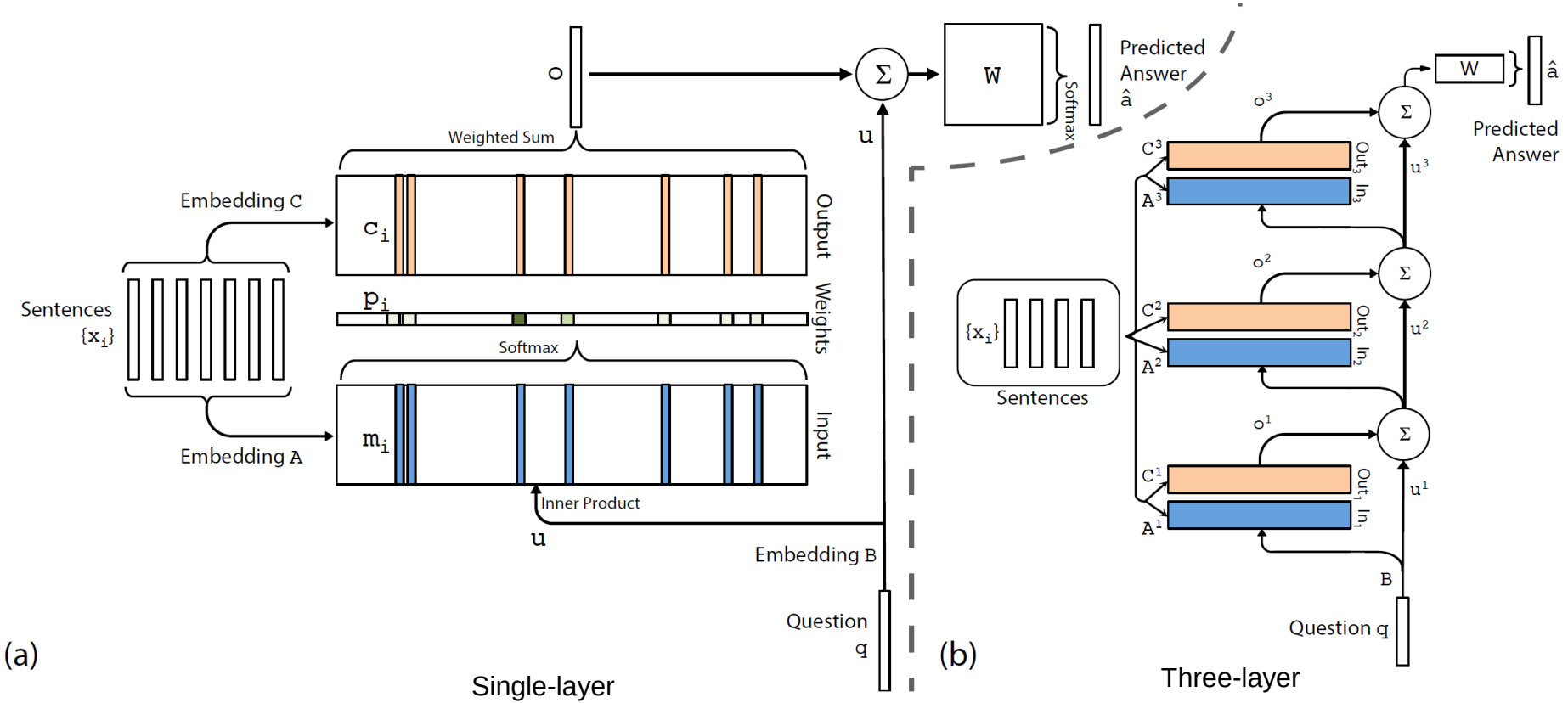
CS182/282A: Designing, Visualizing and Understanding Deep Neural Networks

John Canny

Spring 2019

Lecture 15: Neural Dialog Systems

Last Time: Q&A Memory Network



Last Time: Multi-Hop Inference

Story (1: 1 supporting fact)	Support	Hop 1	Hop 2	Hop 3
Daniel went to the bathroom.		0.00	0.00	0.03
Mary travelled to the hallway.		0.00	0.00	0.00
John went to the bedroom.		0.37	0.02	0.00
John travelled to the bathroom.	yes	0.60	0.98	0.96
Mary went to the office.		0.01	0.00	0.00
Where is John? Answer: bathroom Prediction: bathroom				

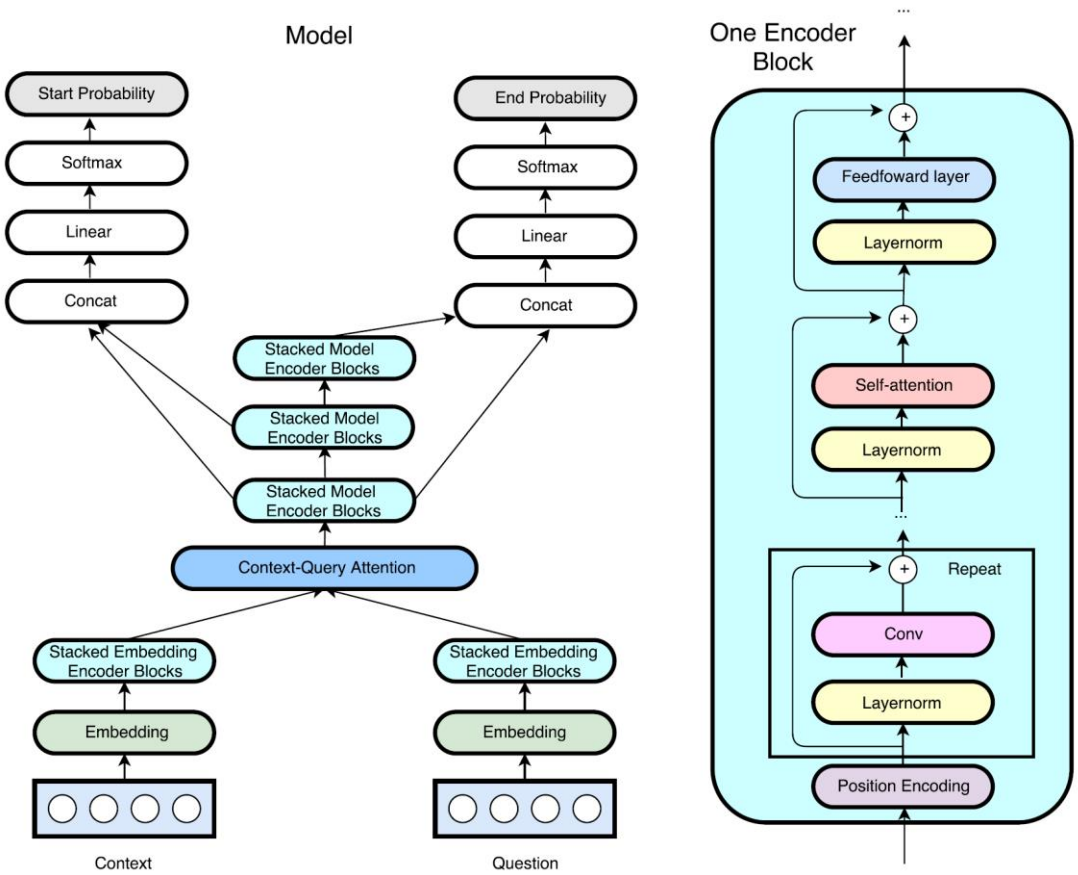
Story (2: 2 supporting facts)	Support	Hop 1	Hop 2	Hop 3
John dropped the milk.		0.06	0.00	0.00
John took the milk there.	yes	0.88	1.00	0.00
Sandra went back to the bathroom.		0.00	0.00	0.00
John moved to the hallway.	yes	0.00	0.00	1.00
Mary went back to the bedroom.		0.00	0.00	0.00
Where is the milk? Answer: hallway Prediction: hallway				

Story (16: basic induction)	Support	Hop 1	Hop 2	Hop 3
Brian is a frog.	yes	0.00	0.98	0.00
Lily is gray.		0.07	0.00	0.00
Brian is yellow.	yes	0.07	0.00	1.00
Julius is green.		0.06	0.00	0.00
Greg is a frog.	yes	0.76	0.02	0.00
What color is Greg? Answer: yellow Prediction: yellow				

