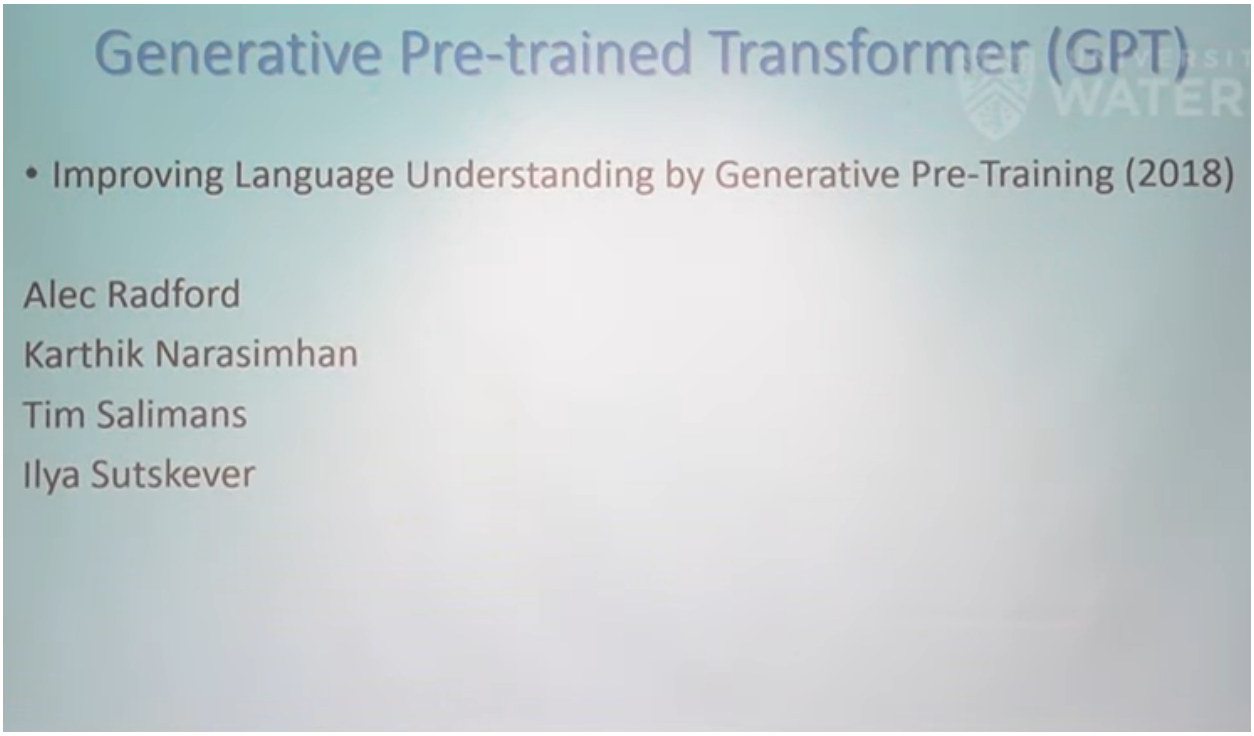
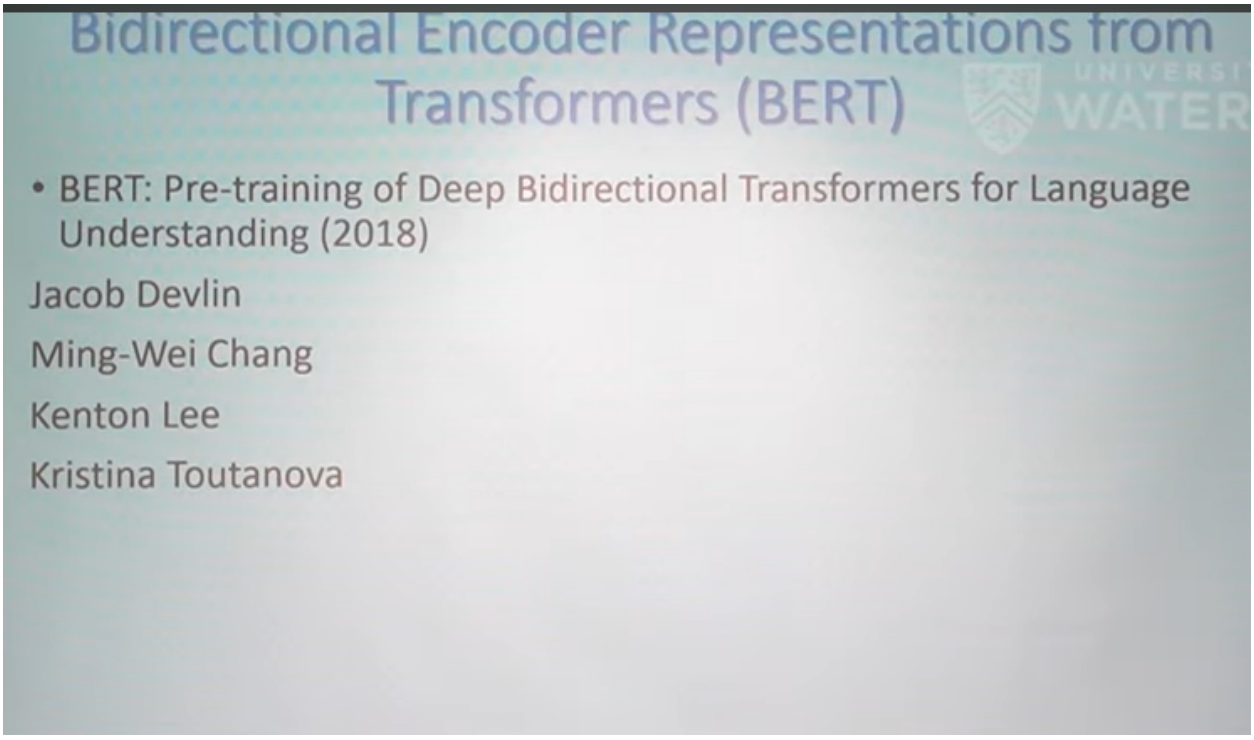
Lecture 11: BERT and GPT

