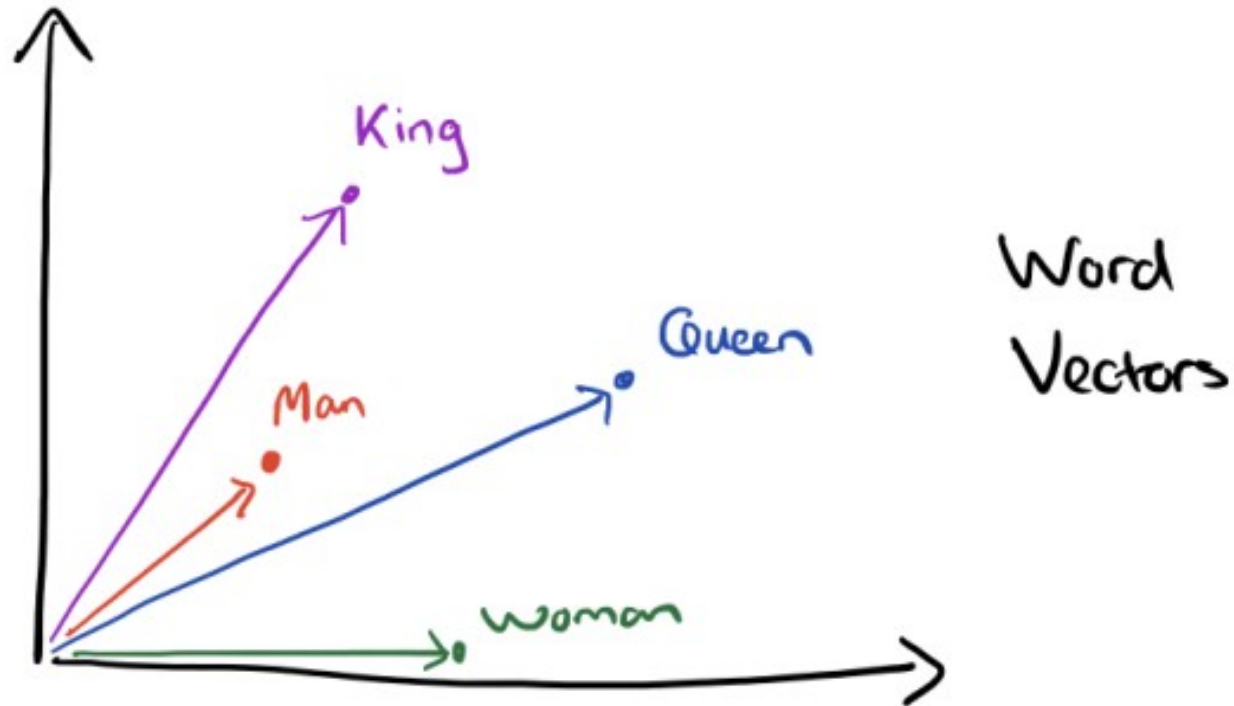


# Google BERT

State-of-the-art NLP  
Model

*Presented by Danny  
Luo*

# Learning Language Representations



# General Language Model

**Goal:** Build a general, pre-trained language representation model

**Why:** This model can be adapted to various NLP tasks easily, we don't have to retrain a model from scratch every time.

**How: ?**

*“For several years, people have been getting very good results “pre-training” [Deep Neural Networks] as a language model and then fine-tuning on some downstream NLP task (question answering, natural language inference, sentiment analysis, etc.).”*