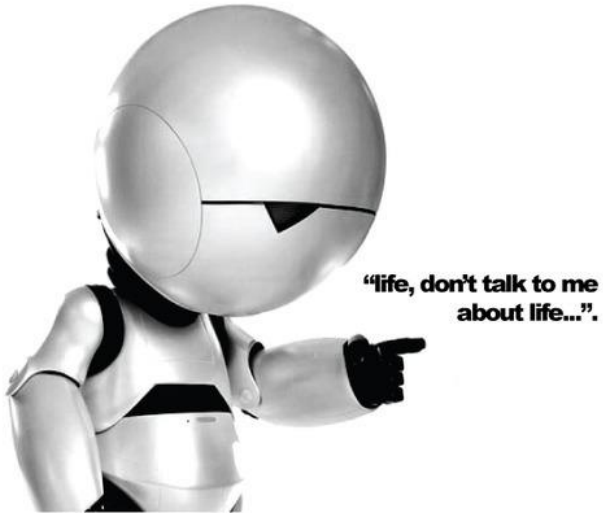
Story (18: size reasoning)	Support	Hop 1	Hop 2	Hop 3
The suitcase is bigger than the chest.	yes	0.00	0.88	0.00
The box is bigger than the chocolate.		0.04	0.05	0.10
The chest is bigger than the chocolate.	yes	0.17	0.07	0.90
The chest fits inside the container.		0.00	0.00	0.00
The chest fits inside the box.		0.00	0.00	0.00
Does the suitcase fit in the chocolate? Answer: no Prediction: no				

Note: Answers are single-word, predicted by the output softmax

Last Time: QANet – Convolution + Transformer



This Time: Dialog



Goal-Directed Dialog

Goal-Directed Dialog Systems, in contrast to chatbots, aim to not only engage the user, but help the user with goal-directed tasks.

Traditional dialog systems use slot-filling:
“We’d like a table for two at 8pm, outside if possible”

Fills slots for
- Number of people
- Time
- Location preference

The interaction may need several turns to:
- Clarify users intention (slot doesn’t match)
- Ask for a different option (request can’t be met)
- Fill in missing slots

Machine Learning Goal-Directed Dialog

The idea is to learn from sample dialog how to respond to user queries.

Task Summary

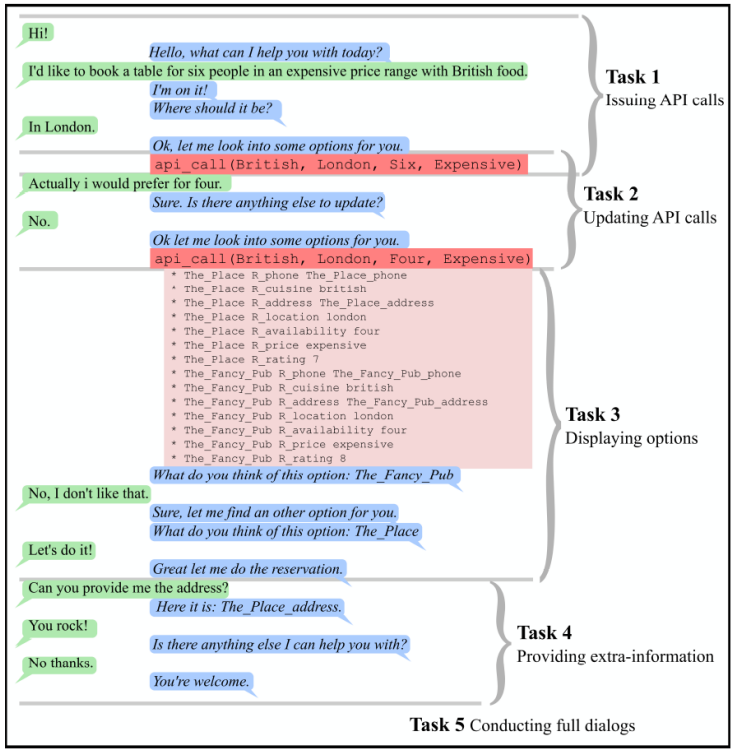
Task 1: Issuing API calls. A user query contains from 0 to 4 of the required fields. Agent must ask questions for filling the missing.

Task 2: Updating API calls. After an API call fromTask 1, user asks to update their requests. The agent ask user if they are done and issue the updated API call.

Task 3: Displaying options. Given a user request, we query the KB to get possible responses.

Task 4: Providing extra information. Users then ask for the phone number of the restaurant, its address or both.

Task 5: Conducting full dialogs We combine Tasks 1-4 to generate full dialogs just as in Figure 1.



Bordes et al “LEARNING END-TO-END GOAL-ORIENTED DIALOG” ICML 2017

Datasets

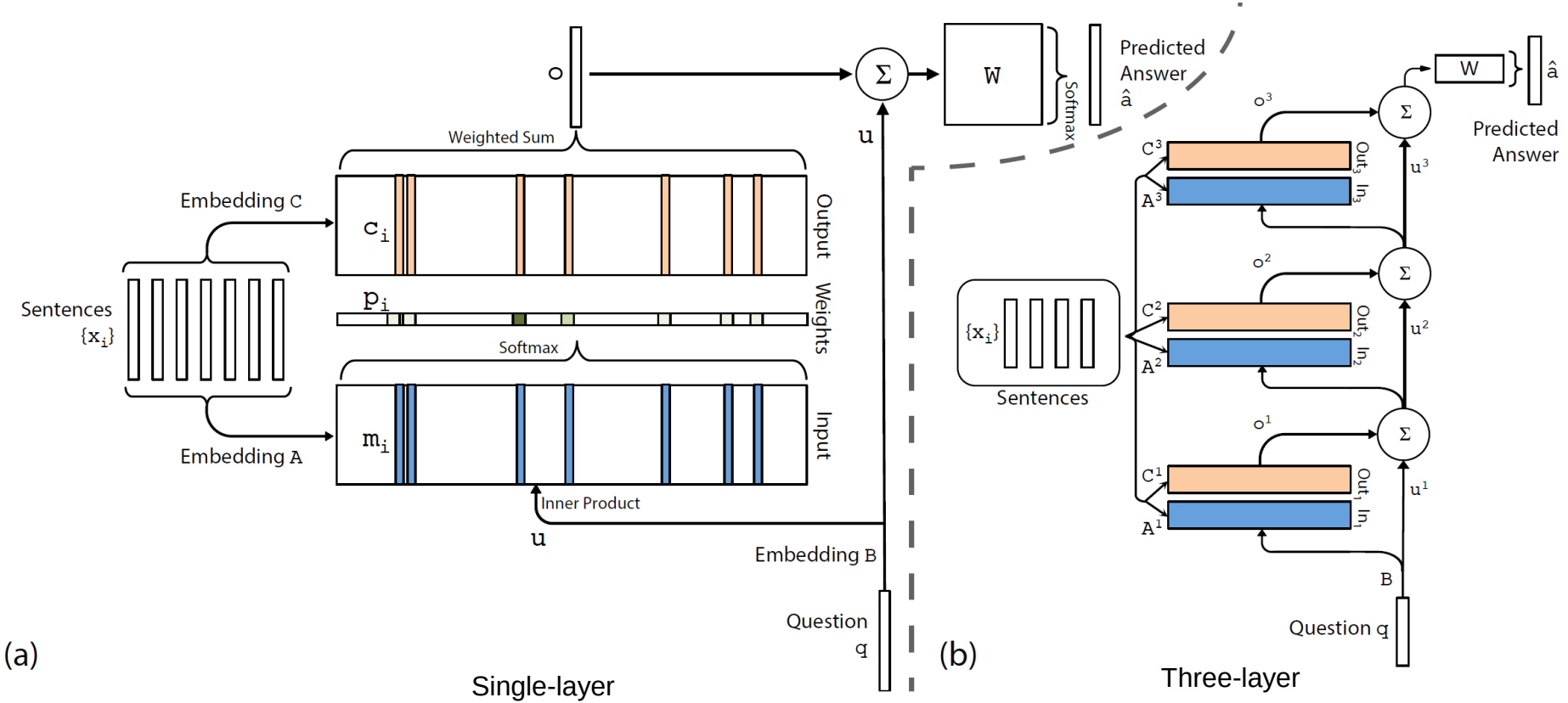
Restaurant Reservations: Contains two KBs of 4,200 facts and 600 restaurants each (5 types of cuisine 5 locations 3 price ranges 8 ratings). Use one of the KBs to generate the standard training, validation and test dialogs, and use the other KB only to generate test dialogs, termed Out-Of-Vocabulary (OOV) test sets.

