Big-model-based Text Understanding and Generation

Ganqu Cui, Si Sun, Fengyu Wang

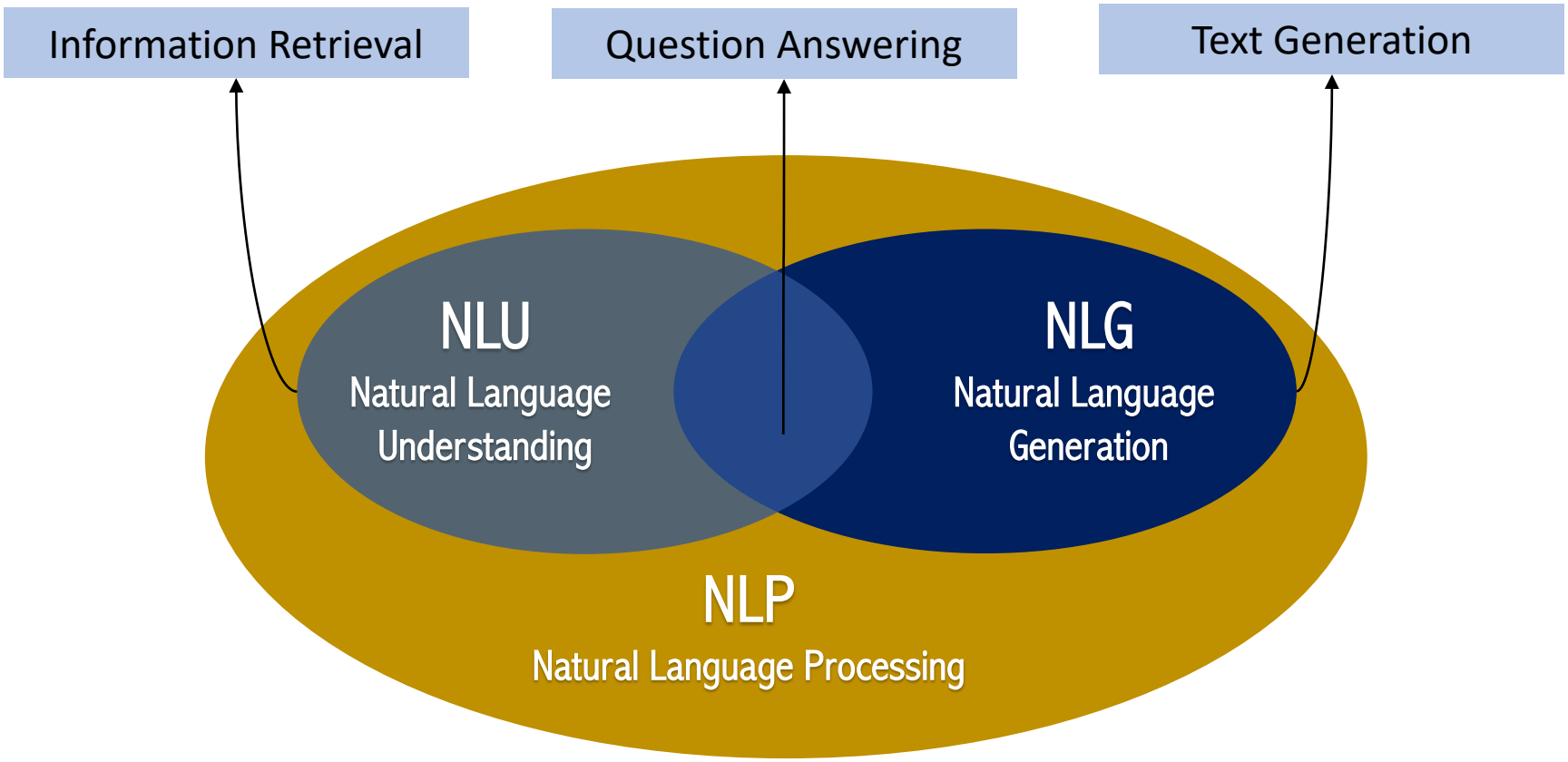
THUNLP

Outline

- Introduction
- Information Retrieval
- Question Answering
- Text Generation

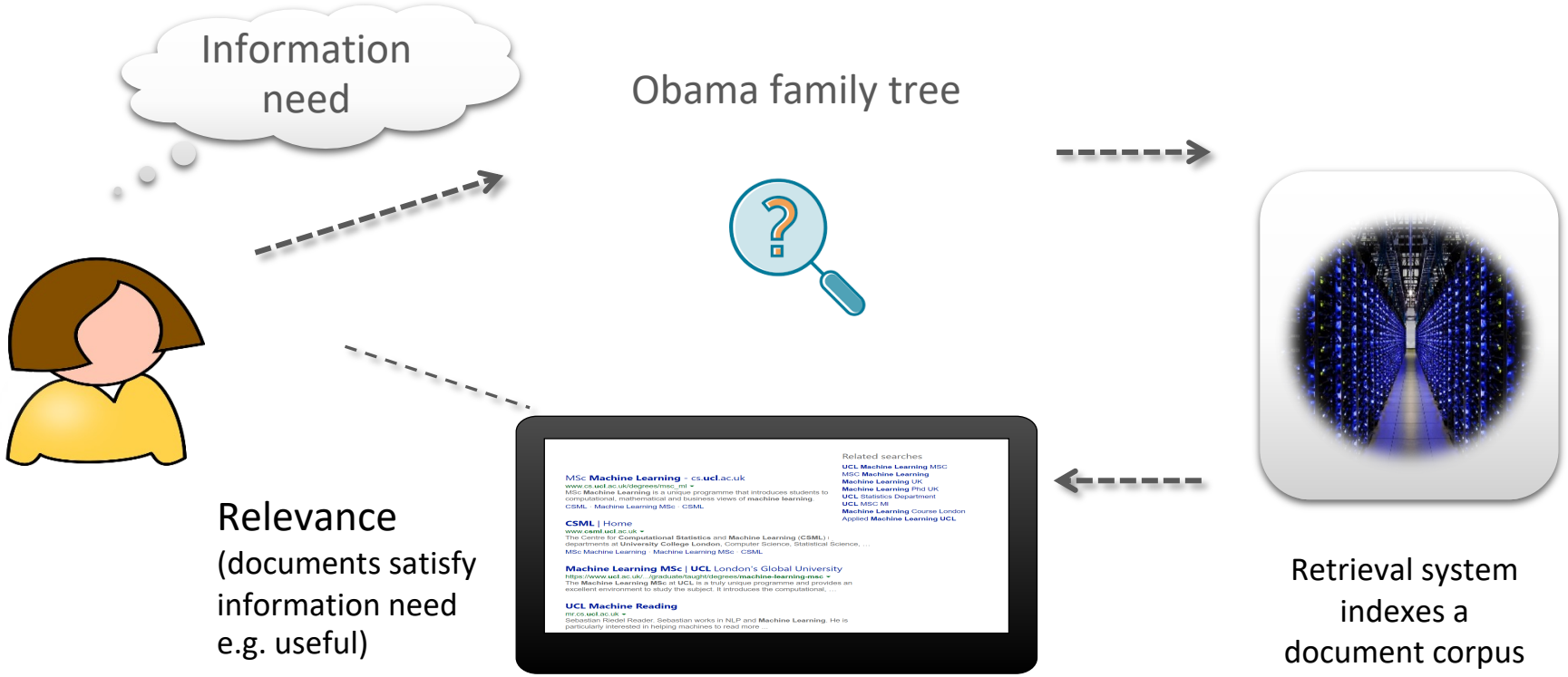
Introduction

- Typical NLP applications: understanding and generation
- Big models bring revolutions