– BERT author

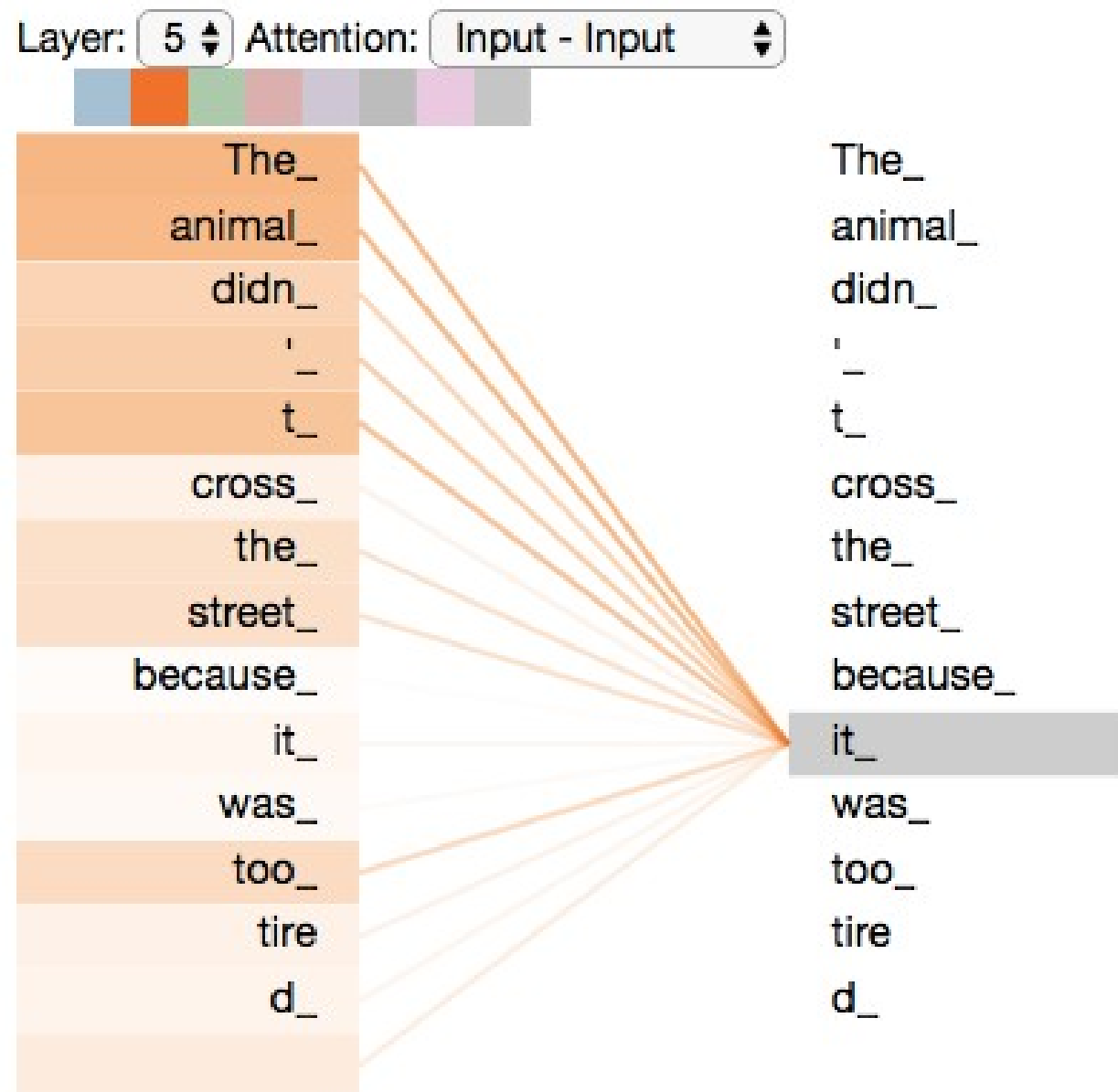
# Neural Language Models

- CNN, RNN
- Transformer
- Bidirectional Transformer (BERT)

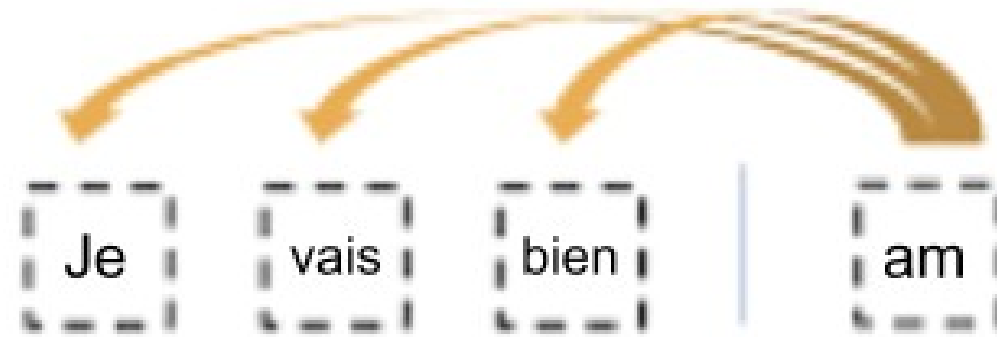
# Bidirectional Encoder Representation of Transformer (BERT)

- **Bidirectional:** BERT is naturally bidirectional
- **Generalizable:** Pre-trained BERT model can be fine-tuned easily for downstream NLP task
- **High-Performance:** Fine-tuned BERT models beat state-of-the-art results for many NLP tasks
- **Universal:** Trained on Wikipedia + BookCorpus. No special dataset needed