Dialog State Tracking Challenge: Another restaurant booking dataset, but using data from real users. We use data from DSTC2 (Henderson et al., 2014a), which was designed for dialog state tracking hence every dialog turn is labeled with a state (a user intent + slots) to be predicted.

Online Concierge Service: Data extracted from a real online concierge service: users make requests through a text-based chat interface that are handled by human operators who can make API calls.

Bordes et al “LEARNING END-TO-END GOAL-ORIENTED DIALOG” ICML 2017

Q&A Memory Network



Experiments

Task	Rule-based Systems	TF-IDF Match no type	TF-IDF Match + type	Nearest Neighbor	Supervised Embeddings	Memory Networks no match type	Memory Networks + match type
T1: Issuing API calls	*100 (100)*	5.6 (0)	22.4 (0)	55.1 (0)	**100** (100)	**99.9** (99.6)	**100** (100)
T2: Updating API calls	*100 (100)*	3.4 (0)	16.4 (0)	68.3 (0)	68.4 (0)	**100** (100)	98.3 (83.9)
T3: Displaying options	*100 (100)*	8.0 (0)	8.0 (0)	58.8 (0)	64.9 (0)	**74.9** (2.0)	**74.9** (0)
T4: Providing information	*100 (100)*	9.5 (0)	17.8 (0)	28.6 (0)	57.2 (0)	59.5 (3.0)	**100** (100)
T5: Full dialogs	*100 (100)*	4.6 (0)	8.1 (0)	57.1 (0)	75.4 (0)	**96.1** (49.4)	93.4 (19.7)
T1(OOV): Issuing API calls	*100 (100)*	5.8 (0)	22.4 (0)	44.1 (0)	60.0 (0)	72.3 (0)	**96.5** (82.7)
T2(OOV): Updating API calls	*100 (100)*	3.5 (0)	16.8 (0)	68.3 (0)	68.3 (0)	78.9 (0)	**94.5** (48.4)
T3(OOV): Displaying options	*100 (100)*	8.3 (0)	8.3 (0)	58.8 (0)	65.0 (0)	74.4 (0)	**75.2** (0)
T4(OOV): Providing inform.	*100 (100)*	9.8 (0)	17.2 (0)	28.6 (0)	57.0 (0)	57.6 (0)	**100** (100)
T5(OOV): Full dialogs	*100 (100)*	4.6 (0)	9.0 (0)	48.4 (0)	58.2 (0)	65.5 (0)	**77.7** (0)
T6: Dialog state tracking 2	33.3 (0)	1.6 (0)	1.6 (0)	21.9 (0)	22.6 (0)	**41.1** (0)	**41.0** (0)
Concierge(*)	n/a	1.1 (0.2)	n/a	13.4 (0.5)	14.6 (0.5)	**16.7** (1.2)	n/a(†)

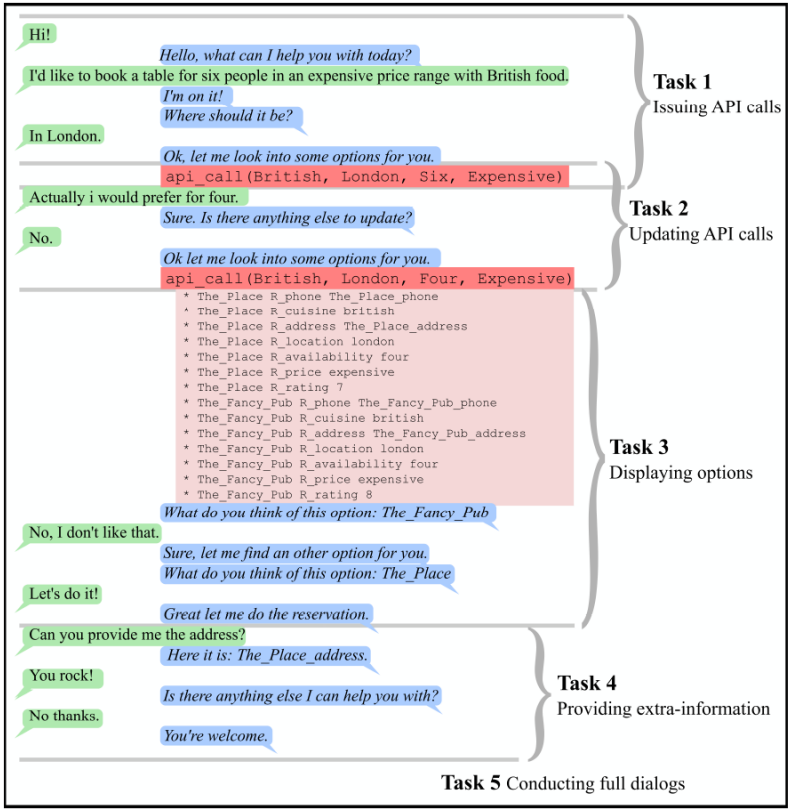
Match type = extend entity descriptions with their type (cuisine type, location, price range, party size, rating, phone number and address) to help match OOV items.
Supervised embedding = task-specific word embedding, using a margin loss on a prediction task.

