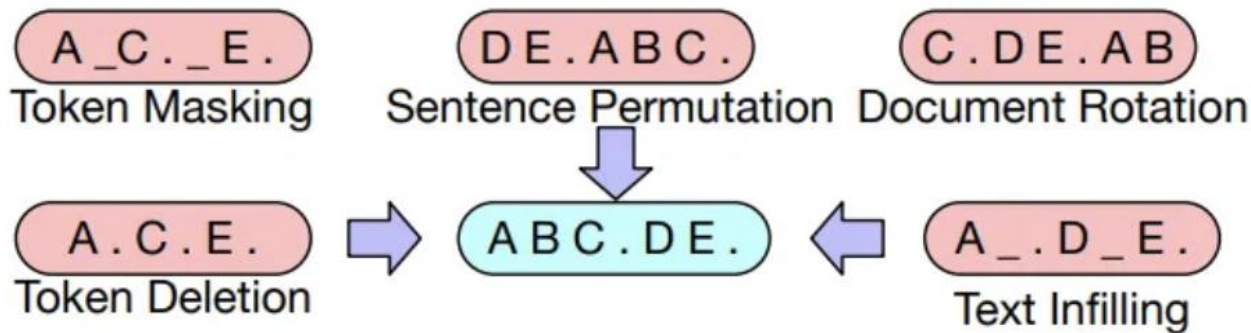
Pre-trained : Masked Language Model




- BART (Bidirectional and Auto-Regressive Transformers)

Encoder - decoder

Pre-trained : Denoising Autoencoder



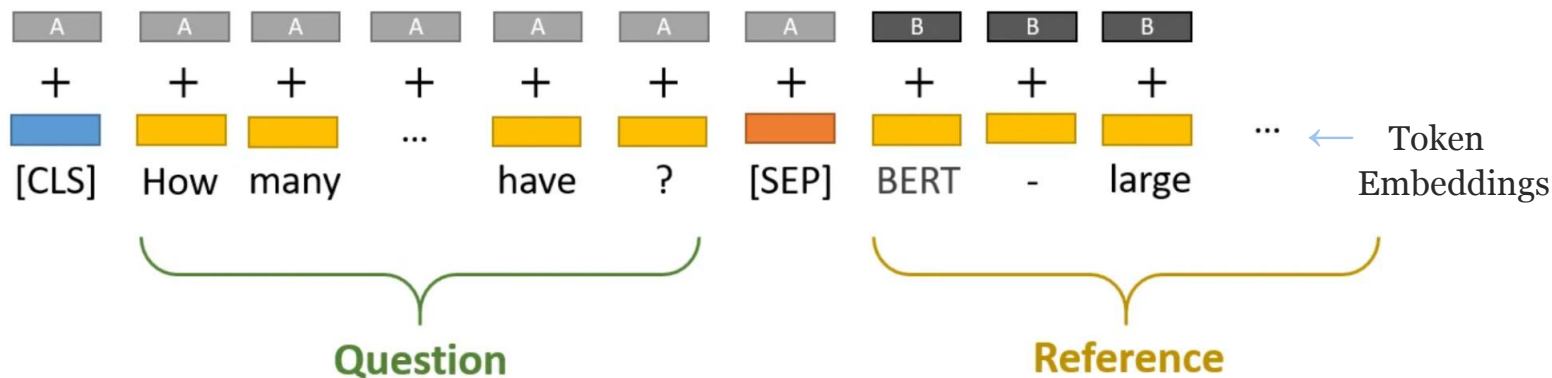


PTMs downstream tasks (Fine Tuning)

- Language Translation
- Text Summarization
- Question Answering
- Others

BERT

Q&A Example



Question: How many parameters does BERT-large have?

Reference Text: BERT-large is really big... it has 24 layers and an embedding size of 1,024, for a total of 340M parameters! Altogether it is 1.34GB, so expect it to take a couple minutes to download to your Colab instance.

The two pieces of text are separated by the special [SEP] token.

Reference by [1*W6tR3aYI2HNuociJFKGfIQ.png](#)

```
{
  "data": [
    {
      "paragraphs": [
        {
          "context": "The Normans (Norman: Nourmands; French:  
Normands; Latin: Normanni) were the people who in the  
10th and 11th centuries gave their name to Normandy,  
a region in France. ",
          "qas": [
            {
              "answers": [
                {
                  "answer_start": 159,
                  "text": "France"
                }
              ],
              "id": "56ddde6b9a695914005b9628",
              "is_impossible": false,
              "question": "In what country is Normandy located?"
            }
          ]
        }
      ],
      "title": "Normans"
    }
  ],
  "version": 2
}
```