# Self-Attention



# Types of Attention



*Encoder-Decoder Attention*



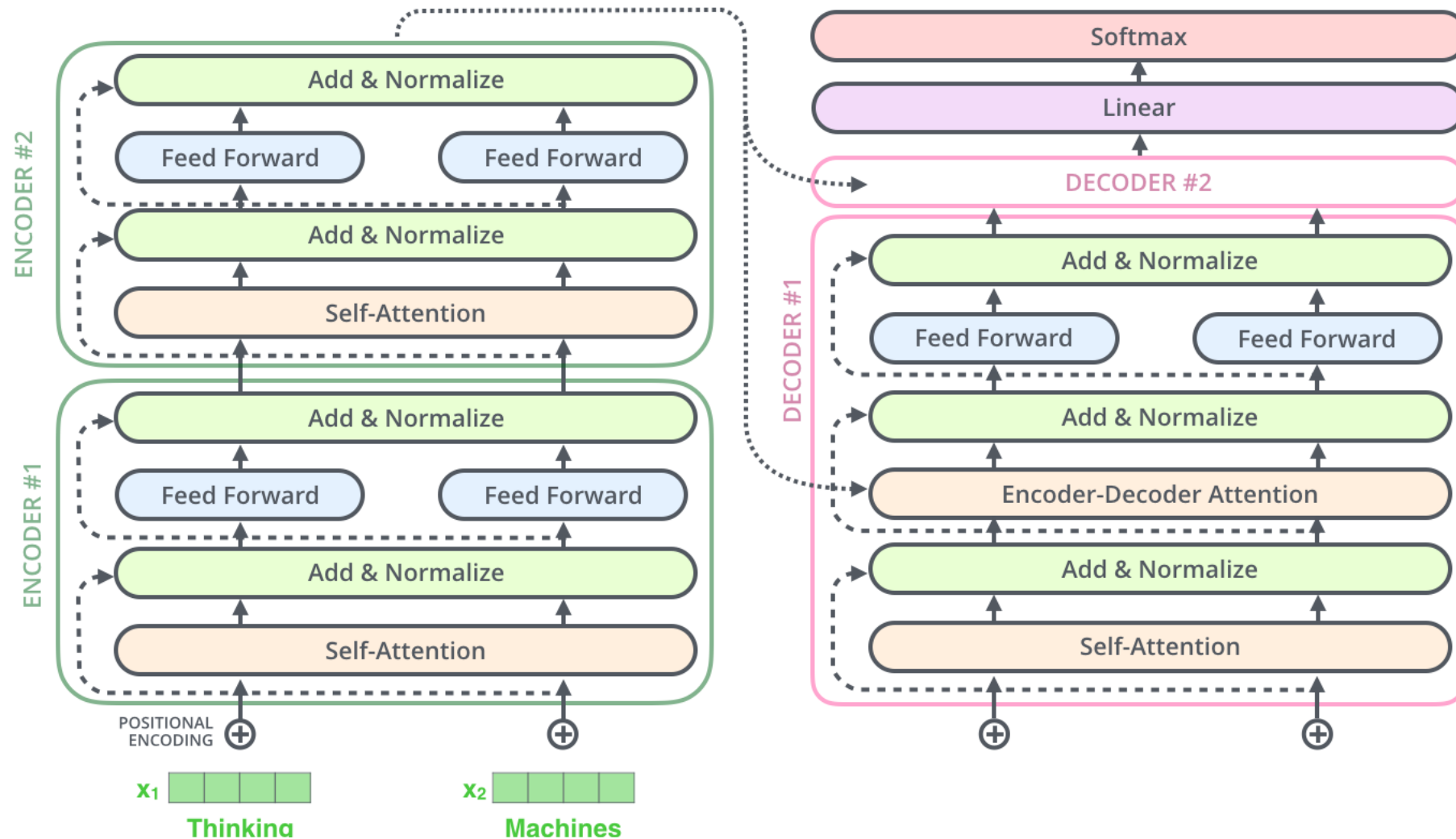
*Encoder Self-Attention*



*Decoder Self-Attention*



# Transformer





# Bidirectionality

- Language models are typically left-to-right or right-to-left
- However, most NLP tasks want **the best contextual representation of each word**, not just the left or right context

# Context is Everything

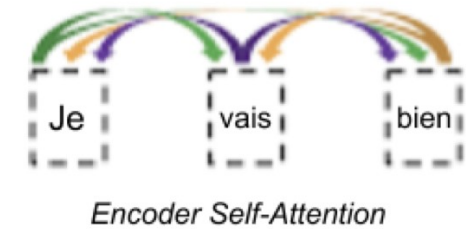
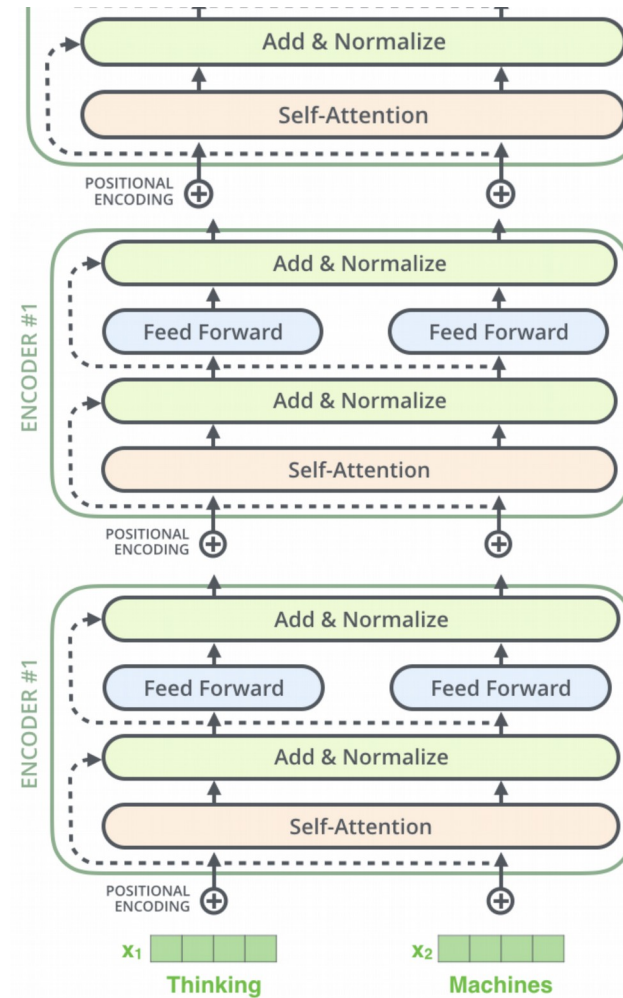
- No Context (Word2Vec)
  - river [bank]
  - [bank] deposit
- Left-to-Right Context (RNN)
  - I made a [bank] deposit
  - I made a [...]
- Bidirectional Context (?)
  - I made a [bank] deposit
  - I made a [...] deposit