Introduction

- Information retrieval
  - Find relevant documents given queries

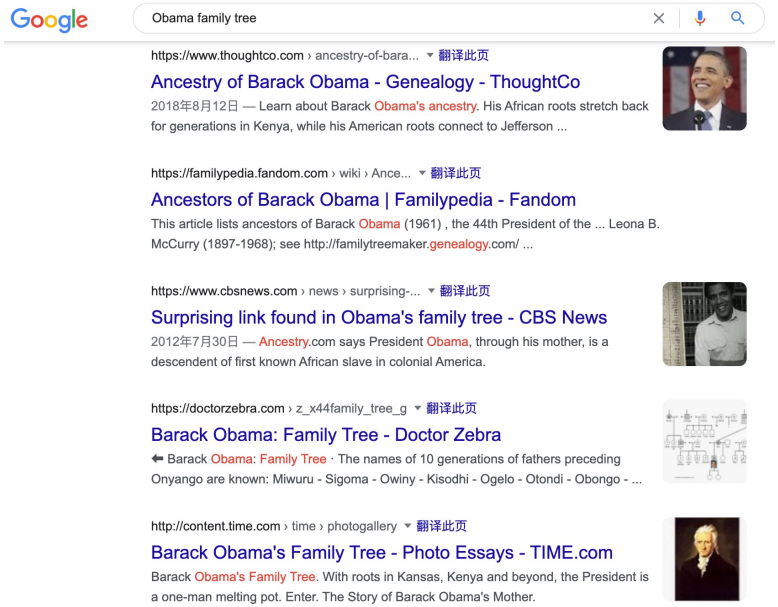


Relevance between text queries and documents

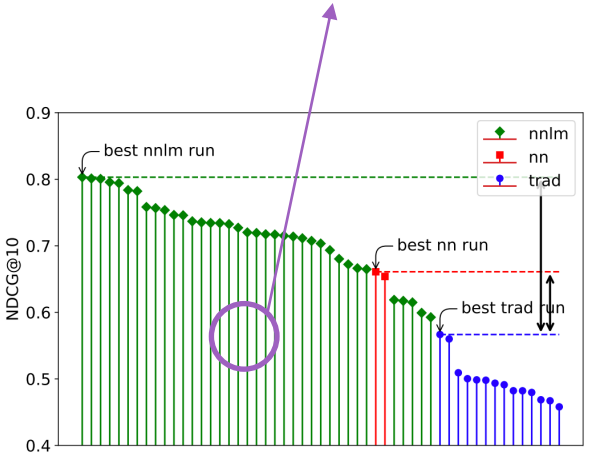
Introduction

- Information retrieval
  - Big models can provide more intelligent and accurate search results

Query: Obama family tree
Ranked Document List:



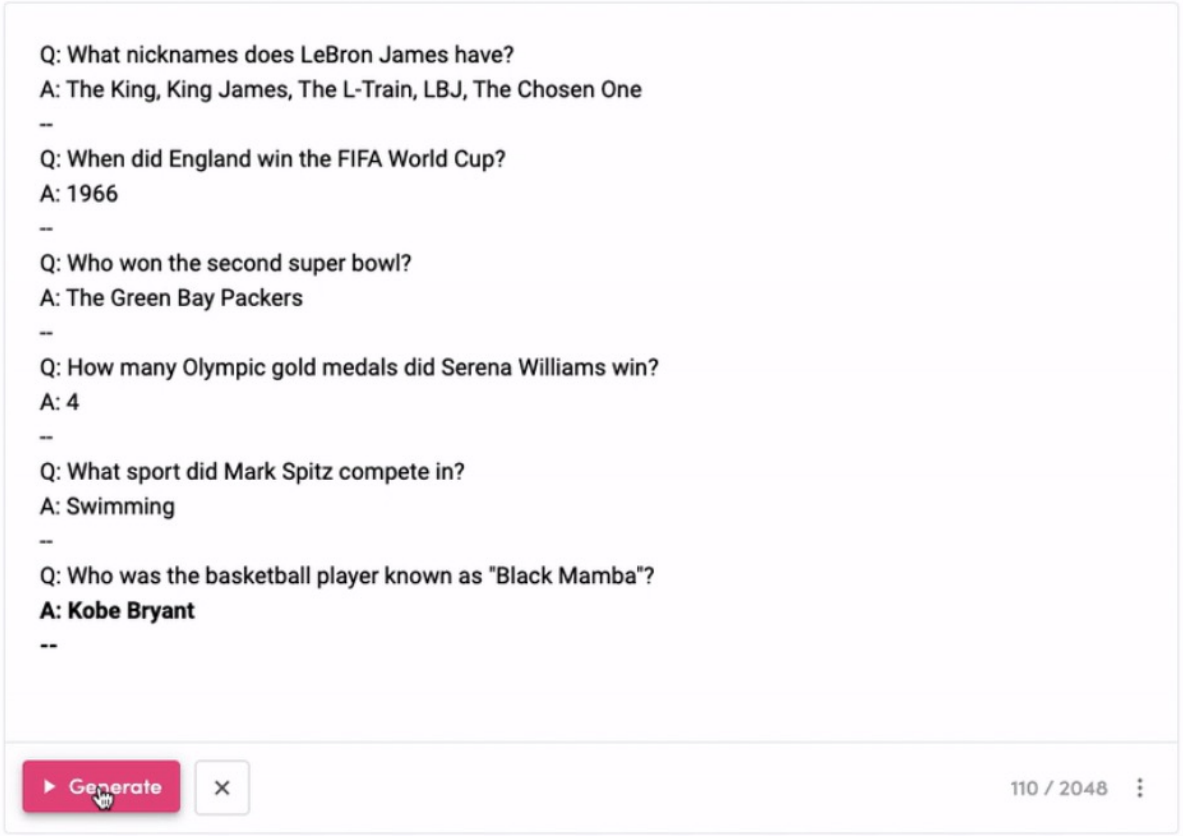
PLM-based methods ranked high



Passage retrieval task
(TREC 2020 Deep Learning Track)

Introduction

- Question answering
  - Big models can answer more complex questions



https://www.ai21.com/blog/ai21-studio-use-cases

Introduction

- Text generation
  - Machine translation; poetry generation; dialogue systems…
  - Big models can generate more fluent and natural texts

http://jiuge.thunlp.org/

Information Retrieval (IR)

