

Question Answering

- ✓ **Video:** Week 3 Overview
6 min
- ✓ **Reading:** Week 3 Overview
10 min
- ✓ **Video:** Transfer Learning in NLP
7 min
- ✓ **Reading:** Transfer Learning in NLP
10 min
- ✓ **Video:** ELMo, GPT, BERT, T5
7 min
- ✓ **Reading:** ELMo, GPT, BERT, T5
10 min
- ▶ **Video:** Bidirectional Encoder Representations from Transformers (BERT)
4 min
- 📖 **Reading:** Bidirectional Encoder Representations from Transformers (BERT)
10 min
- ▶ **Video:** BERT Objective
2 min
- 📖 **Reading:** BERT Objective
10 min
- ▶ **Video:** Fine tuning BERT
2 min
- 📖 **Reading:** Fine tuning BERT
10 min
- ▶ **Video:** Transformer: T5
3 min
- 📖 **Reading:** Transformer T5
10 min
- ▶ **Video:** Multi-Task Training Strategy
5 min
- 📖 **Reading:** Multi-Task Training Strategy
10 min
- ▶ **Video:** GLUE Benchmark
2 min
- 📖 **Reading:** GLUE Benchmark
10 min
- ▶ **Video:** Question Answering
2 min
- 📖 **Reading:** Question Answering
10 min
- 📄 **Lab:** SentencePiece and BPE
2h
- 📖 **Reading:** Content Resource
10 min

Assignment



ELMo, GPT, BERT, T5

The models mentioned in the previous video were discovered in the following order.

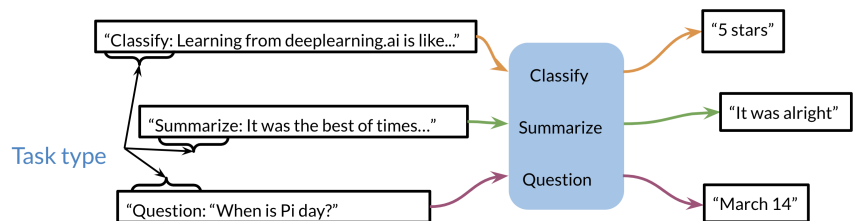


In CBOW, you want to encode a word as a vector. To do this we used the context before the word and the context after the word and we use that model to learn and creates features for the word. CBOW however uses a fixed window C (for the context).

What's ELMo does, it uses a bi-directional LSTM, which is another version of an RNN and you have the inputs from the left and the right.

Then Open AI introduced GPT. GPT unfortunately is uni-directional but it makes use of transformers. Although ELMo was bi-directional, it suffered from some issues such as capturing longer-term dependencies. Transformers help with that, but since GPT was still unidirectional, BERT was then introduced which stands for the Bi-directional Encoder Representation from Transformers. Last but not least, T5 was introduced which makes use of transfer learning and uses the same model to predict on many tasks. Here is an illustration of how it works.

T5: Text-to-Text



✓ Complete [Go to next item](#)