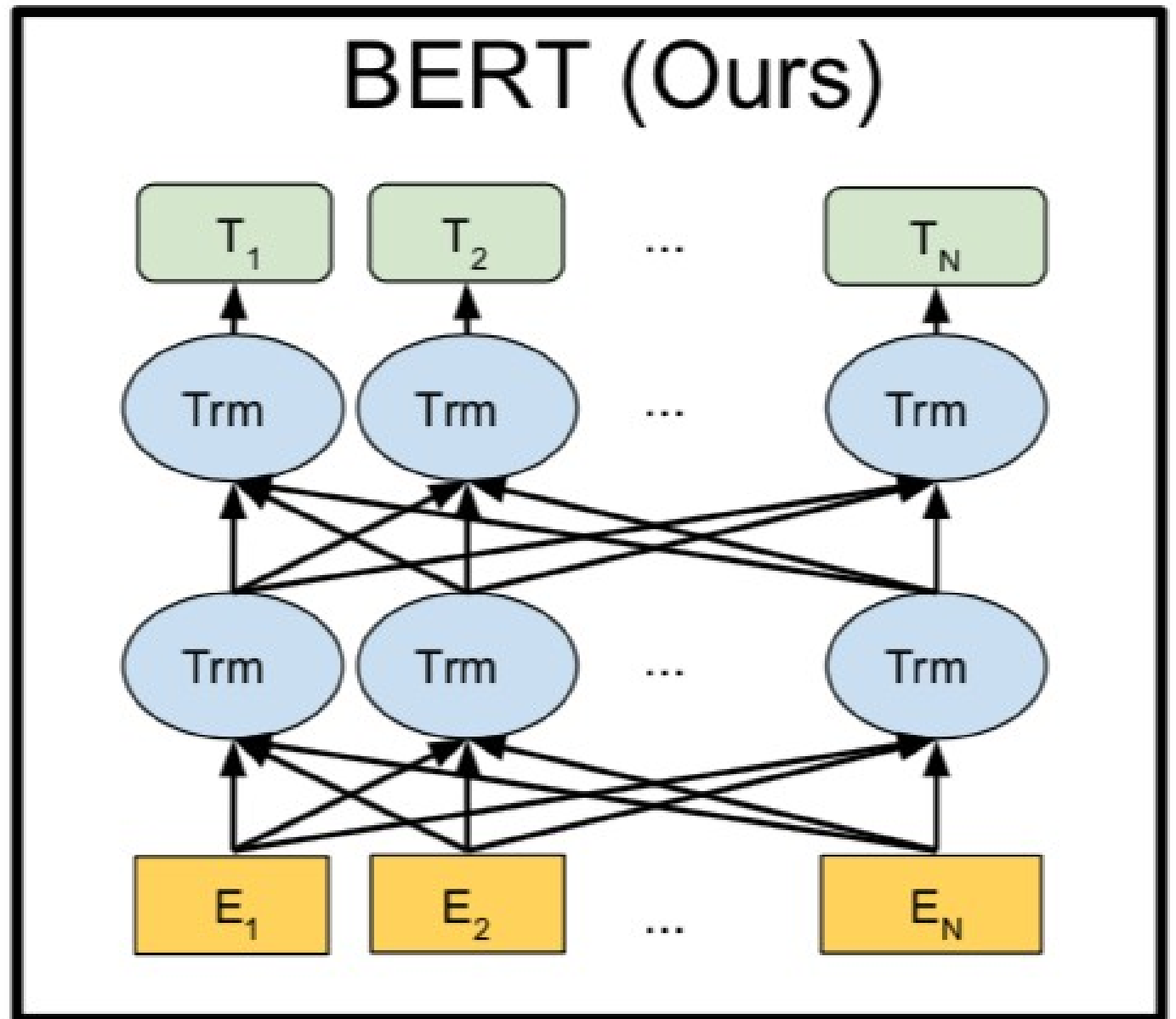
# Bidirectionality

- Language models are typically left-to-right or right-to-left
- However, most NLP tasks want **the best contextual representation of each word**, not just the left or right context
- People have tried combining a left-to-right model and a right-to-left model
  - Shallowly bidirectional (e.g. ELMo)
- **BERT:** A naturally bidirectional DNN model

# Bidirectional Transformer



# Bidirectional Encoder Representat ion of Transformer



# Model Details

## BERT Models

- **BERT-Base, Uncased** : 12-layer, 768-hidden, 12-heads, 110M parameters
- **BERT-Large, Uncased** : 24-layer, 1024-hidden, 16-heads, 340M parameters
- **BERT-Base, Cased** : 12-layer, 768-hidden, 12-heads , 110M parameters
- **BERT-Large, Cased** : 24-layer, 1024-hidden, 16-heads, 340M parameters (Not available yet. Needs to be re-generated).