Bordes et al “LEARNING END-TO-END GOAL-ORIENTED DIALOG” ICML 2017

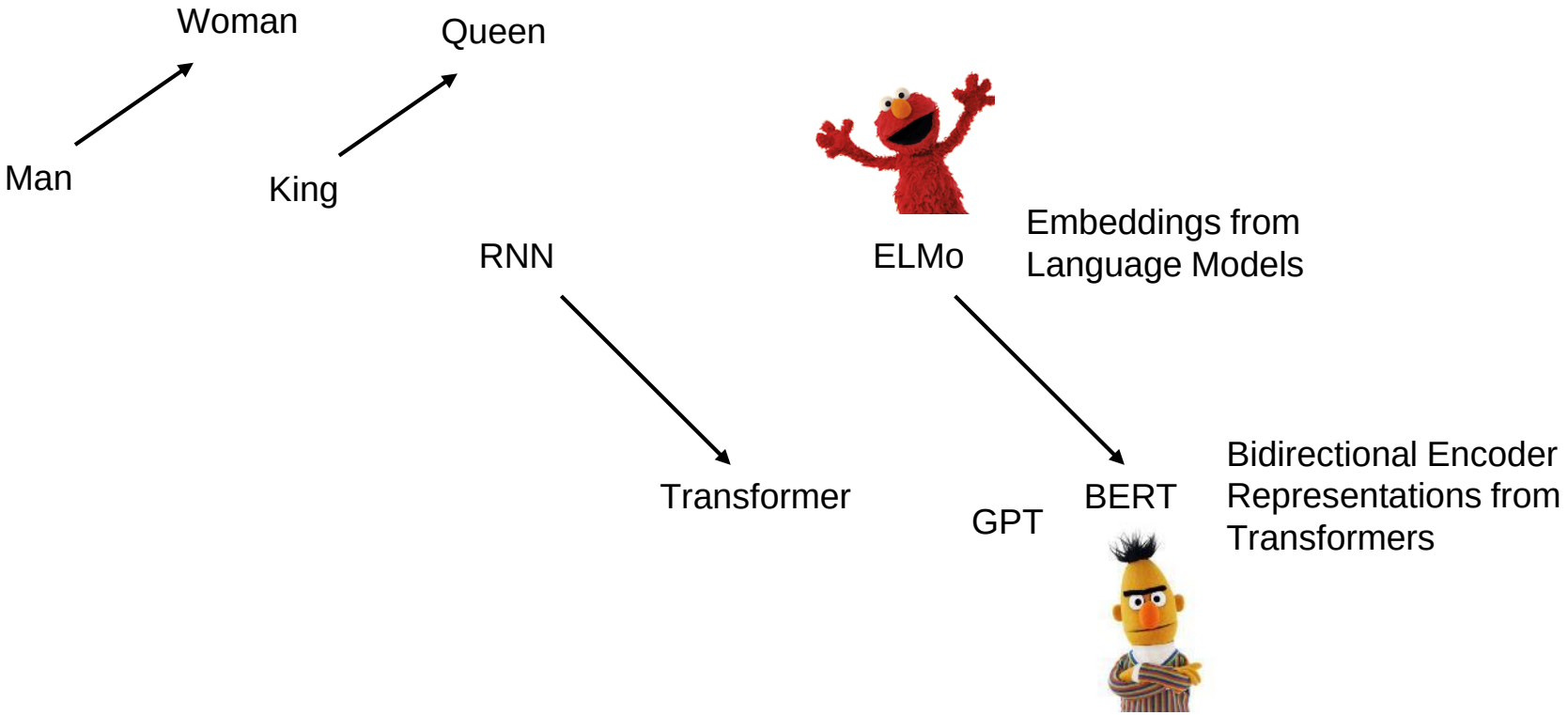
Experiments

Ground Truth = held out transcripts.

Note “display options” task includes user feedback:



BERT and Dialog



BERT

BERT is a language model (next word predictor) trained on a large dataset of natural language that is fine-tuned for particular tasks.

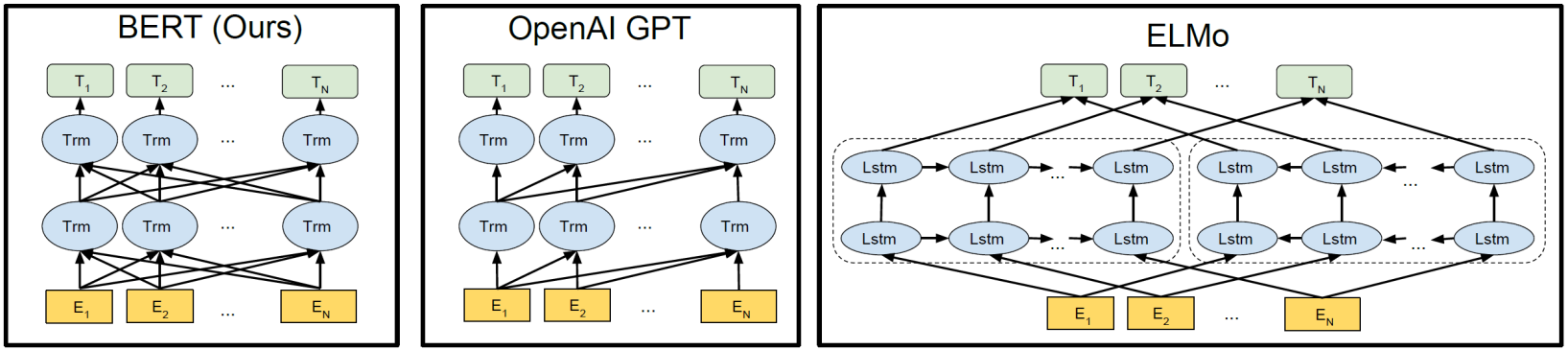
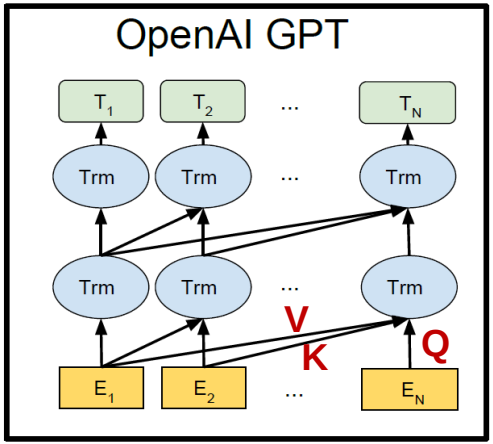


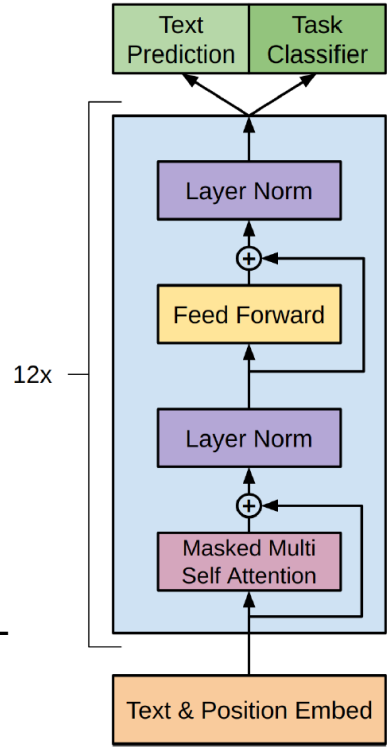
Figure 1: Differences in pre-training model architectures. BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to-left LSTM to generate features for downstream tasks. Among three, only BERT representations are jointly conditioned on both left and right context in all layers.

But note that only GPT is a generative model.

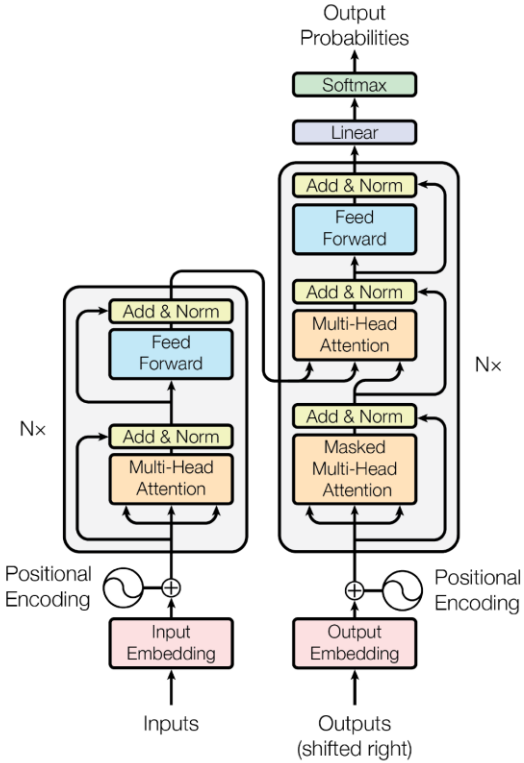
GPT

Generative Pre-Training (OpenAI) is a transformer-based generator using only a simplified transformer decoder stage:





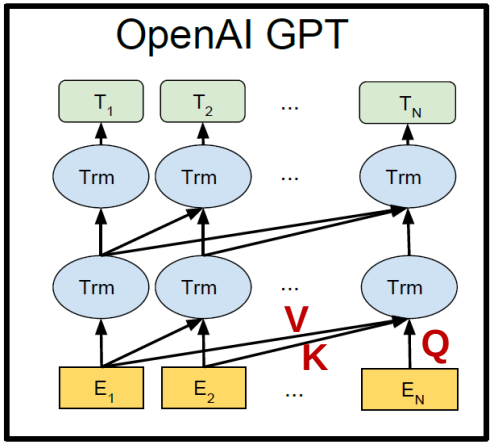
GPT

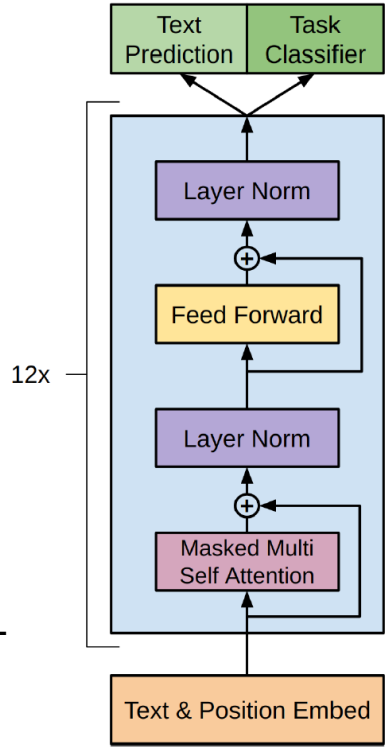


Full transformer

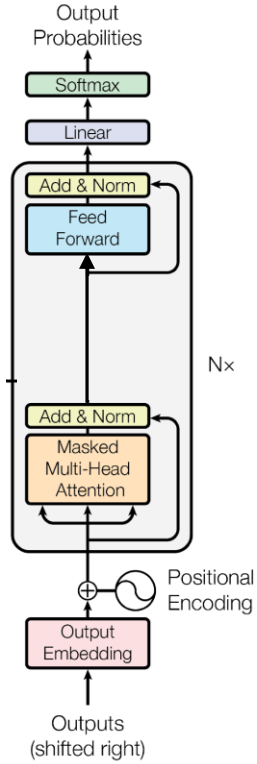
GPT

Generative Pre-Training (OpenAI) is a transformer-based generator using only a simplified transformer decoder stage:





GPT



Full transformer

GPT

GPT is trained initially on a large text corpus to minimize a language modeling loss

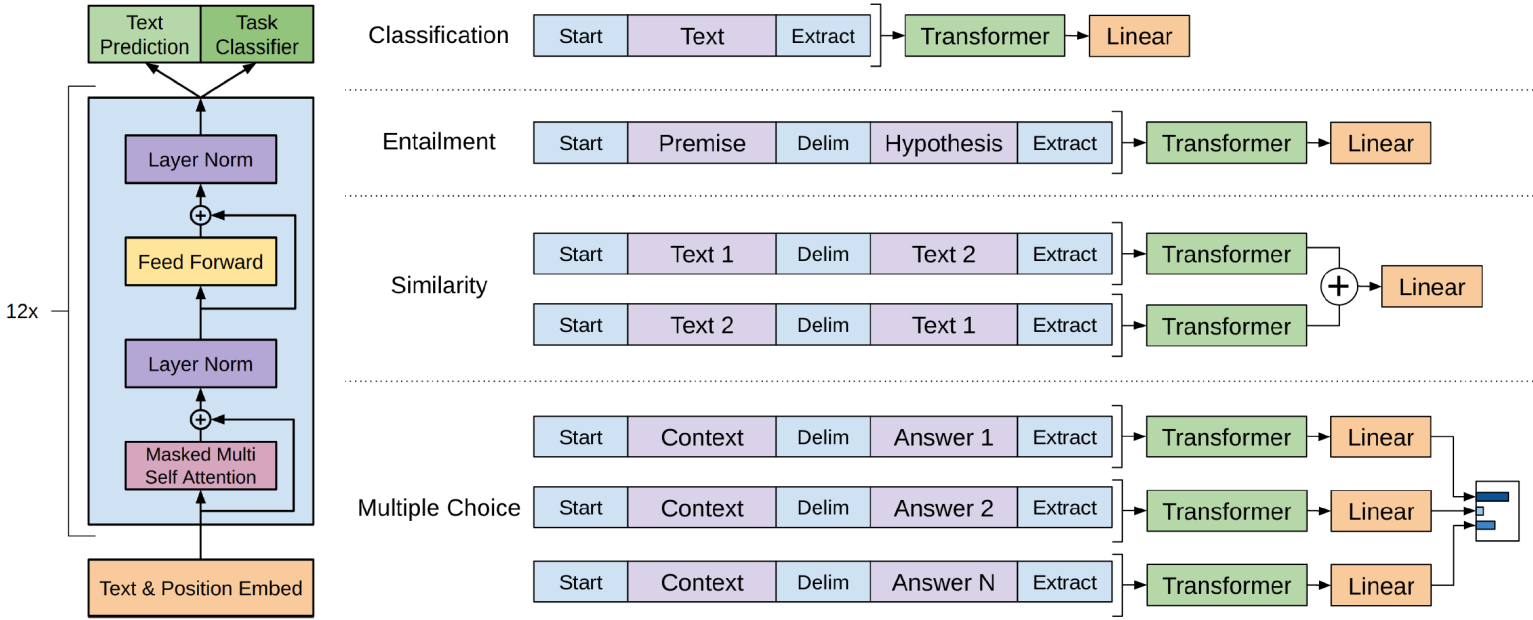
$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

Then for each task, a custom linear layer is added and the entire network retrained (with slower adjustment of the transformer weights).

The language model loss is retained during task-specific retraining of the model.

GPT

Generative Pre-Training can be applied to a variety of (non-generative) tasks:



GPT Task performance

Entailment tasks (predict entailment, contradiction, or neutral):
“Anne drove to work” → “Anne has a job”

if a person would say B is probably true given A.

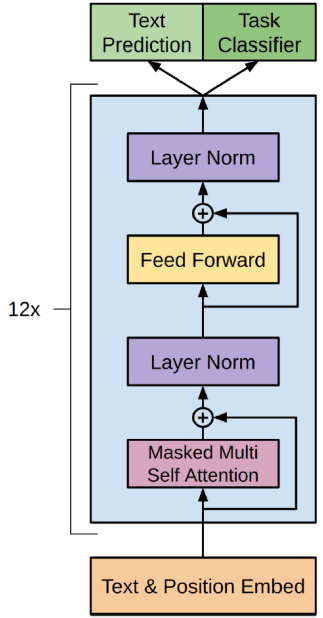


Table 2: Experimental results on natural language inference tasks, comparing our model with current state-of-the-art methods. 5x indicates an ensemble of 5 models. All datasets use accuracy as the evaluation metric.

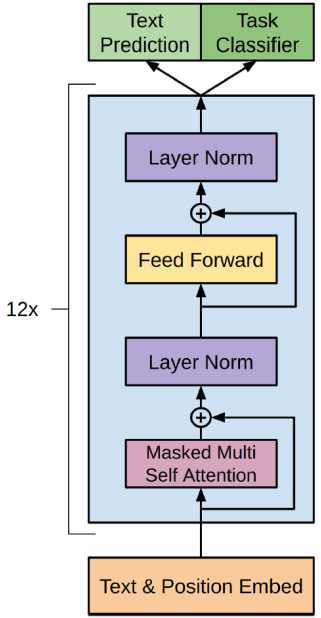
Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	89.3	-	-	-
CAFE [58] (5x)	80.2	79.0	89.3	-	-	-
Stochastic Answer Network [35] (3x)	80.6	80.1	-	-	-	-
CAFE [58]	78.7	77.9	88.5	83.3		
GenSen [64]	71.4	71.3	-	-	82.3	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	**61.7**
Finetuned Transformer LM (ours)	**82.1**	**81.4**	**89.9**	**88.3**	**88.1**	56.0

GPT Task performance

Question answering and commonsense reasoning:
RACE contains questions from high-school and middle-school exams.
Story Cloze is story completion

Table 3: Results on question answering and commonsense reasoning, comparing our model with current state-of-the-art methods.. 9x means an ensemble of 9 models.