Si Sun

s-sun17@mails.tsinghua.edu.cn

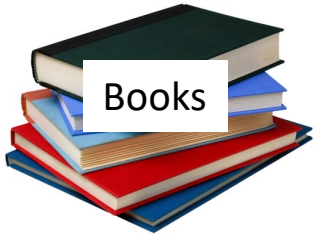
THUNLP

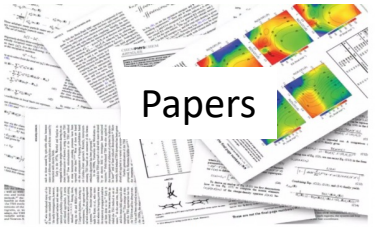
Outline

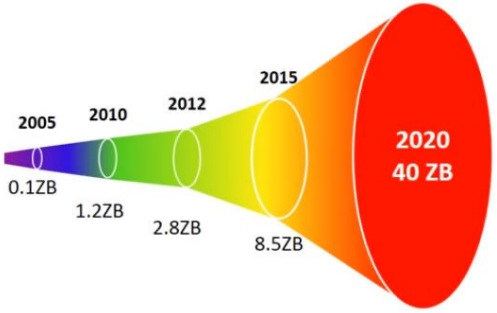
- IR Background
- IR Formulation
- Traditional IR
- Neural IR (Big Model)
- Advanced Topics

Background

- Information explosion
  - Amount
    - 40ZB, 50% annual growth rate
  - Variety
    - Update period in minutes





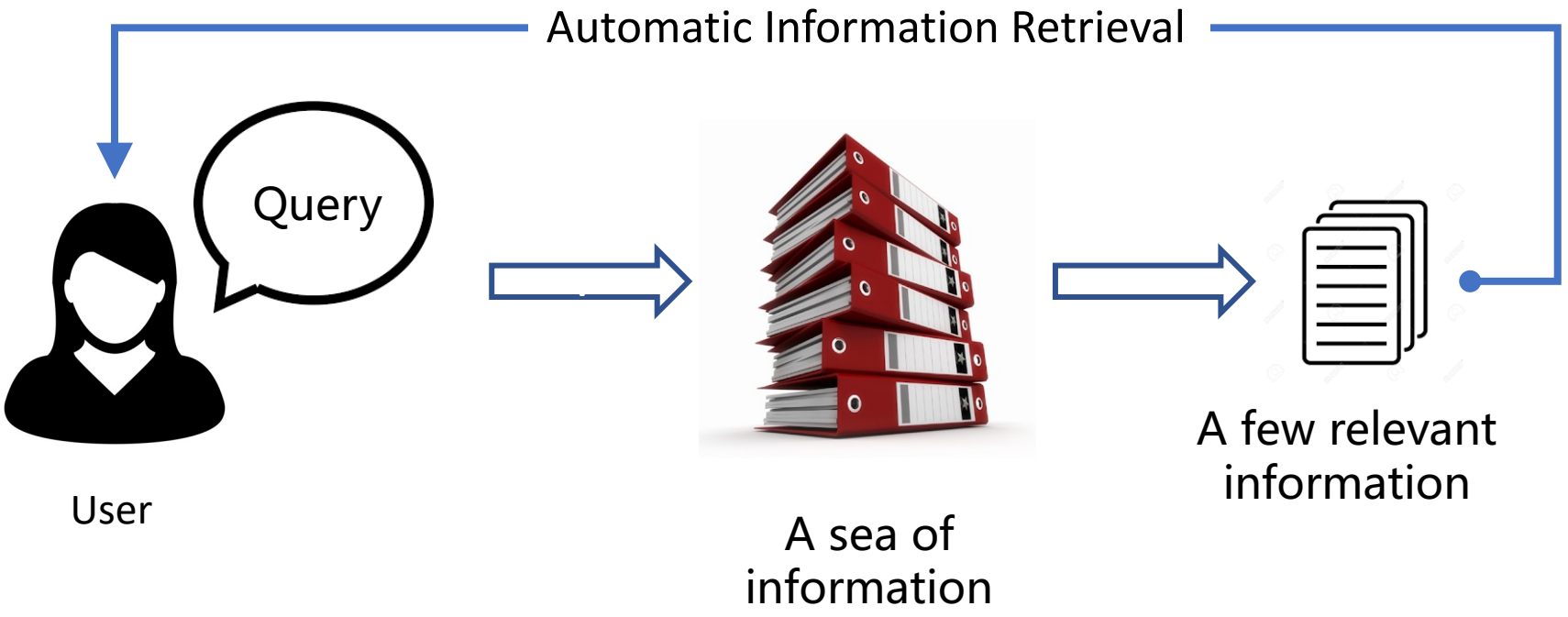


(IDC 2020)

Background

- Rising demand for automatic information retrieval
  - 4.39 billion information users
  - Annual growth rate of 6~21%

(IDC 2020)



Application

- Typical application

Search Engine

Public opinion analysis / Fact verification

QA system

Retrieval-Augment Text Generation

Application

- Examples
  - Document Ranking

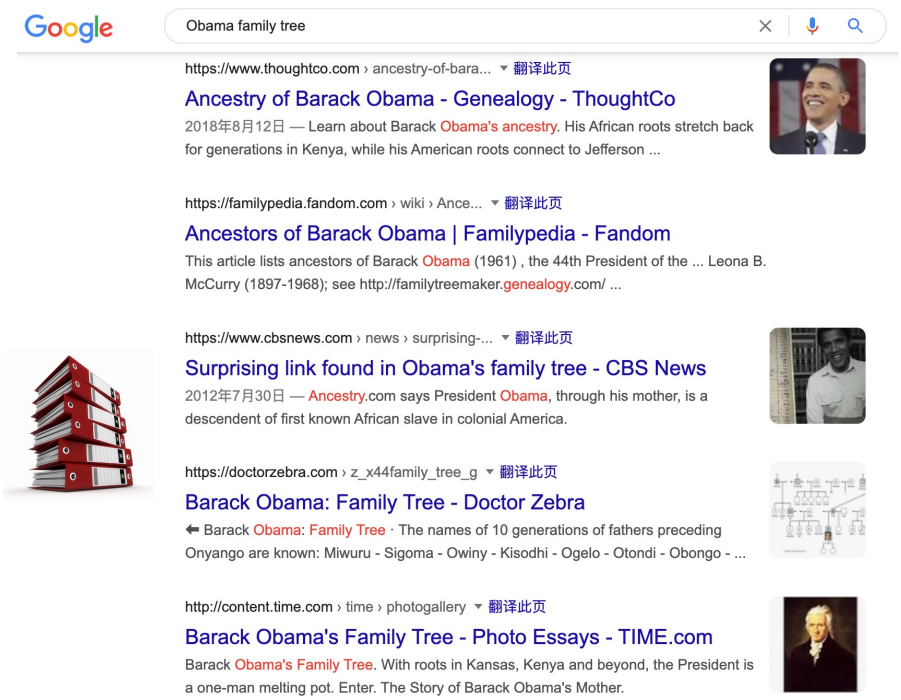
    Query: Obama family tree

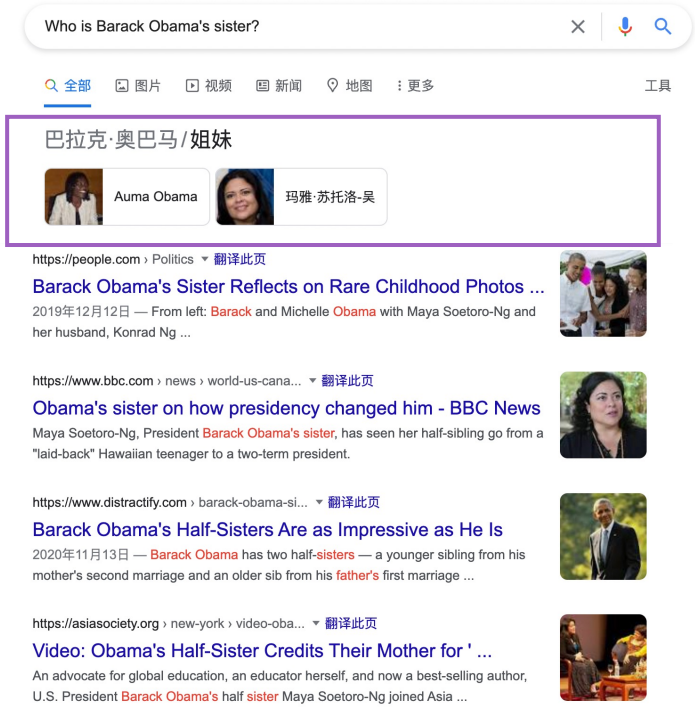
    Ranked Document List:

  - Question Answering

    Query: Who is Barack Obama's sister?

    Answer:



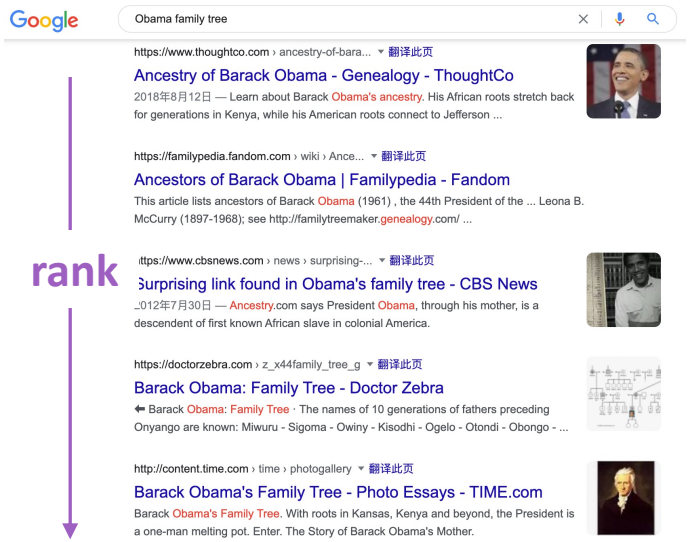


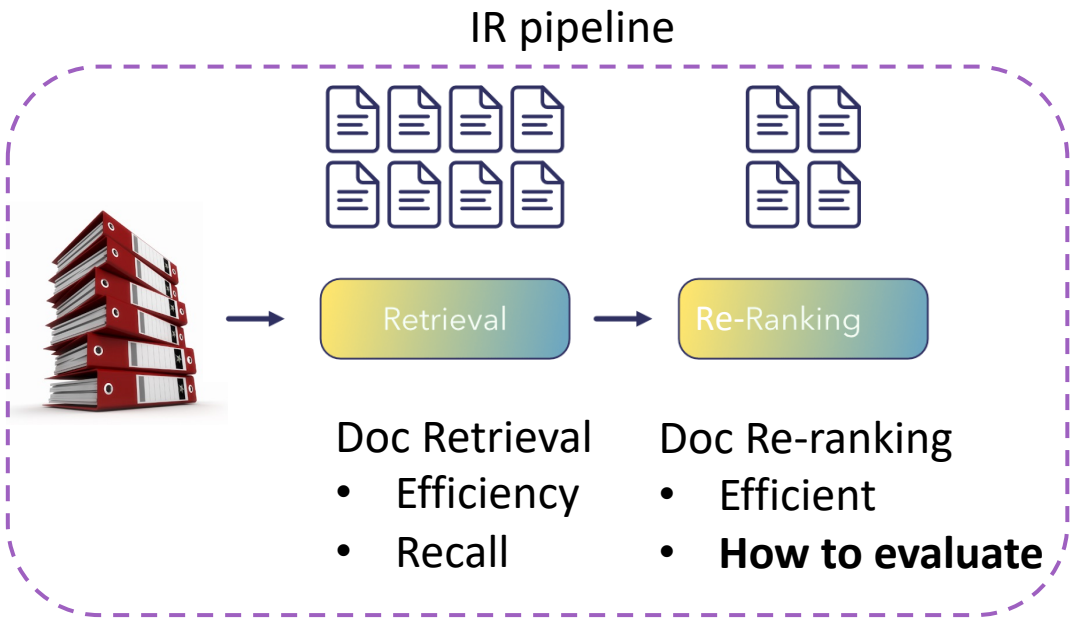
Outline

- IR Background
- IR Formulation
  - How to formulate
  - How to evaluate
- Traditional IR
- Neural IR (Big Model)
- Advanced Topics

IR Formulation

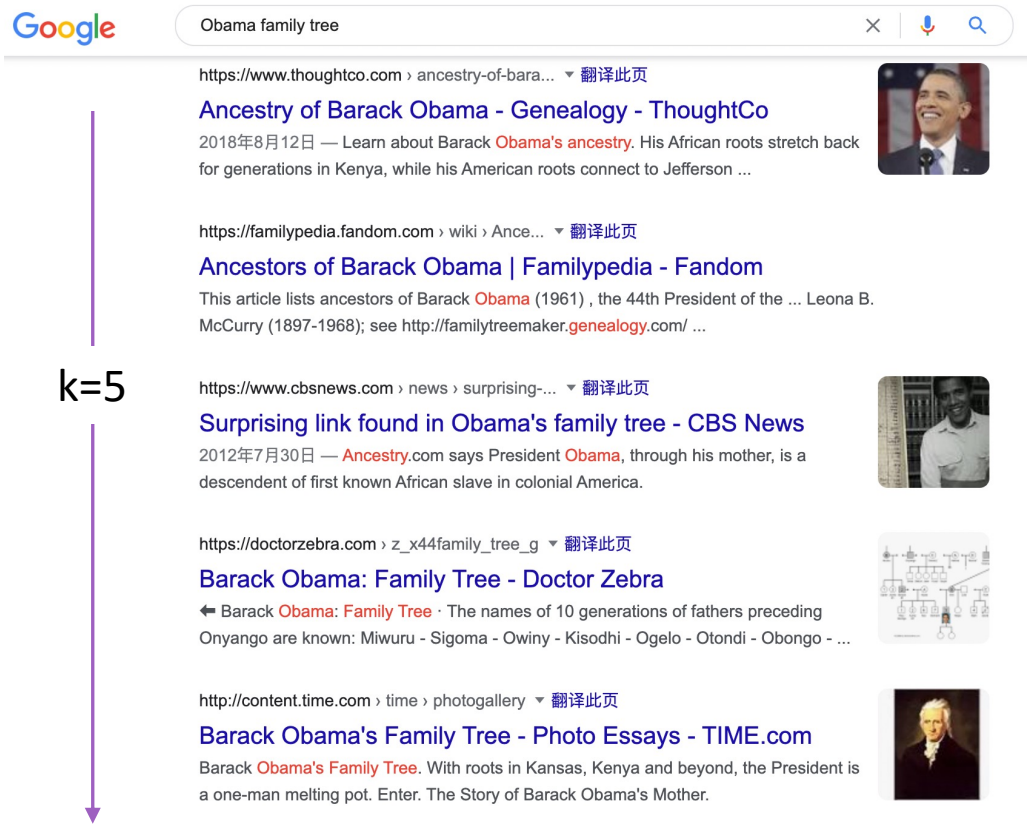
- How to formulate?
  - Given a query q
  - Given a document collection $D = \{\dots, d_i, \dots\}$
  - IR system computes the relevance score $f(q, d_i)$ and **ranks** all documents based on the scores