# How to Train Bidirectional Model?

- **Problem:** Cannot train bidirectional model like normal Language Model
  - Words would be able to indirectly “see itself” through multiple layer connections
- **Solution:** Use two novel ‘unsupervised’ prediction tasks



Input: the man went to the [MASK1] . he bought a [MASK2] of milk.  
Labels: [MASK1] = store; [MASK2] = gallon

## Task 1: Masked Language Model

Sentence A: the man went to the store .  
Sentence B: he bought a gallon of milk .  
Label: IsNextSentence

Sentence A: the man went to the store .  
Sentence B: penguins are flightless .  
Label: NotNextSentence

## Task 2: Next Sentence Prediction

# Training

Trained on large 'unsupervised' corpus

- Wikipedia + BookCorpus

## Hardware

- BERT-Base trained on 4 Cloud TPUs for 4 days
- BERT-Large trained on 16 Cloud TPUs for 4 days

# Fine-tuning

- BERT can be fine-tuned inexpensively for many NLP tasks
  - GLUE
  - SQuAD
- Only requires one additional output layer, minimal number of parameters learned from scratch

The Black Death is thought to have originated in the arid plains of Central Asia, where it then travelled along the Silk Road, reaching Crimea by 1343. From there, it was most likely carried by Oriental rat fleas living on the black rats that were regular passengers on merchant ships. Spreading throughout the Mediterranean and Europe, the Black Death is estimated to have killed 30–60% of Europe's total population. In total, the plague reduced the world population from an estimated 450 million down to 350–375 million in the 14th century. The world population as a whole did not recover to pre-plague levels until the 17th century. The plague recurred occasionally in Europe until the 19th century.

**Where did the black death originate?**

*Ground Truth Answers:* the arid plains of Central Asia Central Asia Central Asia

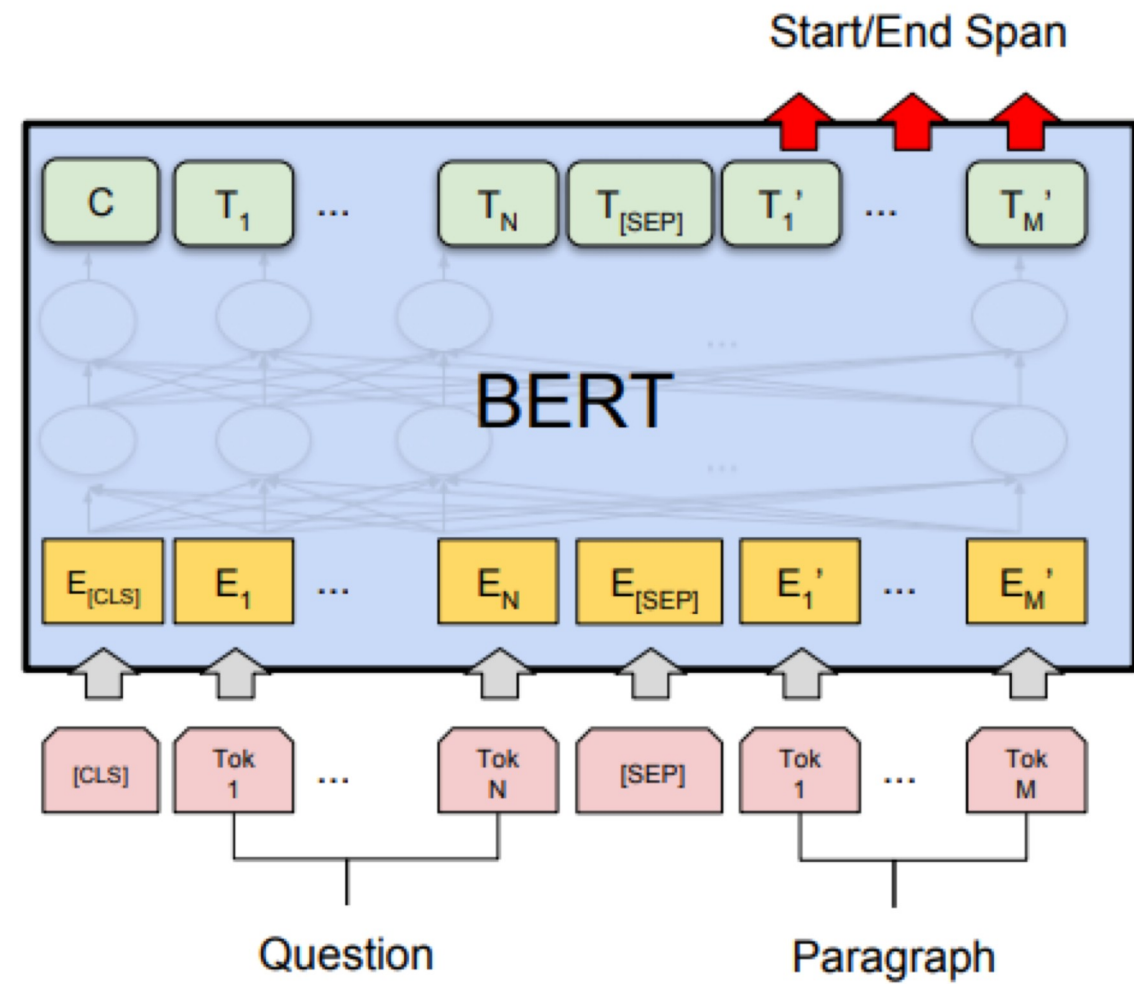
**How did the black death make it to the Mediterranean and Europe?**