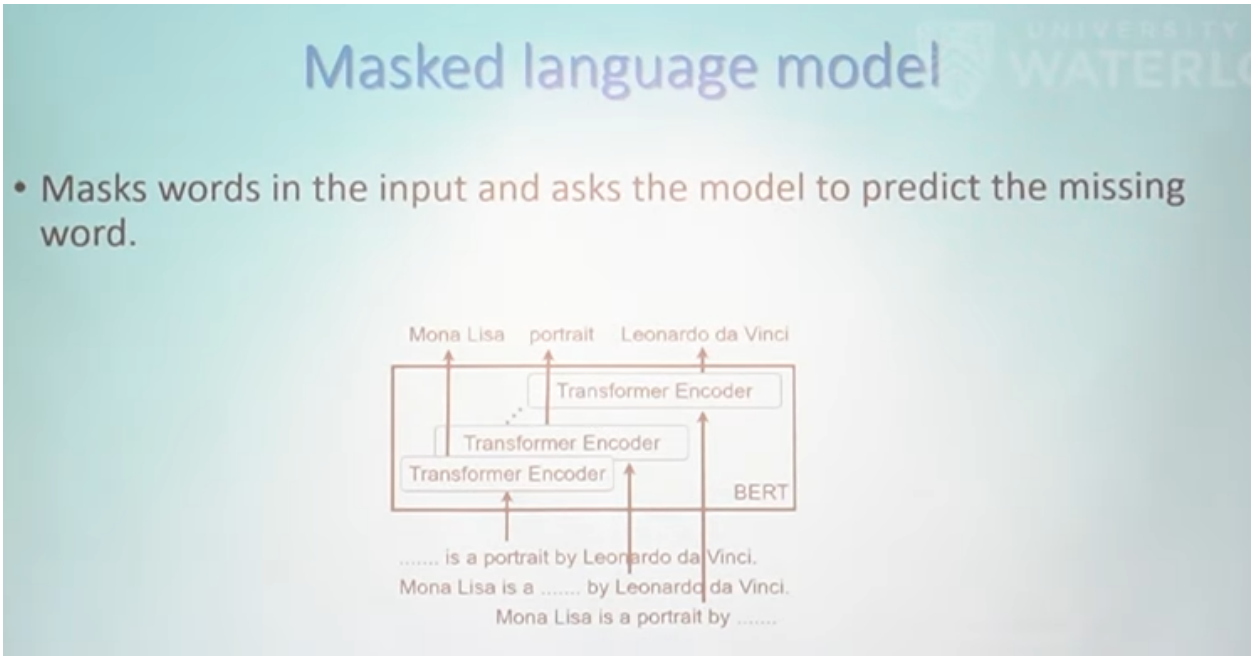




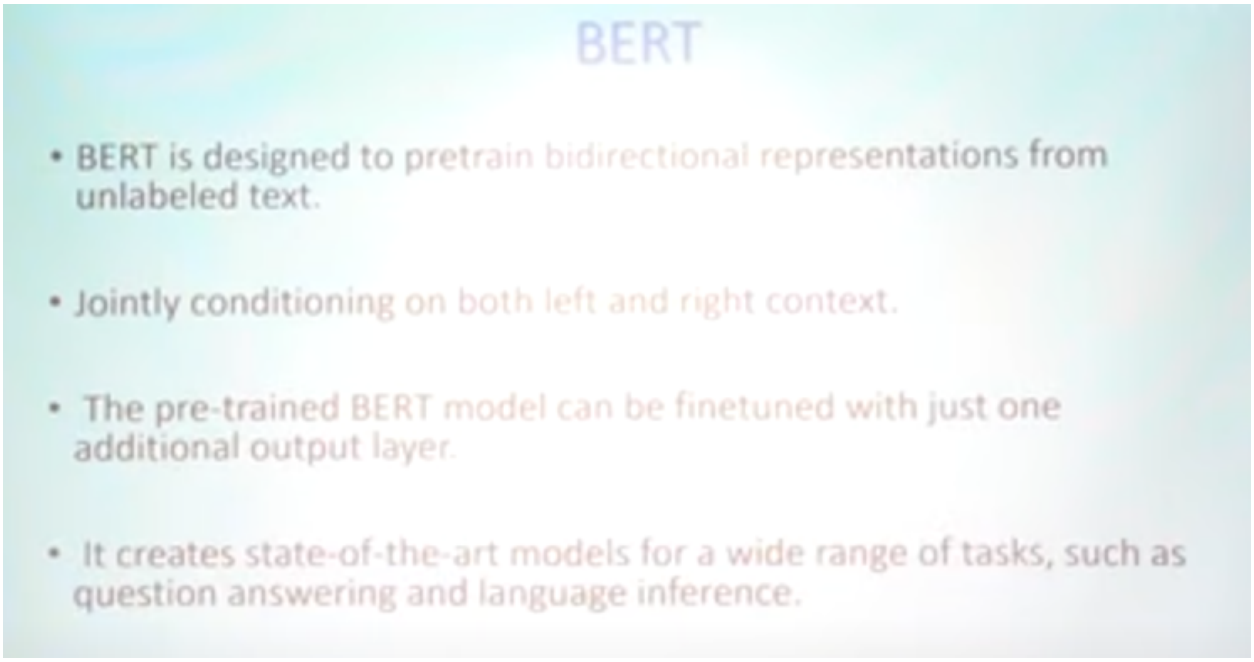
BERT: Bidirectional Encoder Representations from Transformers

Stack of encoders from the Transformer. We train it in an unsupervised manner.

We mask some words in our input sentence, pass it to the Encoder and ask it to predict the masked words. It checks the conditional probability of a token given all the words before it and ahead it. Hence, the name bi-directional.