Method	Story Cloze	RACE-m	RACE-h	RACE
val-LS-skip [55]	76.5	-	-	-
Hidden Coherence Model [7]	77.6	-	-	-
Dynamic Fusion Net [67] (9x)	-	55.6	49.4	51.2
BiAttention MRU [59] (9x)	-	60.2	50.3	53.3
Finetuned Transformer LM (ours)	**86.5**	**62.9**	**57.4**	**59.0**

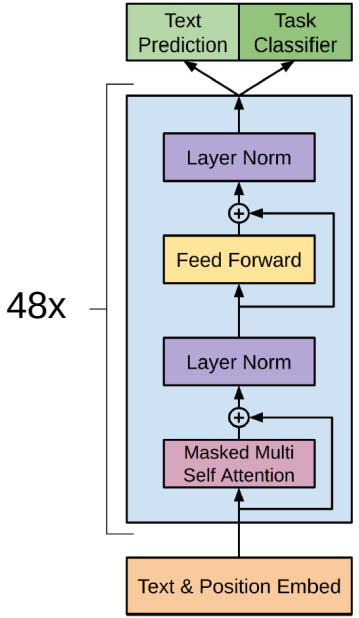


GPT2

A depth-48 version of GPT trained on a massive dataset (WebText):

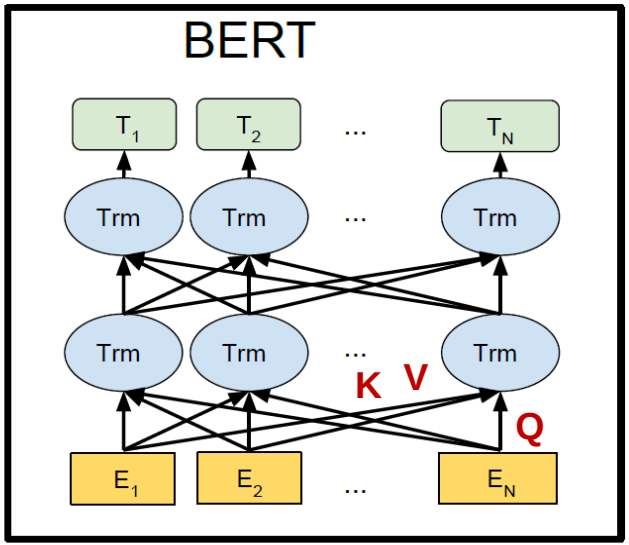
A train carriage containing controlled nuclear materials was stolen in Cincinnati today. Its whereabouts are unknown.

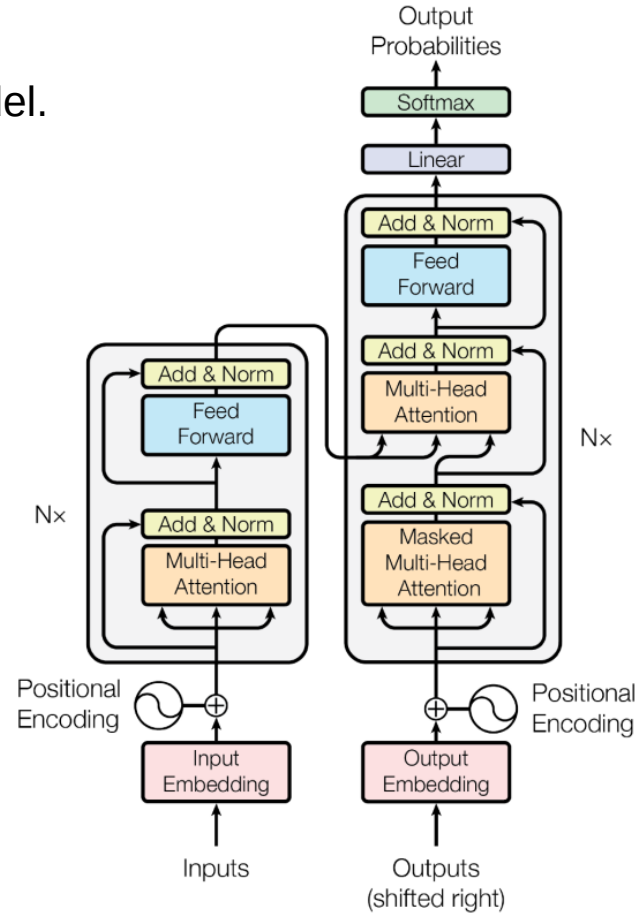
In an email to Ohio news outlets, the U.S. Department of Energy said it is working with the Federal Railroad Administration to find the thief.
“The theft of this nuclear material will have significant negative consequences on public and environmental health, our workforce and the economy of our nation,” said Tom Hicks, the U.S. Energy Secretary, in a statement. “Our top priority is to secure the theft and ensure it doesn’t happen again.”
The stolen material was taken from the University of Cincinnati’s Research Triangle Park nuclear research site, according to a news release from Department officials. The Nuclear Regulatory Commission did not immediately release any information.
According to the release, the U.S. Department of Energy’s Office of Nuclear Material Safety and Security is leading that team’s investigation.
“The safety of people, the environment and the nation’s nuclear stockpile is our highest priority,” Hicks said. “We will get to the bottom of this and make no excuses.”



BERT

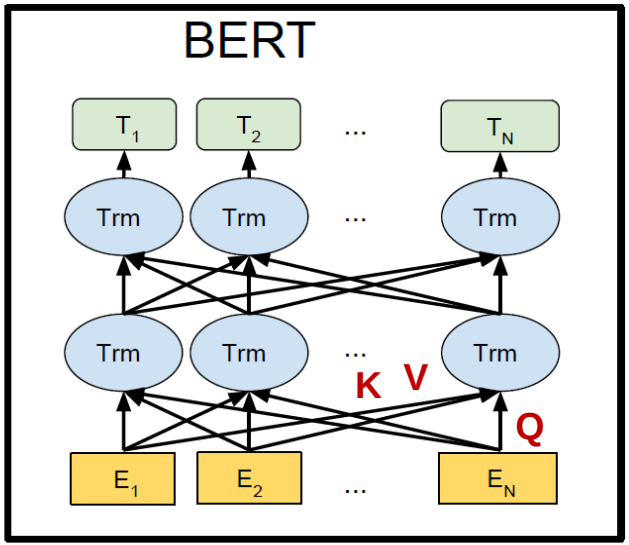
BERT is a bidirectional (Transformer encoder) model.

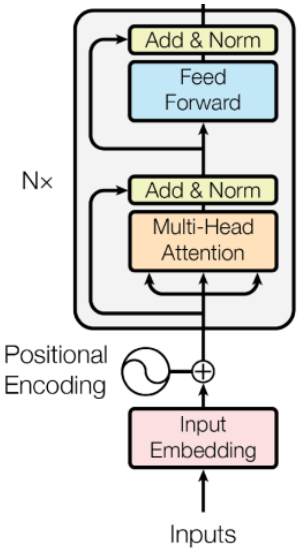




BERT

BERT is a bidirectional (Transformer encoder) model.

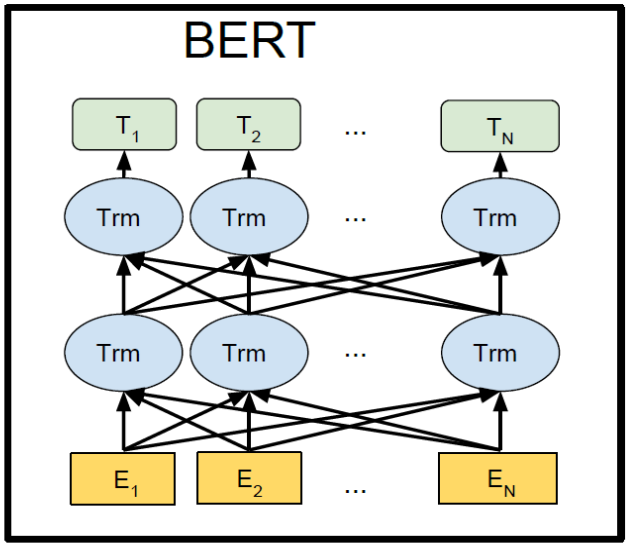


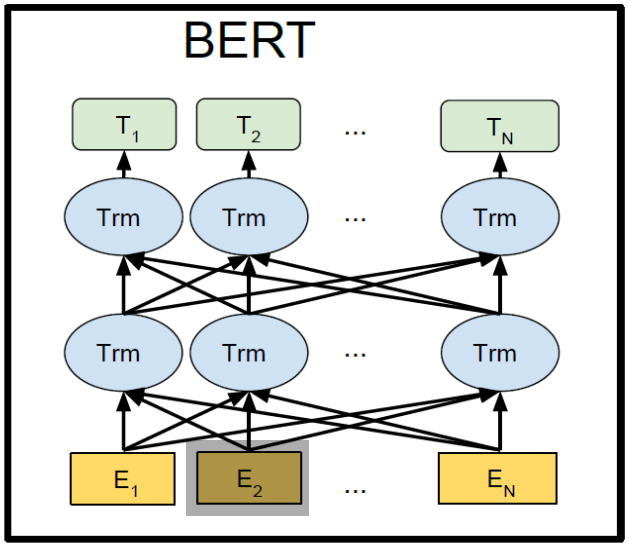


BERT

BERT is trained with two types of loss:

- Word prediction: 15% of input words are removed and then re-predicted





BERT

BERT is trained with two types of loss:

- Next sentence prediction: from an actual corpus of consecutive sentence pairs, create a dataset with 50% real pairs, and 50% “fake” pairs (where the second sentence is a random one).

$$\text{Input} = \texttt{[CLS] the man went to [MASK] store [SEP]}$$
$$\texttt{he bought a gallon [MASK] milk [SEP]}$$
$$\text{Label} = \texttt{IsNext}$$

$$\text{Input} = \texttt{[CLS] the man [MASK] to the store [SEP]}$$
$$\texttt{penguin [MASK] are flight ##less birds [SEP]}$$
$$\text{Label} = \texttt{NotNext}$$

BERT

BERT is trained with two types of loss:

- Next sentence prediction: from an actual corpus of consecutive sentence pairs, create a dataset with 50% real pairs, and 50% “fake” pairs (where the second sentence is a random one).

- This is an example of a very general loss for unsupervised learning called “contrastive loss”. The loss contrasts true positives with “near miss” negatives.

BERT Task specialization

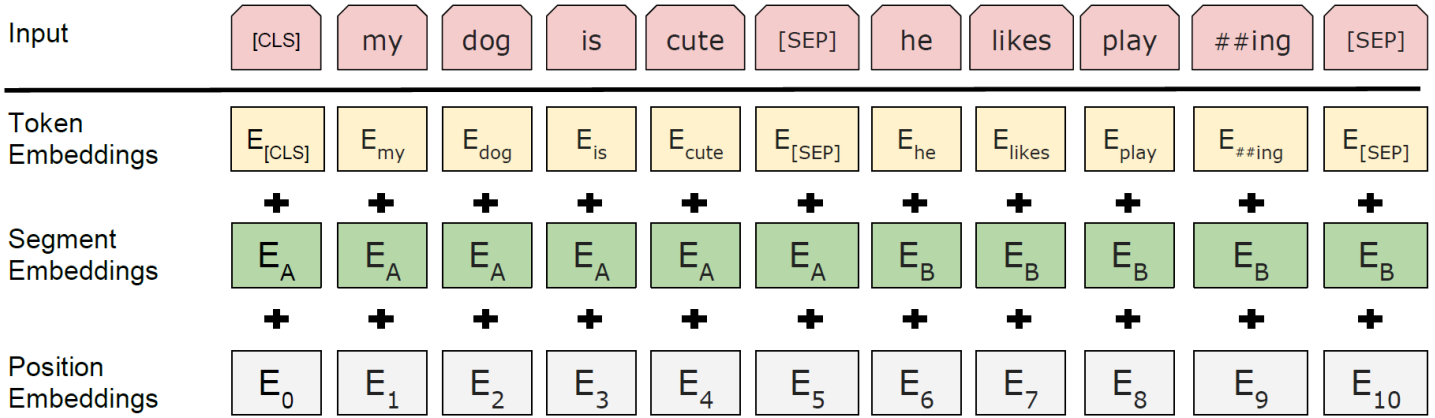
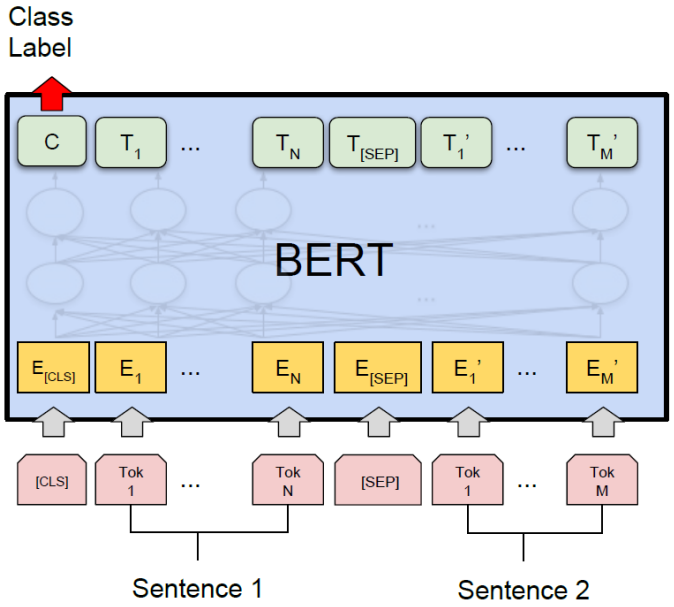
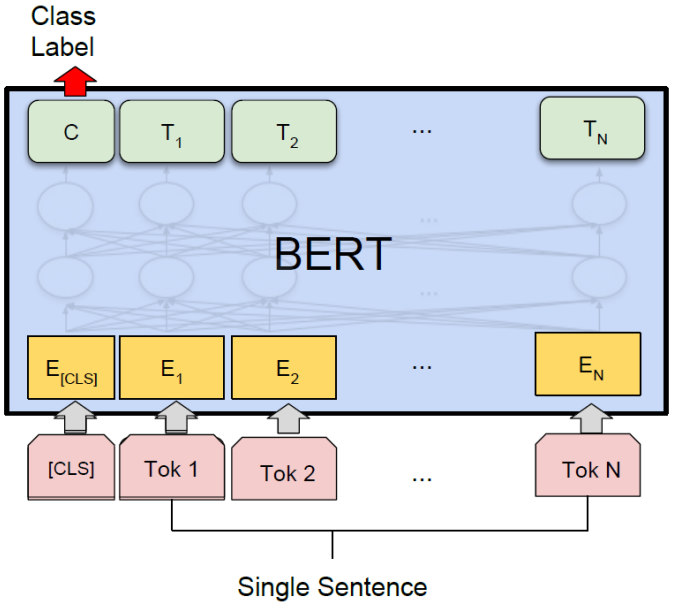


Figure 2: BERT input representation. The input embeddings is the sum of the token embeddings, the segmentation embeddings and the position embeddings.