*Ground Truth Answers:* merchant ships. merchant ships Silk Road

**How much of the European population did the black death kill?**

*Ground Truth Answers:* 30–60% of Europe's total population 30–60% of Europe's total population 30–60%

# SQuAD – Stanford Question Answering Dataset



(c) Question Answering Tasks:  
SQuAD v1.1

# SQuAD Performance


## SQuAD1.1 Leaderboard

Rank	Model	EM	F1
	Human Performance <i>Stanford University</i> (Rajpurkar et al. '16)	82.304	91.221
1 Oct 05, 2018	BERT (ensemble) <i>Google AI Language</i> <a href="https://arxiv.org/abs/1810.04805">https://arxiv.org/abs/1810.04805</a>	87.433	93.160
2 Sep 09, 2018	nlNet (ensemble) <i>Microsoft Research Asia</i>	85.356	91.202
3 Jul 11, 2018	QANet (ensemble) <i>Google Brain &amp; CMU</i>	84.454	90.490

# GLUE Performance

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.9	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	88.1	91.3	45.4	80.0	82.3	56.0	75.2
BERT <sub>BASE</sub>	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
BERT <sub>LARGE</sub>	<b>86.7/85.9</b>	<b>72.1</b>	<b>91.1</b>	<b>94.9</b>	<b>60.5</b>	<b>86.5</b>	<b>89.3</b>	<b>70.1</b>	<b>81.9</b>





*“But for us the really amazing and unexpected result is that when we go from a big model (12 Transformer blocks, 768-hidden, 110M parameters) to a really big model (24 Transformer blocks, 1024-hidden, 340M parameters), we get huge improvements even on very small datasets (small == less than 5,000 labeled examples).”*

- BERT author



# Conclusion

- BERT is a strong pre-trained language model that uses bidirectional transformers
  - Trained on two novel language modelling tasks
- BERT may be fine-tuned to beat many SOTA results on various NLP tasks

What I didn't explain:

- The mechanics of BERT pre-training (Consult source code on github)
- How to fine-tune BERT to NLP tasks
- Ablation Studies: Critical evaluation of BERT methodology



**End**

# Discussion Points

- Could you combine this with a stacked, pre-trained transformer decoder for machine translation? (Does this make sense?)
- Will we need to train models from scratch in the future?

# Discussion Points by Gordon

- In ablation studies between BERT and OpenAI GPT, **the authors never examine the effect of adding the Wikipedia corpus to the training data.** In their "LTR & No NSP" ablation study, they acknowledge that the training data is different, but don't seem to consider this significant. **How well do you think OpenAI GPT would do with this additional training data, and how well would BERT do without it?**
- Some of their **justifications seem to lie on very slight numerical differences.** For example they state that 1M pretraining steps was necessary to achieve higher accuracy and that BERT\_base achieves almost 1% additional accuracy on MNLI when trained on 1M steps compared to 500k steps. First of all, from the graph the difference looks less than "almost 1%", and the difference is even less significant between 600k and 1M steps. Similarly when they are comparing using the model in a feature-based approach they find that the best feature aggregation method achieves only 0.3 less F1 than the model that fine-

# Discussion Points by Florian

- In the paper they mask randomly 15% of the tokens and from those 10% are replaced with random word or a synonym. What do you think is the importance of that step? Does leaving it out results in a small performance decrease or in complete failure?
- They showed the model can be used to various NLP problems. Can you think it would also work on other sequence based problems that are not necessarily NLP. For example on customer transactional data.