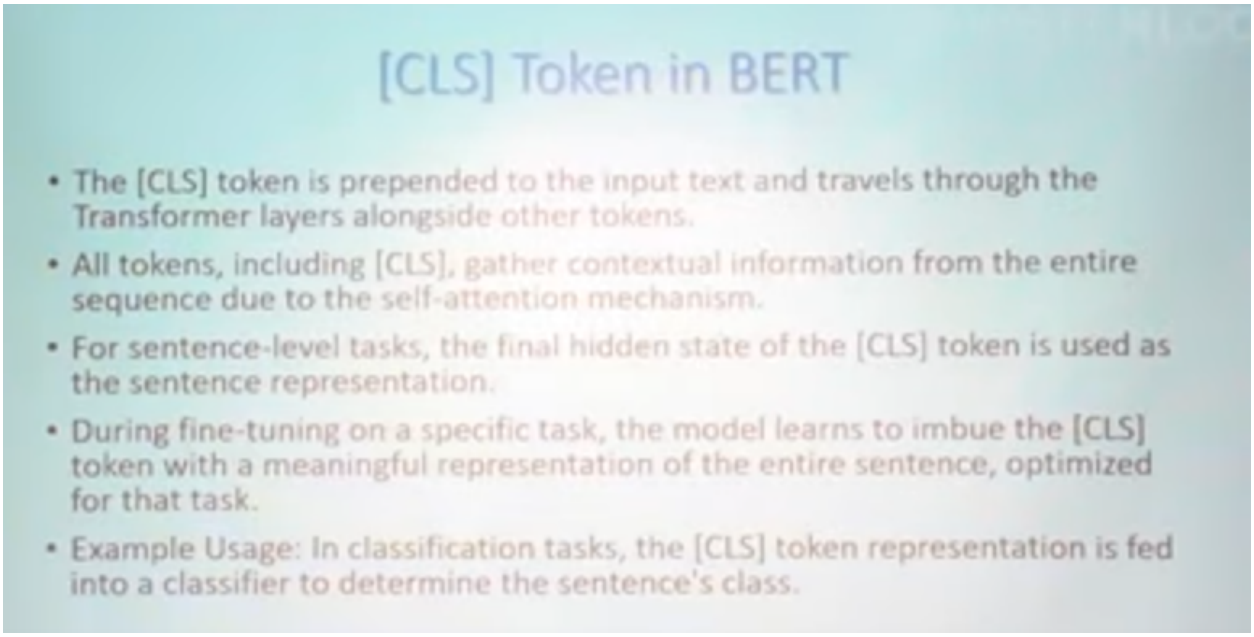


An additional task done while training, given two sentences A and B, it checks the likeliness of sentence B following sentence A. However, studies show that this second task doesn’t lead to significant performance improvement.