Evaluation Metrics

- Widely-used metrics
  - MRR@k
  - MAP@k
  - NDCG@k



Evaluation Metrics

- MRR (Mean Reciprocal Rank)
  - MRR is the average of the reciprocal ranks of the **first** relevant results for a query set Q:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}$$

  - For example:

$$MRR = (1/3 + 1/2 + 1) / 3 = 0.61$$

Query	Search Results	$\frac{1}{rank_i}$
cat	catten, cati, **cats, …**	1/3
torus	torii, **tori**, toruses, …	1/2
virus	**viruses**, virii, viri, ….	1

Evaluation Metrics

- MAP (Mean Average Precision)
  - MAP is the mean of the average precision score for a set of queries.

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}$$ where Q is the number of queries

  - Suppose you have two queries:

Query1: **Four related docs**. Four related docs were retrieved and their ranks are **1, 2, 4 and 7**

$$AveP(Q1) = (1/1 + 2/2 + 3/4 + 4/7)/4 = 0.83$$

Query2: **Five related docs**. Three related docs were retrieved and their ranks are **1, 3 and 5**

$$AveP(Q2) = (1/1 + 2/3 + 3/5 + 0 + 0)/5 = 0.45$$

$$MAP = (0.83 + 0.45) / 2 = 0.64$$

Evaluation Metrics

- NDCG (Normalized Discounted Cumulative Gain)
  - divides docs into different levels according to the relevance with the query

$$\text{NDCG} = \frac{DCG}{IDCG}$$

Normalize

Ideal DCG (IDCG): The DCG value of gold-standard relevant documents (ordered by their ideal rank)

Gain

$$\text{DCG} = \sum_{i=1}^{n} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

Cumulate

Discount

Where n is the number of docs, i is the rank position of document and rel_i is the grade of the doc

Evaluation Metrics

- Discounted Cumulative Gain (DCG)
  - You get five results for a query search and classify them into three grades: Good (3), Fair (2) and Bad (1)
  - The grade values of these five results are $rel_1 = 3$, $rel_2 = 1$, $rel_3 = 2$, $rel_4 = 3$, and $rel_5 = 2$

DCG = 7 * 1 + 1 * 0.63 + 3 * 0.50 + 7 * 0.43 + 3 * 0.39 = 13

Rank	rel_i	$2^{rel_i} - 1$	$\frac{1}{log_2(i+1)}$
1	3	7	1
2	1	1	0.63
3	2	3	0.50
4	3	7	0.43
5	2	3	0.39

$$DCG = \sum_{i=1}^{n} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

Outline

- IR Background
- IR Formulation
- Traditional IR
  - BM25
  - Problems
- Neural IR (Big Model)
- Advanced Topics

Traditional IR

- BM25 (Best Matching 25)
  - Lexical exact-match model
  - Given a query $q = \{w_1, \dots, w_k, \dots, w_m\}$ and a document collection $D = \{\dots, d_i, \dots\}$
  - BM25 computes the relevance score $f(q, d_i)$ as:

$$f(q, d_i) = \sum_{i=1}^{n} \boldsymbol{IDF(q_i)} \cdot \frac{\boldsymbol{TF(q_i, d_i)} \cdot (k+1)}{TF(q_i, d_i) + k \cdot \left(1 - b + b \cdot \frac{|d_i|}{avgdl}\right)}$$

Where k and b are hyper-parameters, $|d_i|$ is the length of d_i, and $avgdl$ is the average document length in the document collection

Peilin Yang, et al. SIGIR 2017. Anserini: Enabling the use of lucene for information retrieval research.

Traditional IR

- TF (Term Frequency)
  - The weight of a term that occurs in a document is simply proportional to the term frequency
  - The number of times that term t occurs in document d:

$$TF(t, D) = \frac{n_t}{n_d}$$

  - Where n_t is the number of times the term t appears in d, and n_d is the word number of the document d

Traditional IR

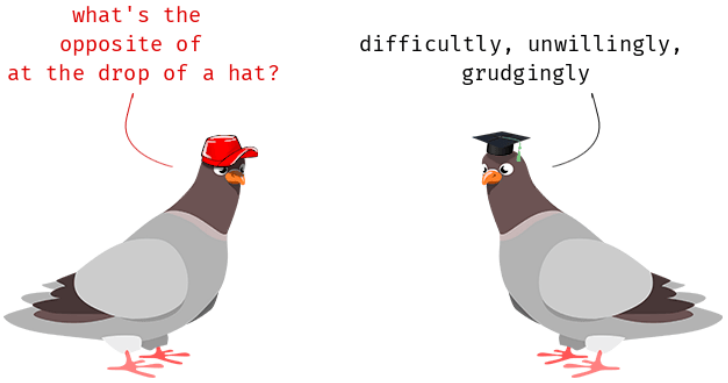
- IDF (Inverse Document Frequency)
  - The specificity of a term can be quantified as an inverse function of the number of documents in which term t appears
  - IDF is a measure to evaluate if term t is common or rare across the document collection D

$$\mathrm{IDF}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

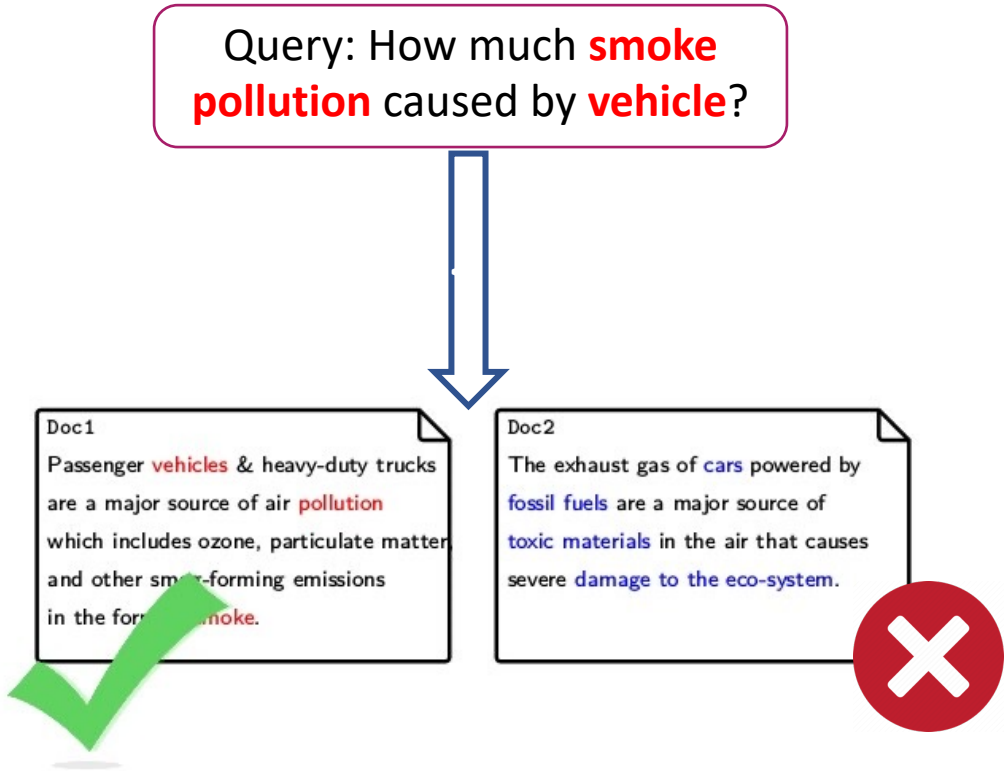
  - Where N is the total number of documents in the corpus, and $|\{d \in D : t \in d\}|$ denotes the number of documents where the term t appears