# Archive – What is a Language Model?

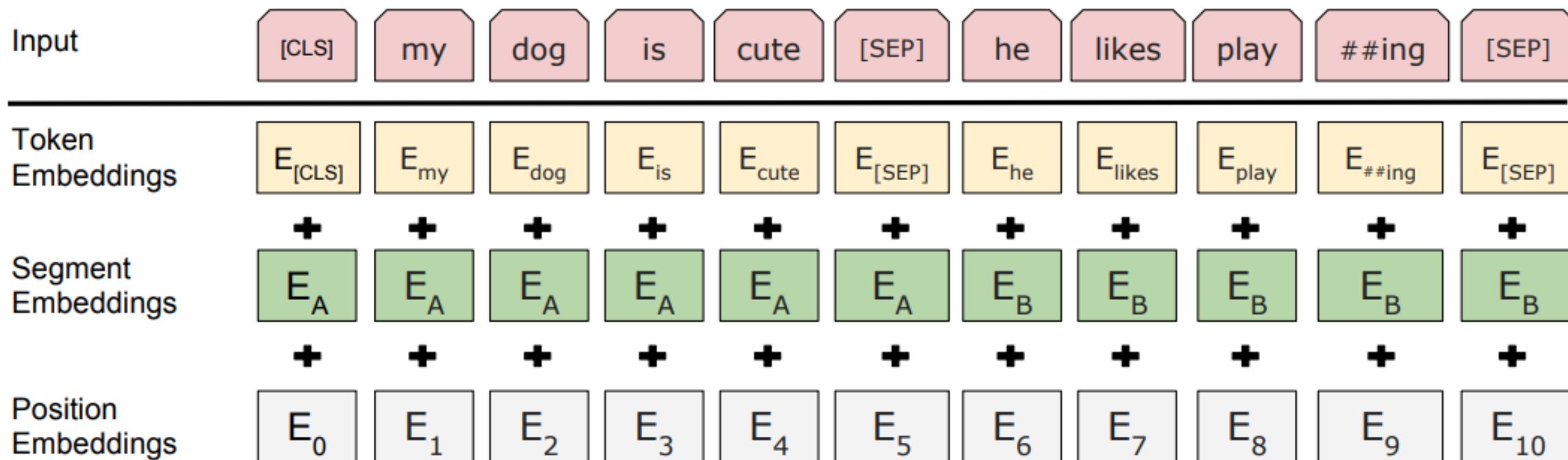
- Learns the probability distribution of a sequence of words
- Given a sequence of length  $m$ , the language model outputs the likelihood of the sequence  $P(w_1, \dots, w_m)$
- Given a sequence of length  $m$ , the language model outputs the likelihood of the sequence
- Given a sequence of words  $w_1, \dots, w_m$ , the model outputs the likelihood of the next word:  $P(w_{m+1} | w_1, \dots, w_m)$
- Given a sequence of words, the model outputs the likelihood of the next word:

# Archive – Simple Language Model Example

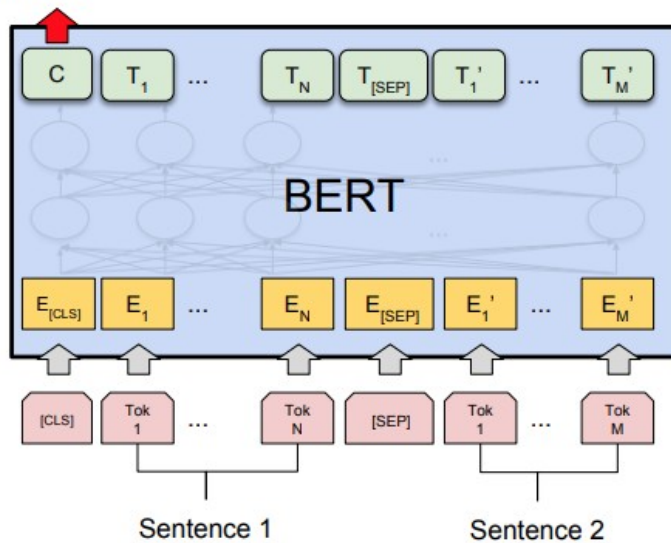
- A language model wants to learn the likelihood of the sentence:  
'The man went to the store' = [ $\langle s \rangle$ , The, man, went, to, the, store,  $\langle /s \rangle$ ]  
'The man went to the store' = [ $\langle s \rangle$ , The, man, went, to, the, store,  $\langle /s \rangle$ ]
  - Typical left-to-right model would learn:
- Typical left-to-right model would learn:  
$$P(\text{The man went to the store}) = P(\text{The} \mid \langle s \rangle) * P(\text{man} \mid \langle s \rangle, \text{The}) * P(\text{went} \mid \langle s \rangle, \text{The}, \text{man}) * \dots$$