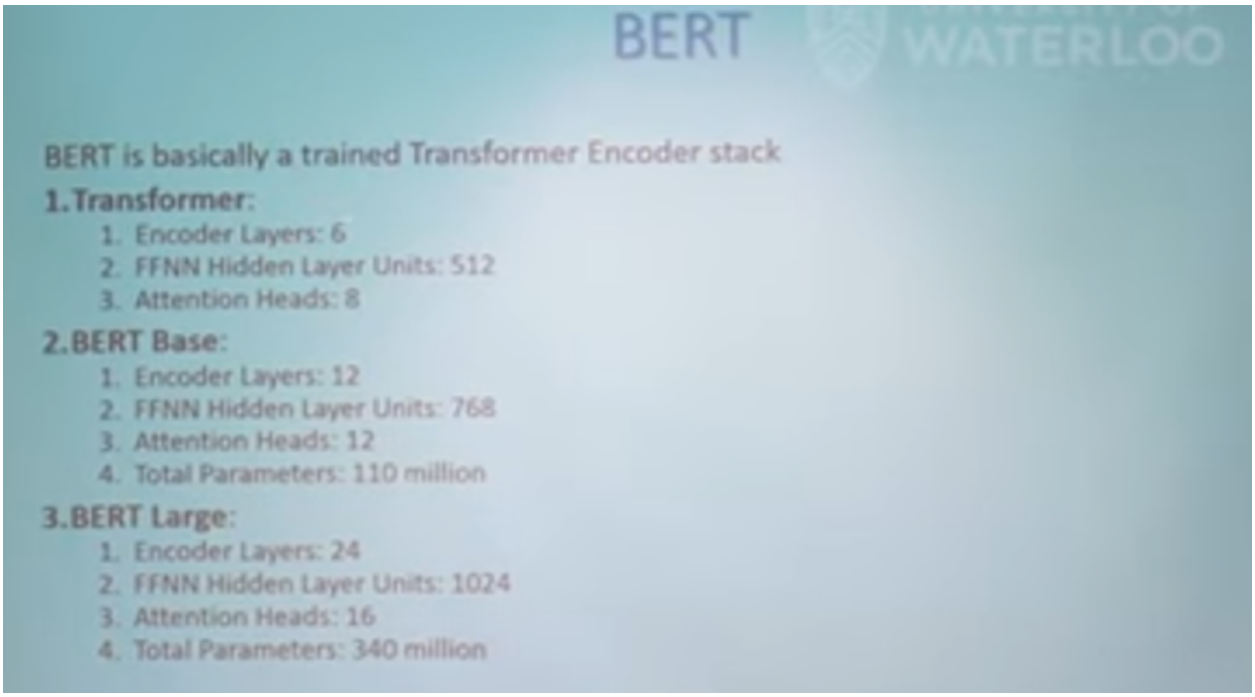
[CLS] Token in BERT

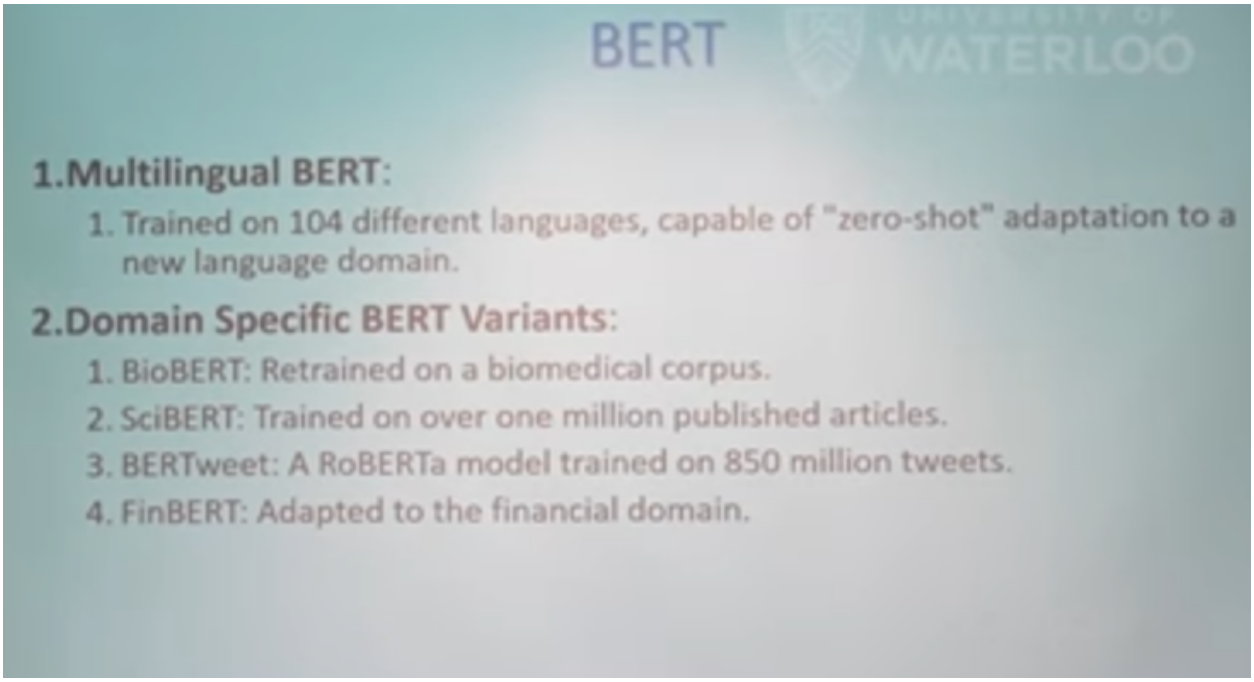
Apart from a representation for every word, the entire sentence can be represented using a single CLS token. This token can be trained by adding more layers and fine-tuning the model according to our application. In fact, during the fine-tuning, CLS token is the one that is impacted the most.



CLS is a free token, it is does not represent any particular word.