Traditional IR

- Problems
  - Vocabulary mismatch



Different vocabulary, same semantics



Traditional IR

- Problems
  - Semantic mismatch

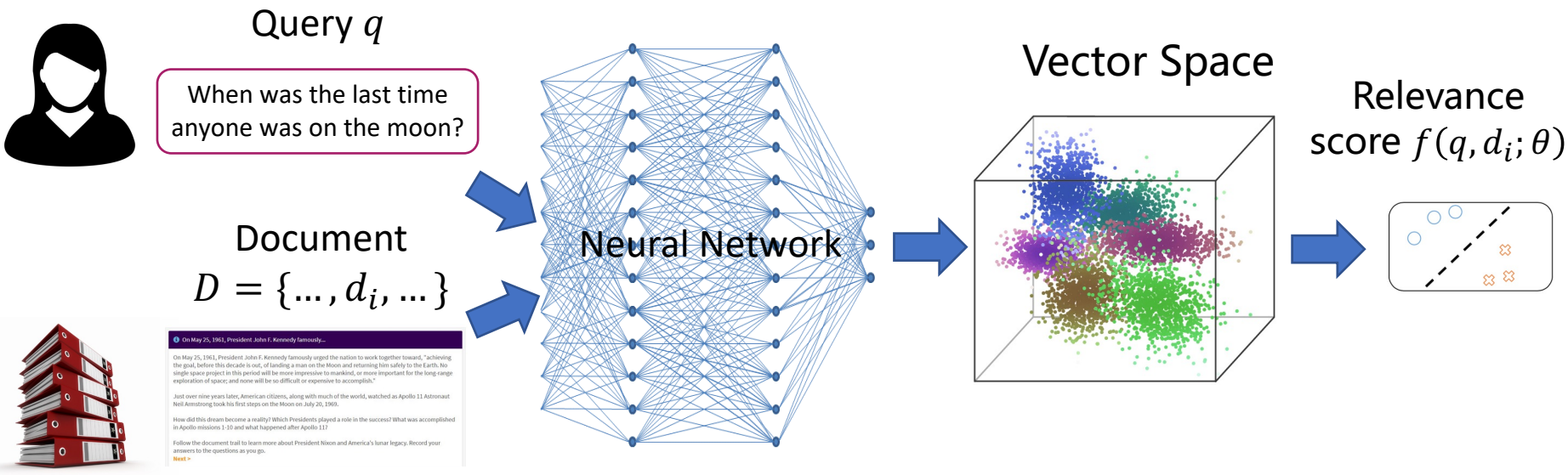
Same vocabulary, different semantics

Outline

- IR Background
- IR Formulation
- Traditional IR
- Neural IR (Big Model)
  - Cross-Encoder
  - Dual-Encoder
- Advanced Topics

Neural IR

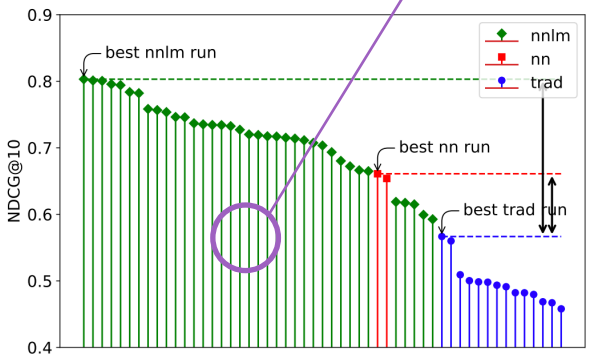
- Neural IR can mitigate traditional IR problems



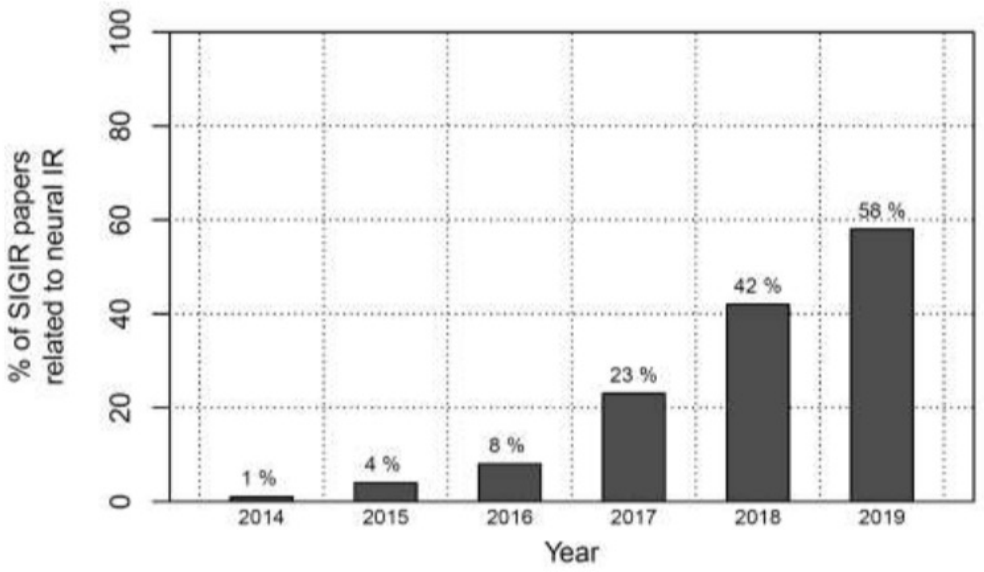
Neural IR

- Neural IR outperform traditional IR significantly
- Being neural has become a tendency for IR

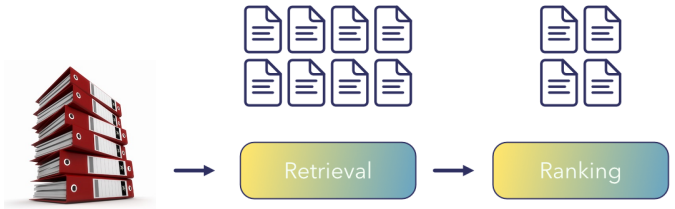
Pre-train Language model-based IR



Passage retrieval task
(TREC 2020 Deep Learning Track)