BERT Task specialization

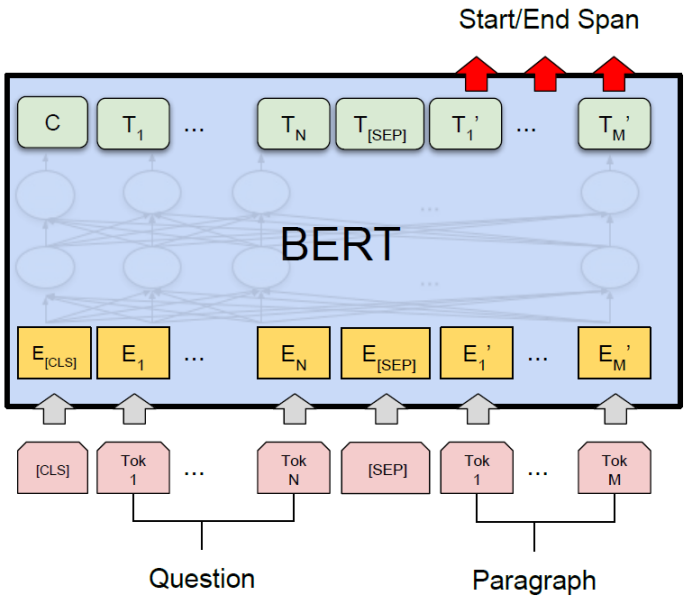


(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

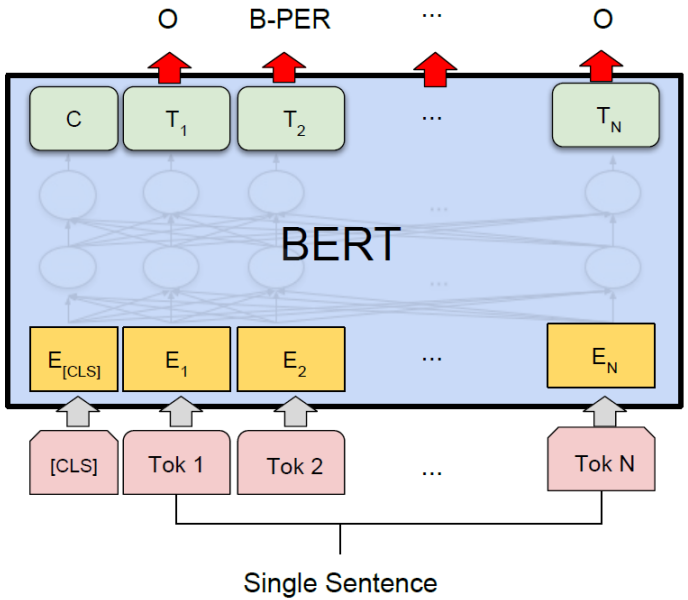


(b) Single Sentence Classification Tasks: SST-2, CoLA

BERT Task specialization



(c) Question Answering Tasks:
SQuAD v1.1



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

BERT Performance

BERT Base: L=12, H=768, A=12, total param =110M
BERT Large: L=24, H=1024, A=16, total param=340M

L=number of layers, H=model dimension, A=number of multi-attention heads

Glue tasks

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
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**91.1**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**81.9**

BERT Performance

SQuAD

System	Dev		Test	
	EM	F1	EM	F1
Leaderboard (Oct 8th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
#1 Single - nlnet	-	-	83.5	90.1
#2 Single - QANet	-	-	82.5	89.3
Published				
BiDAF+ELMo (Single)	-	85.8	-	-
R.M. Reader (Single)	78.9	86.3	79.5	86.6
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

Dialog

DSTC = Dialog System Technology Challenge:

- Sentence selection
- Sentence generation
- Audio-visual scene-aware dialog

Sample tasks for DSTC

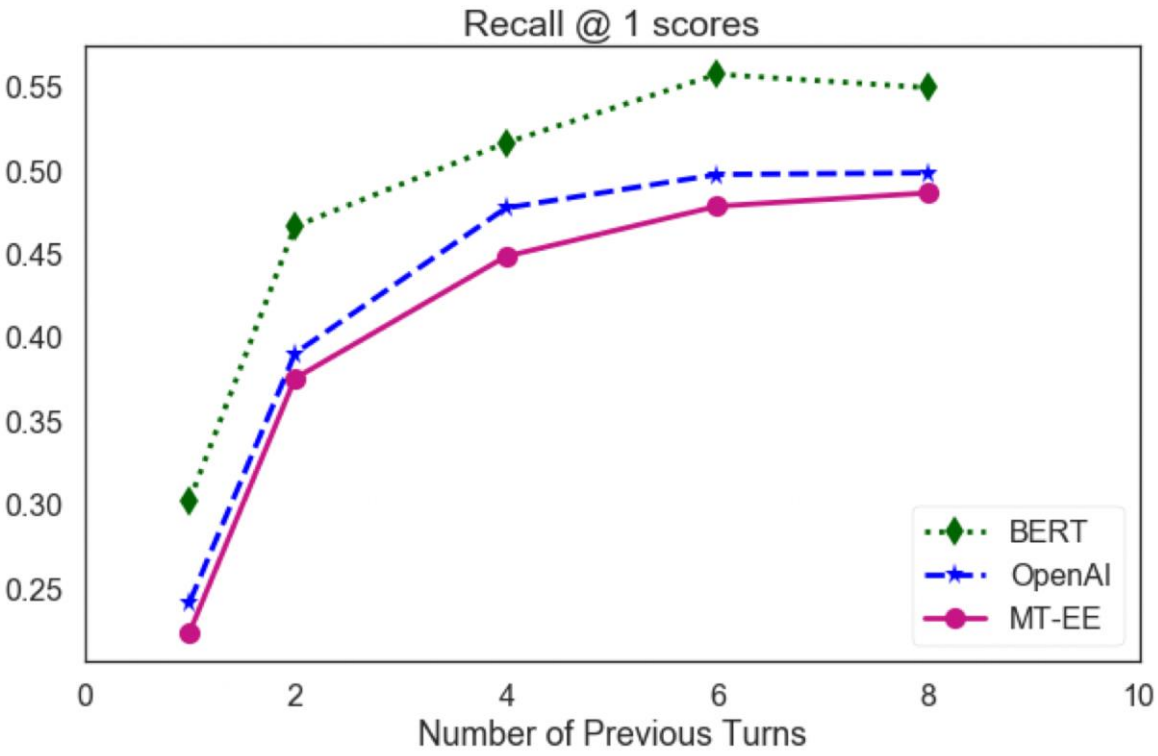
ADVISOR | Hi! What can I help you with?
STUDENT | Hello! I'm trying to schedule classes for next semester. Can you help me?
STUDENT | Hardware has been an interest of mine.
STUDENT | But I don't want too hard of classes
ADVISOR | So are you interested in pursuing Electrical or Computer Engineering?
STUDENT | I'm undecided
STUDENT | I enjoy programming but enjoy hardware a little more.
ADVISOR | Computer Engineering consists of both programming and hardware.
ADVISOR | I think it will be a great fit for you.
STUDENT | Awesome, I think that's some good advice.
STUDENT | What classes should I take to become a Computer Engineer?
ADVISOR | You haven't taken EECS 203, 280, and 270, so it may be in your best interest to take one or two of those classes next semester
STUDENT | Ok. Which of those is in the morning. I like morning classes

Sample tasks for DSTC

[13:11] <user_1> anyone here know memcached?
[13:12] <user_1> trying to change the port it runs on
[13:12] <user_2> user_1: and ?
[13:13] <user_1> user_2: I'm not sure where to look
[13:13] <user_1> !
[13:13] <user_2> user_1: /etc/memcached.conf ?
[13:13] <user_1> haha
[13:13] <user_1> user_2: oh yes, it's much simpler than I thought
[13:13] <user_1> not sure why, I was trying to work through the init.d stuff

Should also use external reference information from unix man pages.

DSTC7 results



CoQA

https://stanfordnlp.github.io/coqa/

CoQA contains 127,000+ questions with answers collected from 8000+ conversations. Each conversation is collected by pairing two crowdworkers to chat about a passage in the form of questions and answers.

CoQA Leaderboard 3/20/19

Leaderboard

Rank	Model	In-domain	Out-of-domain	Overall
	Human Performance *Stanford University* (Reddy et al. '18)	89.4	87.4	88.8
1 Jan 25, 2019	BERT + MMFT + ADA (ensemble) *Microsoft Research Asia*	**87.5**	**85.3**	**86.8**
2 Jan 21, 2019	BERT + MMFT + ADA (single model) *Microsoft Research Asia*	86.4	81.9	85.0
3 Jan 03, 2019	BERT + Answer Verification (single model) *Sogou Search AI Group*	83.8	80.2	82.8
4 Jan 06, 2019	BERT with History Augmented Query (single model) *Fudan University NLP Lab*	82.7	78.6	81.5
5 Jan 31, 2019	BERT Large Finetuned Baseline (single model) *Anonymous*	82.6	78.4	81.4
6 Jan 21, 2019	BERT Large Augmented (single model) *Microsoft Dynamics 365 AI Research*	82.5	77.6	81.1
7 Dec 12, 2018	D-AoA + BERT (single model) *Joint Laboratory of HIT and iFLYTEK Research*	81.4	77.3	80.2

Takeaways

- Pre-training language models on very large corpora improves the state-of-the-art for multiple NLP tasks (ELMo, GPT, BERT).
- Transformer designs (GPT and BERT) have superseded RNN designs (ELMo).
- Single-task execution involves input encoding, a small amount of output hardware, and fine-tuning.
- BiDirectional encoder models (BERT) do better than generative models (GPT) at non-generation tasks, for comparable training data/model complexity.
- Generative models have training efficiency and scalability advantages that may make them ultimately more accurate. They can also solve entirely new kinds of task that involve text generation.