Variations of BERT





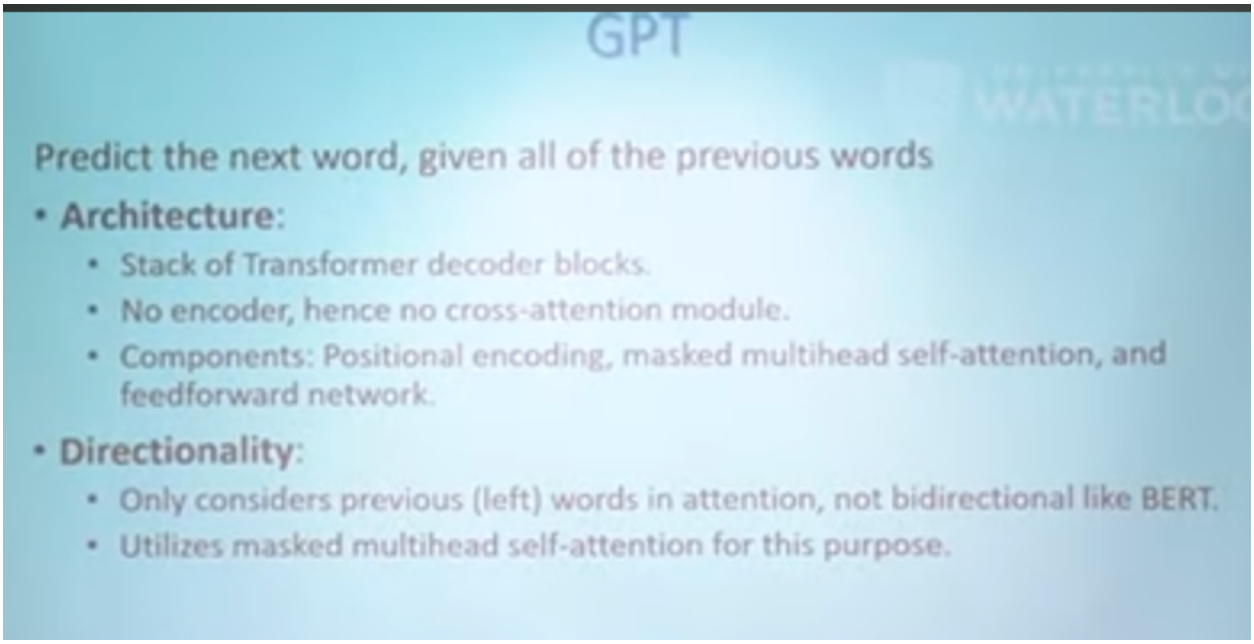
BioBERT: BERT trained on proteins

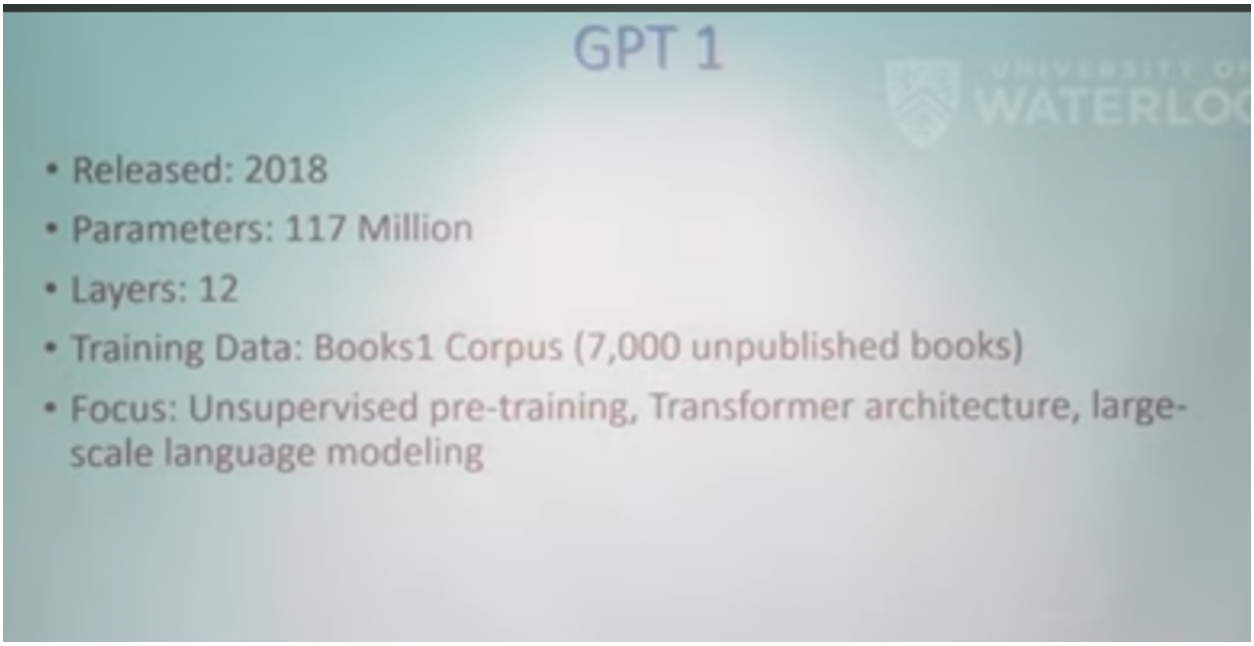
GPT: Generative Pre-Trained Transformer

Stack of Decoders

Since there is no Encoder to GPT, the Cross Attention layer has nothing to attend to(because it gets the key and value from output of encoder). So in GPT, the decoders only have the Masked Attention layer and Feed Forward NN.

In GPT, the process of training is to predict the next word in the sequence.



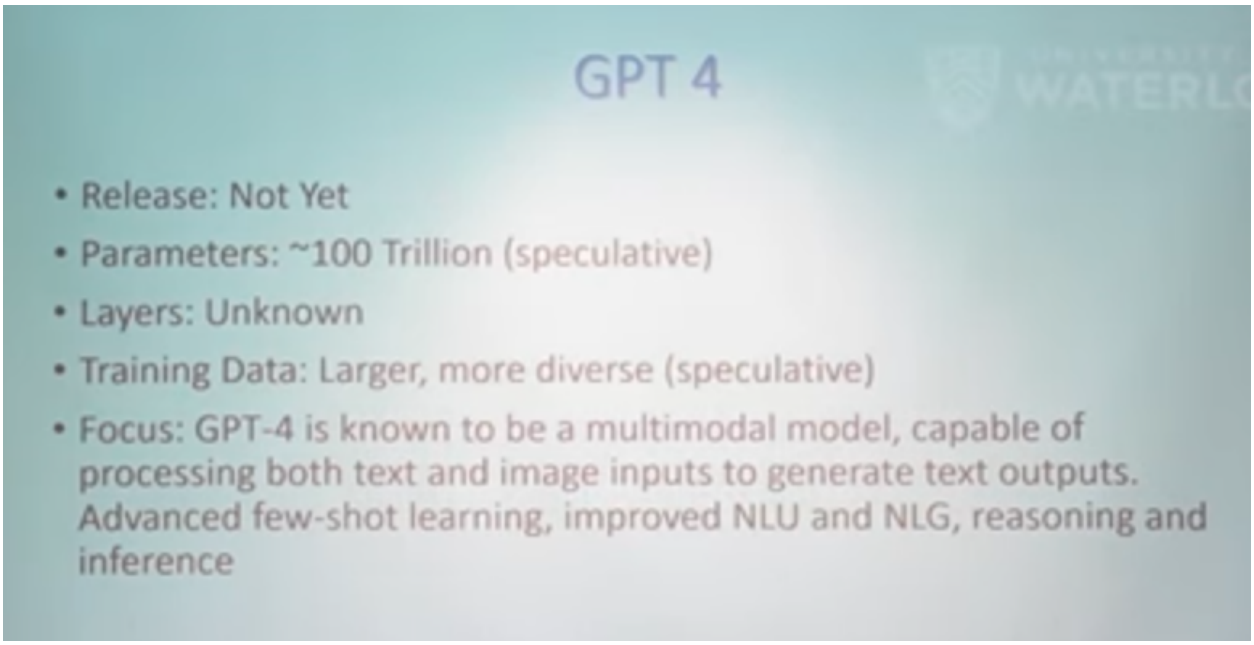


GPT 2

- Released: 2019
- Parameters: ~1.5 Billion
- Layers: 48
- Training Data: 40GB (English)
- Focus: Transformer architecture, self-attention mechanism

GPT 3

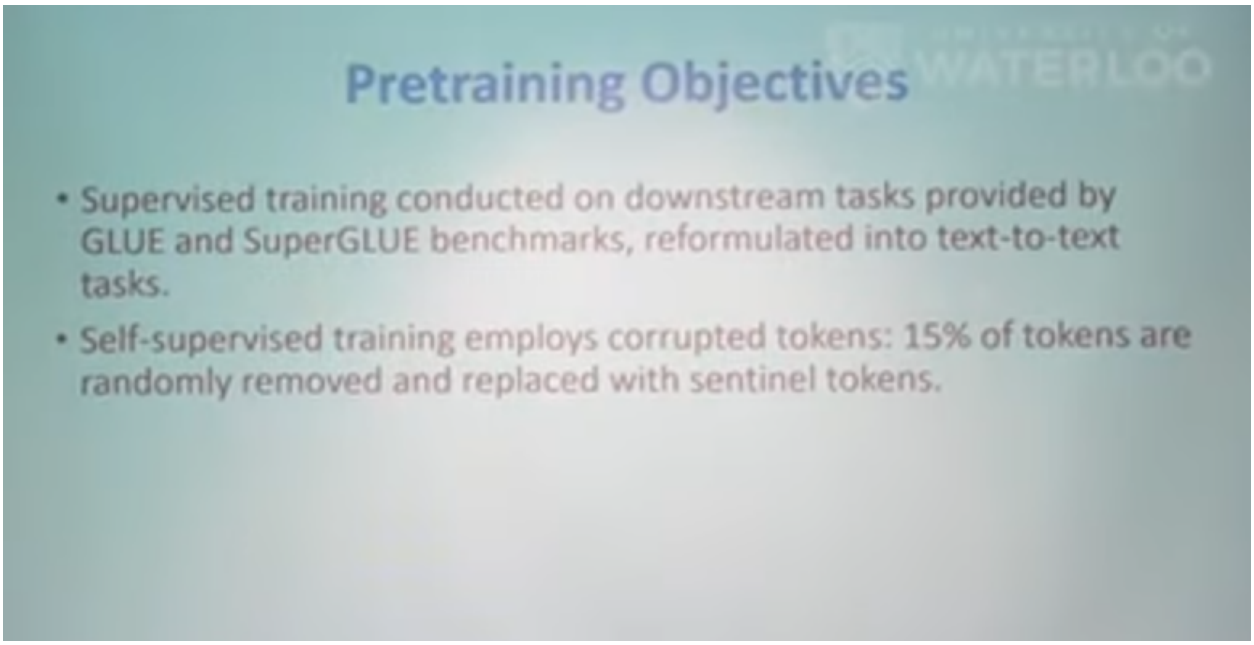
- Released: 2020
- Parameters: 175 Billion
- Layers: 175
- Training Data: 570GB (Multilingual)
- Focus: Few-shot learning, prompt engineering, Python support