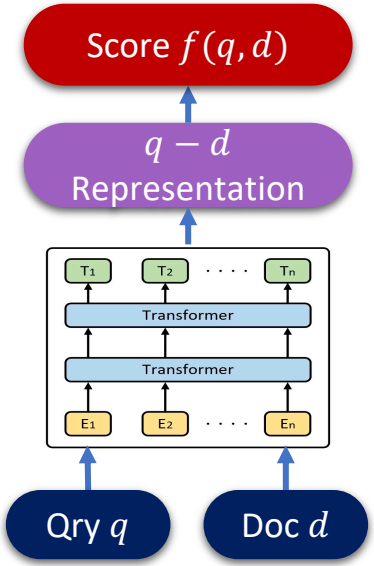
Neural IR



- Architecture

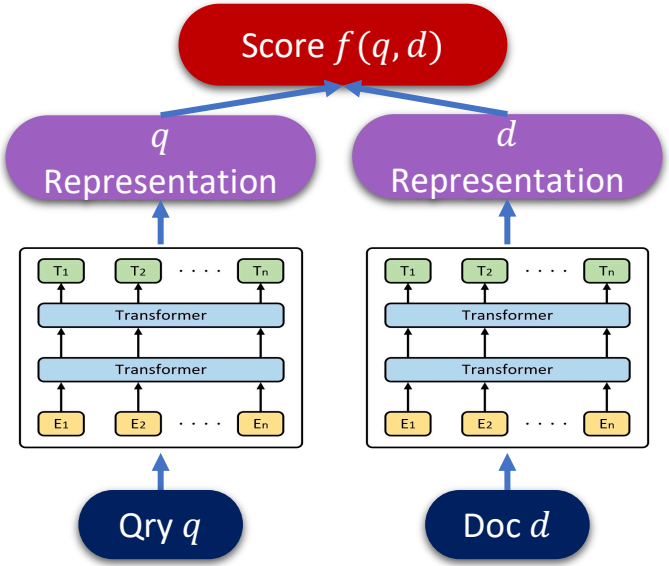
IR Pipeline	Architecture	Characteristics
Re-ranking	Cross-encoder	Model finer semantics of qry and doc; Superior performance; Higher computational cost
Retrieval	Dual-encoder	Independent representations for qry/doc; Reduce computation cost

Cross-Encoder



Big Model

Dual-Encoder



Big Models

Cross-Encoder

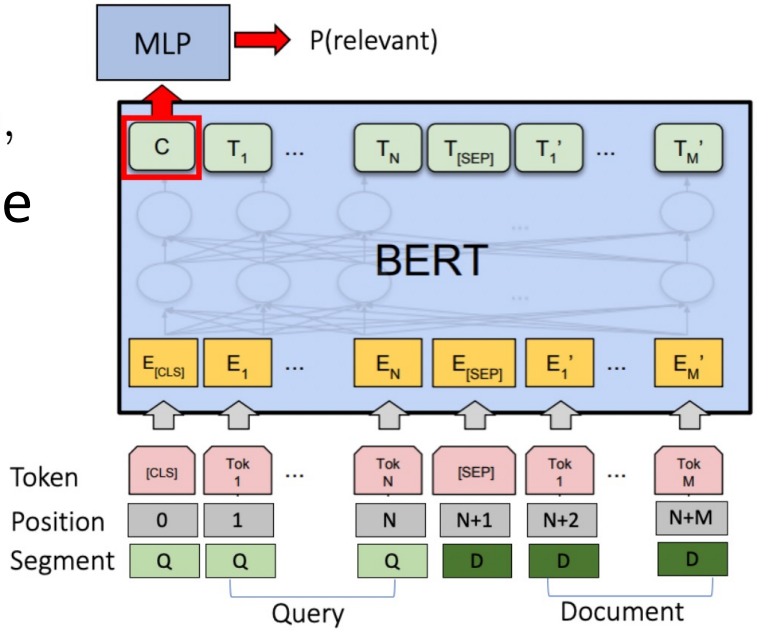
- Cross-Encoder
  - Given a query q and a document d
  - They are encoded to the token-level representations H

$$H = \text{BERT}([\text{CLS}] \circ q \circ [\text{SEP}] \circ d \circ [\text{SEP}]),$$

  - The ranking score $f(q, d; \theta)$ can be calculated as:

$$f(q, d; \theta) = \tanh(\text{Linear}(H_0)).$$

  - Training
    - Training data $\{q_i, d_i^+, d_i^-\}_{i=1}^M$
    - Training loss



Pairwise hinge loss $l_i(\theta) = max(0, 1 - f(q_i, d_i^+; \theta) + f(q_i, d_i^-; \theta))$

Cross-Encoder

- Re-ranking Performance

BERT Re-ranker

Table 2: Search accuracy on Robust04 and ClueWeb09-B. † indicates statistically significant improvements over Coor-Ascent by permutation test with p< 0.05.

	nDCG@20			
	Robust04		ClueWeb09-B	
Model	Title	Description	Title	Description
BOW	0.417	0.409	0.268	0.234
SDM	0.427	0.427	0.279	0.235
RankSVM	0.420	0.435	0.289	0.245
Coor-Ascent	0.427	0.441	**0.295**	0.251
DRMM	0.422	0.412	0.275	0.245
Conv-KNRM	0.416	0.406	0.270	0.242
BERT-FirstP	0.444†	0.491†	0.286	**0.272**†
BERT-MaxP	**0.469**†	**0.529**†	0.293	0.262†
BERT-SumP	0.467†	0.524†	0.289	0.261

T5 Re-ranker

		MS MARCO Passage	
	# Params	Dev	Test
BM25	-	0.184	0.186
+ BERT-large	340 M	0.372	0.365
+ T5-base	220 M	0.381	-
+ T5-large	770 M	0.393	-
+ T5-3B	3 B	**0.398**	0.388

Table 1: MRR@10 figures on the MS MARCO passage, with BERT-large figures from Nogueira et al. (2019a). Model sizes are also shown.

Dai, Zhuyun, et al. SIGIR 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling.
Nogueira Rodrigo, et al. EMNLP 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model.