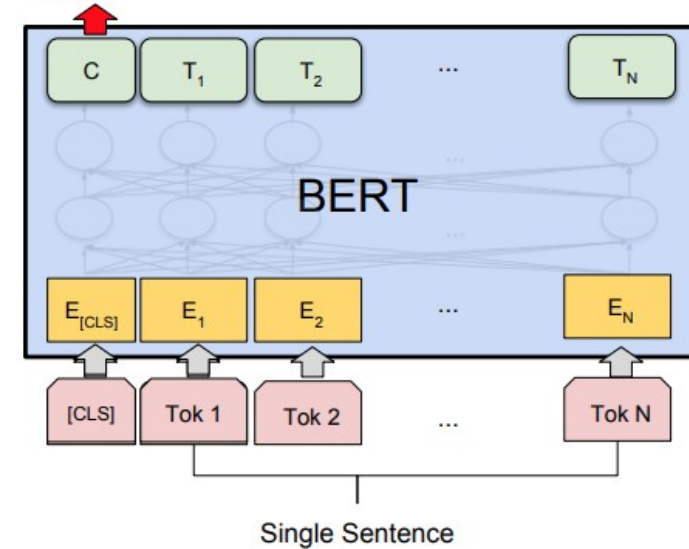
# Archive – Input Embedding



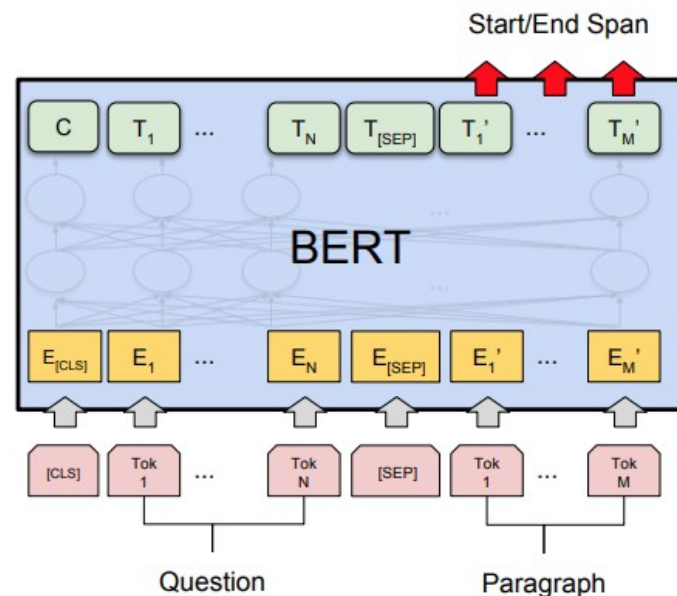
# Archive – Task-Specific Models Using BERT



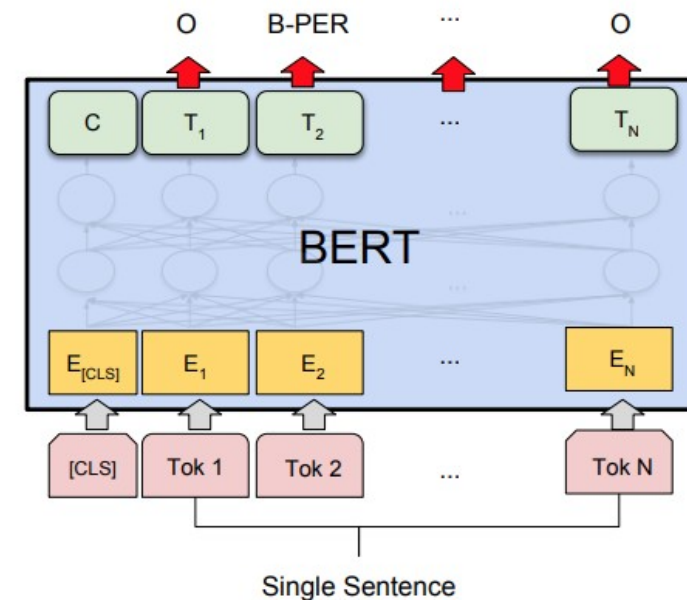
(a) Sentence Pair Classification Tasks:  
MNLI, QQP, QNLI, STS-B, MRPC,  
RTE, SWAG



(b) Single Sentence Classification Tasks:  
SST-2, CoLA



(c) Question Answering Tasks:  
SQuAD v1.1



(d) Single Sentence Tagging Tasks:  
CoNLL-2003 NER

# Archive

# Sources

## Transformer

*Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In the Annual Conference on Neural Information Processing Systems (NIPS).*

- \* <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>
- \* <https://medium.com/syncedreview/best-nlp-model-ever-google-bert-sets-new-standards-in-11-languagetasks-4a2a189bc155>
- \* <http://jalammar.github.io/illustrated-transformer/> • BERT<sub>BASE</sub>: L=12, H=768, A=12, Total Parameters=110M
- \* <https://mchromiak.github.io/articles/2017/Sep/12/Transformer-Attention-is-all-you-need/>
- \* <http://nlp.seas.harvard.edu/2018/04/03/attention.html>

## BERT

*Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv e-prints.*

- <https://github.com/google-research/bert/tree/d45d0fb36d932b603dcdffb149daabd63db10d24>
- <https://github.com/codertimo/BERT-pytorch>
- [https://www.reddit.com/r/MachineLearning/comments/9nfqxz/r\\_bert\\_pretraining\\_of\\_deep\\_bidirectional/](https://www.reddit.com/r/MachineLearning/comments/9nfqxz/r_bert_pretraining_of_deep_bidirectional/)