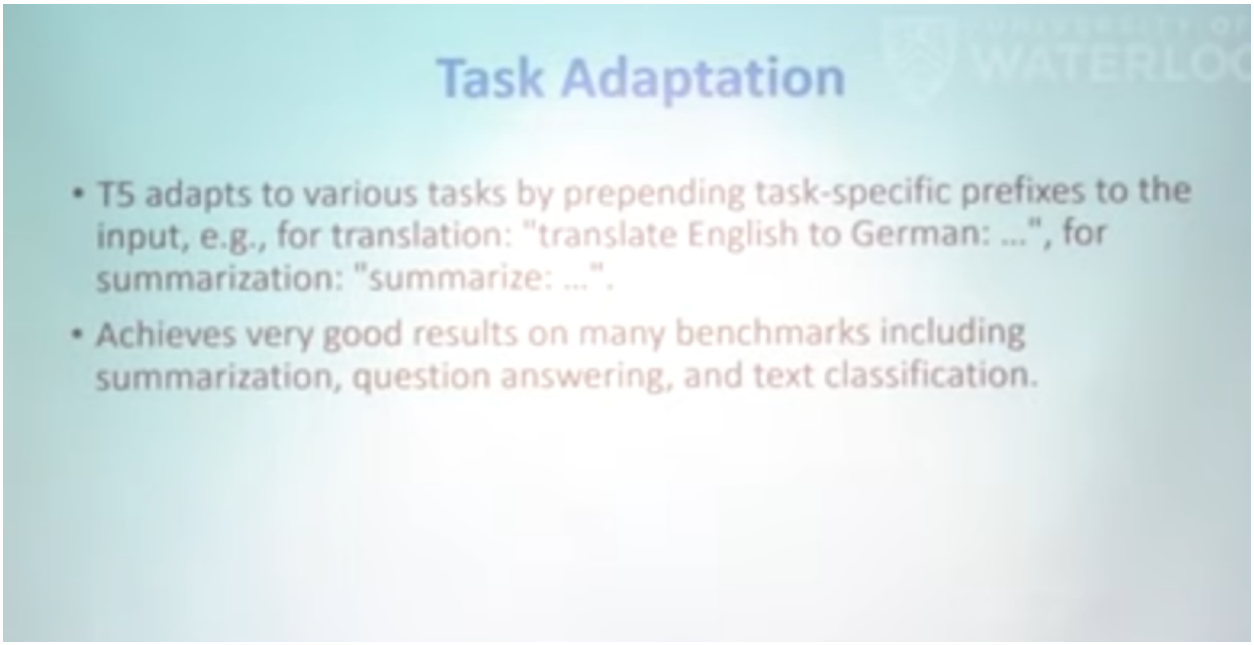
T5: Text-to-Text Transfer Transformer

T5 is roughly BERT and GPT put together. But this is only in terms of the architecture.

One interesting thing done in T5 is that, they have cast many natural language processing problems as text-to-text problems. For example, sentiment analysis involves labeling a sentence as positive or negative. So this can be cast as a text-to-text problem by training it to give out the token positive or negative.



Training T5 is a combination of both: supervised and unsupervised. Training a BERT or GPT is unsupervised while training for NLP tasks like sentiment analysis, translation, summarization is supervised.