Dataset SQuAD 2.0

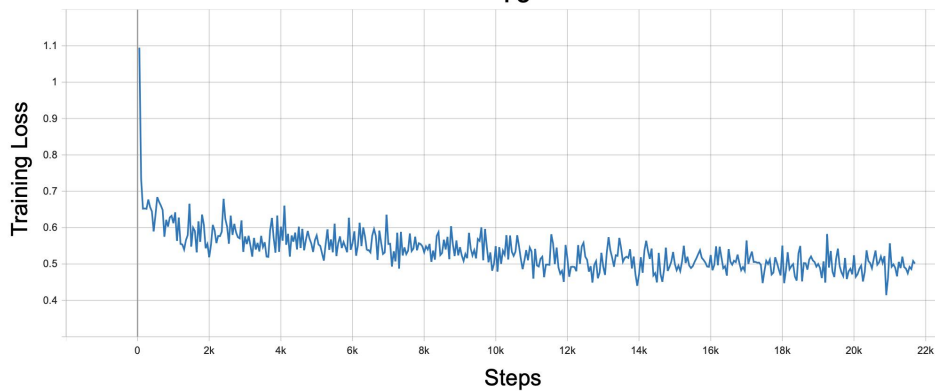


Experiment Result

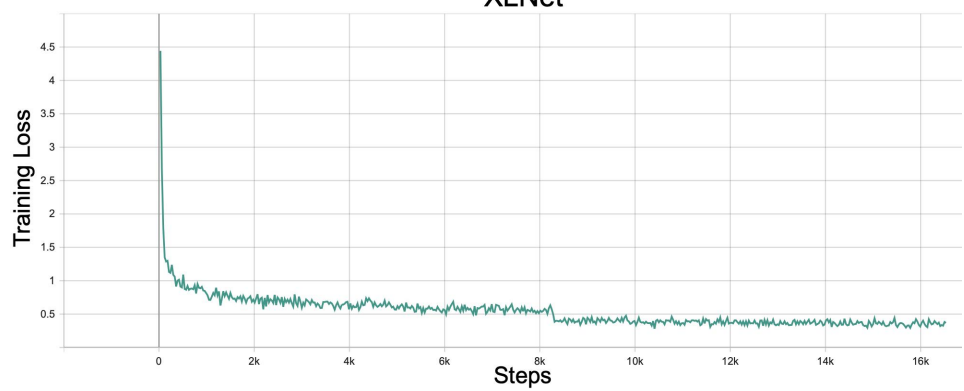
On above models



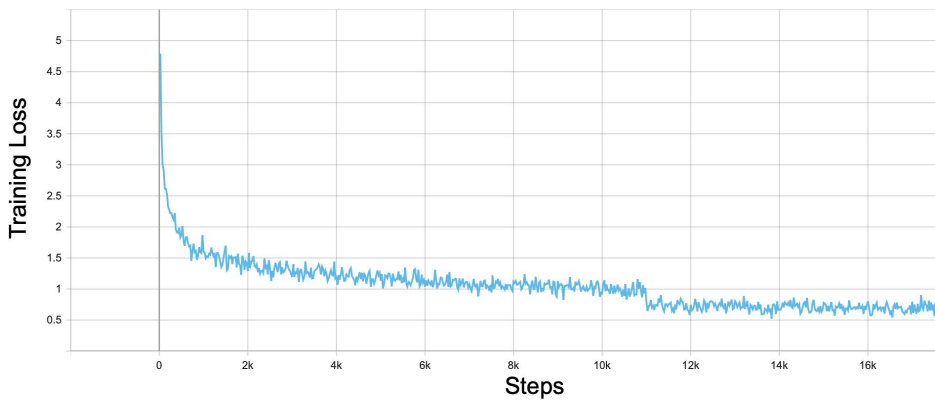
T5



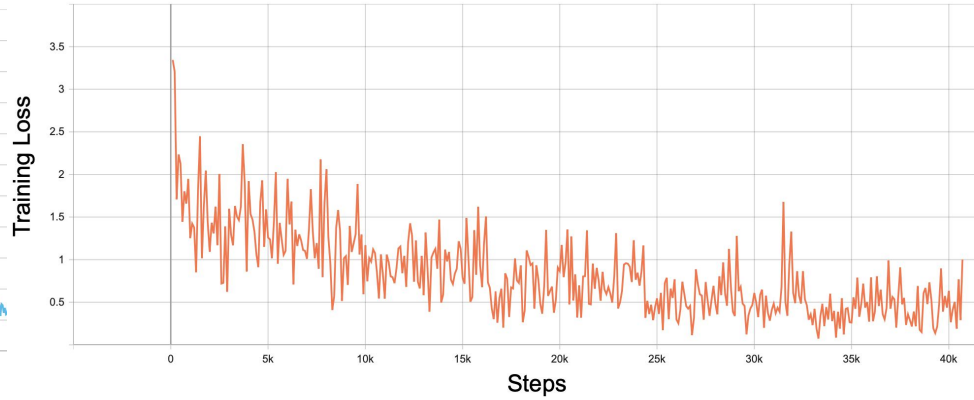
XLNet

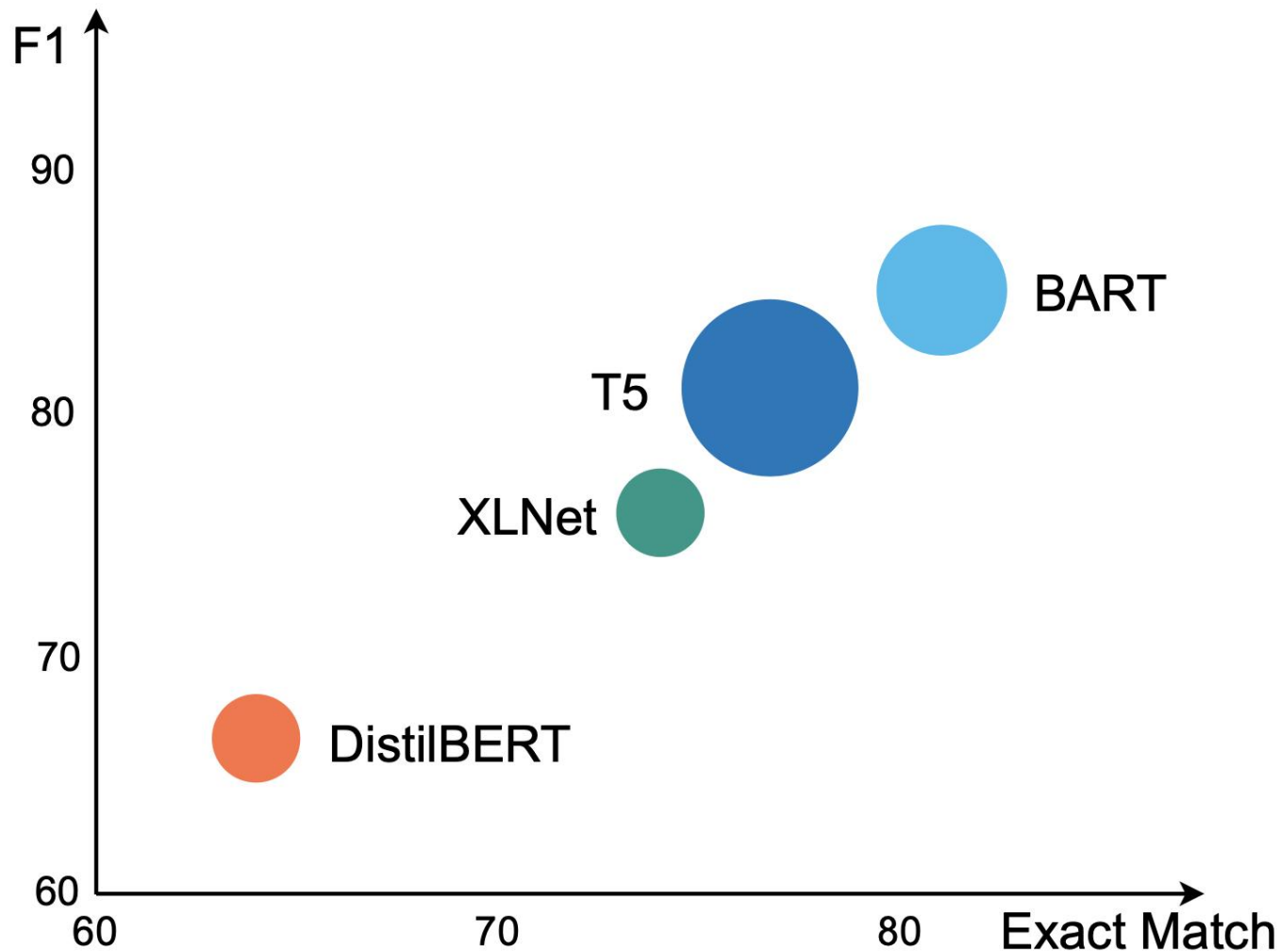


BART



DistilBERT





```
context = r"""
CS 479 involves an introduction to neural network methods, with some
discussion of their relevance to neuroscience, simple neuron models
and networks of neurons. Topics include training feedforward networks,
learning using the backpropagation of errors, unsupervised learning
methods, optimal linear decoding, recurrent neural networks and
convolutional neural networks. Advanced topics of the course may cover
adversarial inputs and biologically plausible learning methods."""

result = question_answerer(
    question = "What high level knowledge can I learn in cs479?",
    context = context)

print(f"Answer: '{result['answer']}'")
```

Answer: 'adversarial inputs and biologically
plausible learning methods.'



Thank you.