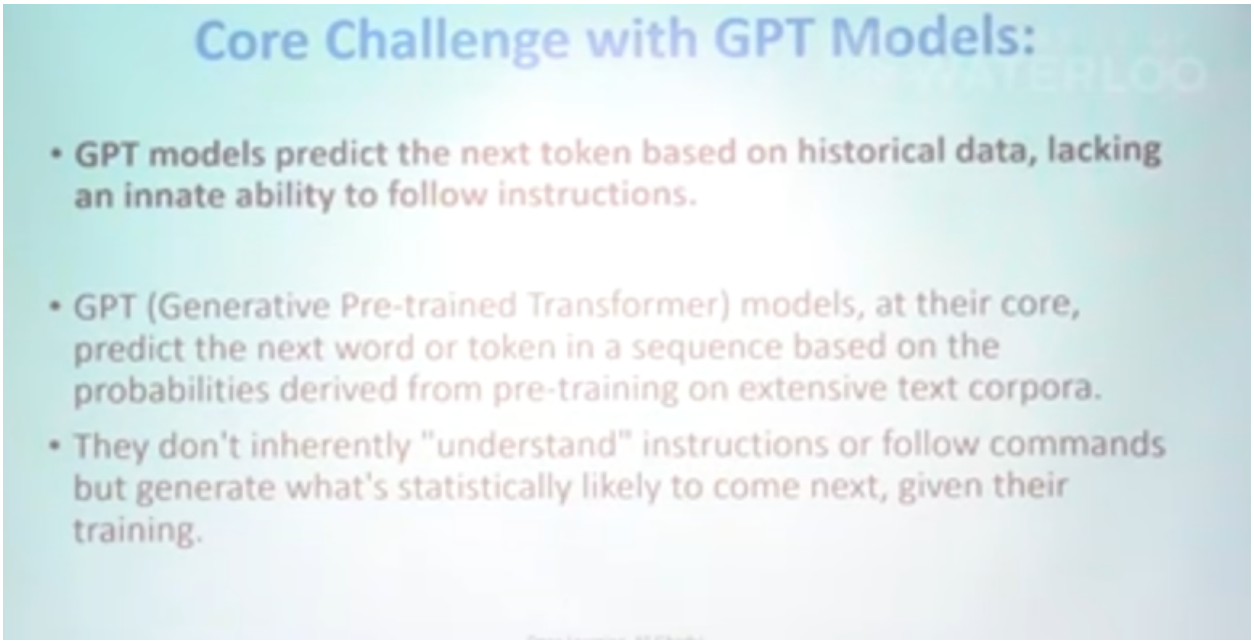


T5 also introduced different prompts. So if we write ‘summarize:’ and then the text to be summarized, it provides us

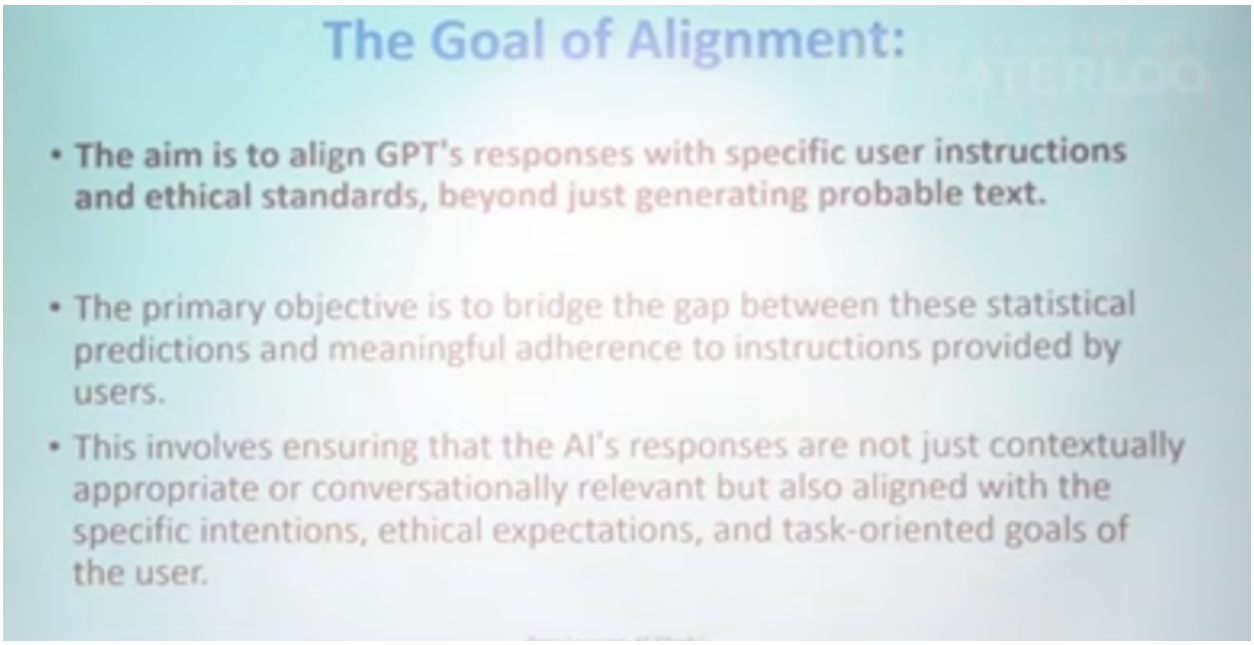
with a summary of that text.

Core Challenges with GPT models

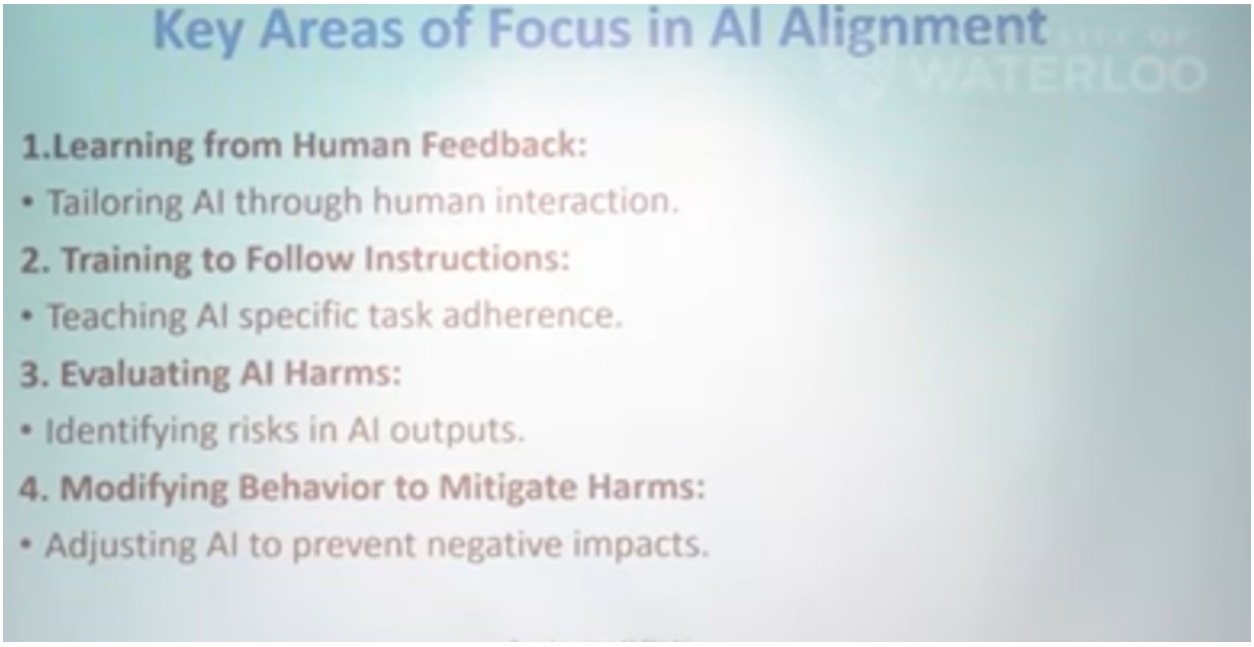
Inherently it is not good at conversations like a chatbot and is also not good at following instructions.



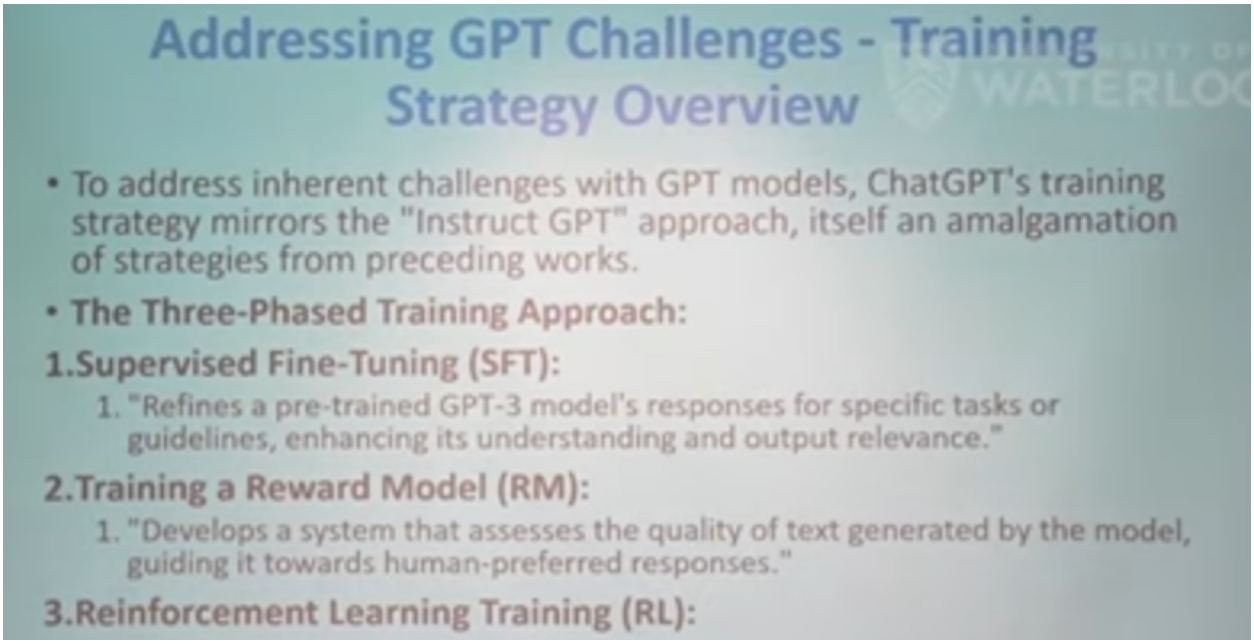
We want GPT to align to our specific instructions and also some moral rules. It should not be offensive in its response.



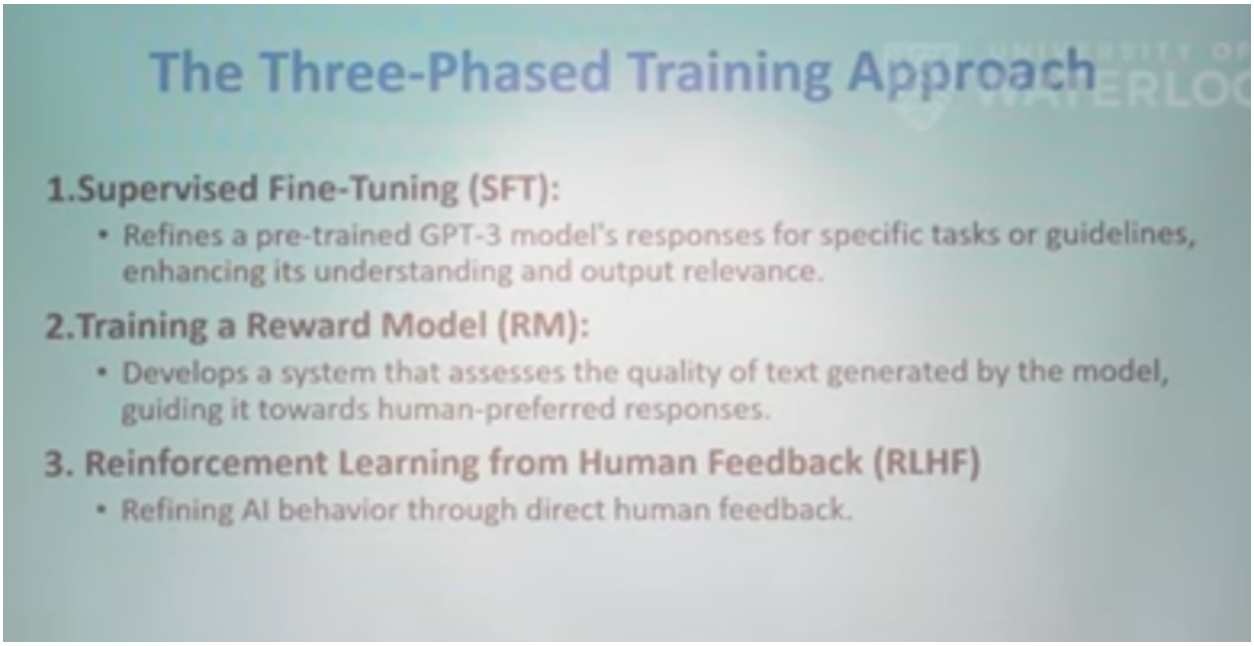
For example, we can try to jailbreak ChatGPT. If ask about a person, we get some response. If put the same prompt again, its response might change and it might end up stating something offensive. But it all comes from the data given to the model. The offensive opinion could have come from a comment by someone about the person. So we need to align the model to what we want as a user.



Three-Phased Training Approach for Alignment



SFT: Given the prompts, you teach the model to follow certain instructions.



RLHF: A dataset is generated from the human feedback received and it is then used to train the model in order to refine its behaviour.