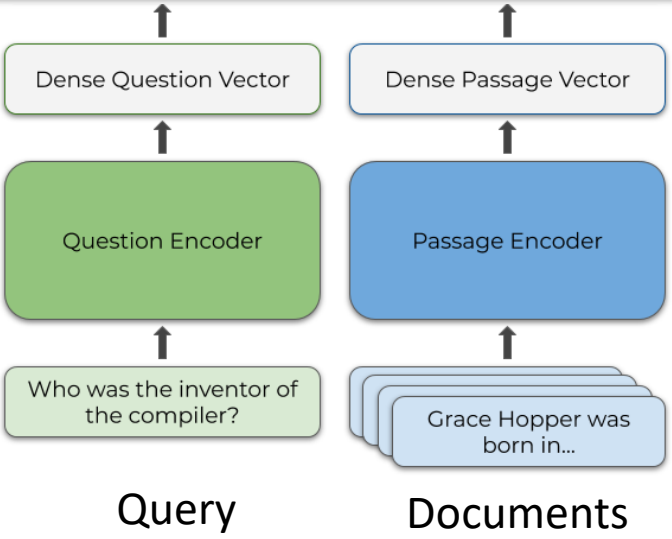
Dual-Encoder

- Dual-Encoder
  - DPR: embed query and documents using dual encoders

$$f(q, d; \theta) = g(q; \theta) \cdot g(d; \theta)$$

  - Negative log likelihood (NLL) training loss

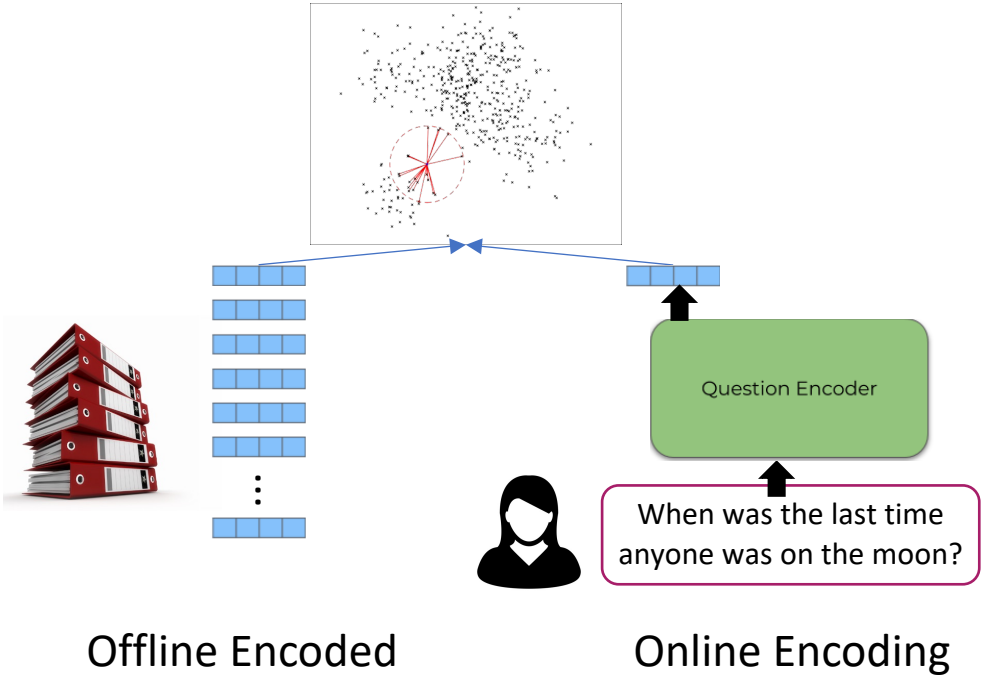
$$l(q, d^+, D^-) = -\log \frac{\exp(f(q, d^+))}{\exp(f(q, d^+)) + \sum_{d^- \in D^-} \exp(f(q, d^-))}$$



Karpukhin Vladimir, et al. EMNLP 2020. Dense Passage Retrieval for Open-Domain Question Answering.

Dual-Encoder

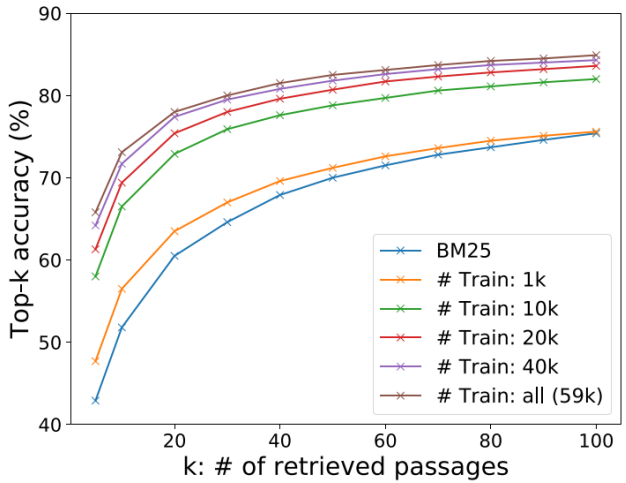
- Offline computation of doc representations
- Nearest neighbor search supported by FAISS
  - Batching & GPU can greatly improve retrieval speed (~1ms per q for 10M documents, KNN)



https://github.com/facebookresearch/faiss

Dual-Encoder

- Retrieval Performance
  - More training examples (from 1k to 59k) further improves the retrieval accuracy consistently
  - Bigger model size, better retrieval performance



	GTR-FT	GTR-PT	GTR
Fine-tuning	✓	✗	✓
NDCG@10 on MS Marco			
Base	0.400	0.258	0.420
Large	0.415	0.262	0.430
XL	0.418	0.259	0.439
XXL	0.422	0.252	0.442
Zero-shot average NDCG@10 w/o MS Marco			
Base	0.387	0.295	0.416
Large	0.412	0.315	0.445
XL	**0.433**	0.315	0.453
XXL	0.430	**0.332**	**0.458**

Table 5: Comparisons (NDCG@10) of the models trained with and without pre-training and fine-tuning. Notably, the GTR-FT XL model already achieves an average zero-shot NDCG@10 of 0.433, which outperforms the previous best dual encoder model TAS-B (NDCG@10=0.415).

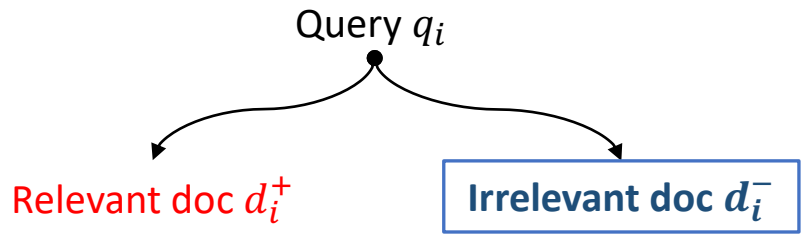
Karpukhin Vladimir, et al. EMNLP 2020. Dense Passage Retrieval for Open-Domain Question Answering.
Ni, Jianmo, et al. arXiv 2022. Large dual encoders are generalizable retrievers.

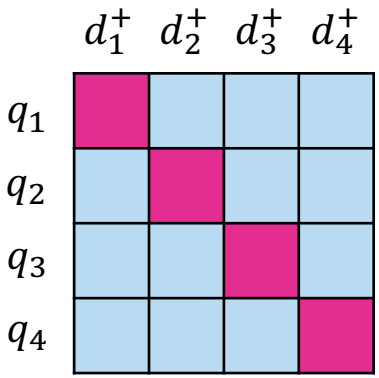
Outline

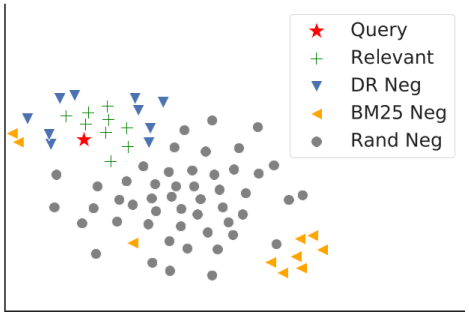
- IR Background
- IR Formulation
- Traditional IR
- Neural IR (Big Model)
- Advanced Topics
  - Negative-enhanced Fine-tuning
  - IR-oriented Pre-training
  - Few/Zero-Shot IR
  - Others

Negative-enhanced Fine-tuning

- How to mine negative?
  - In-batch negative
  - Random negative
  - BM25 negative
  - Self-retrieved hard negative (ICLR 2021)

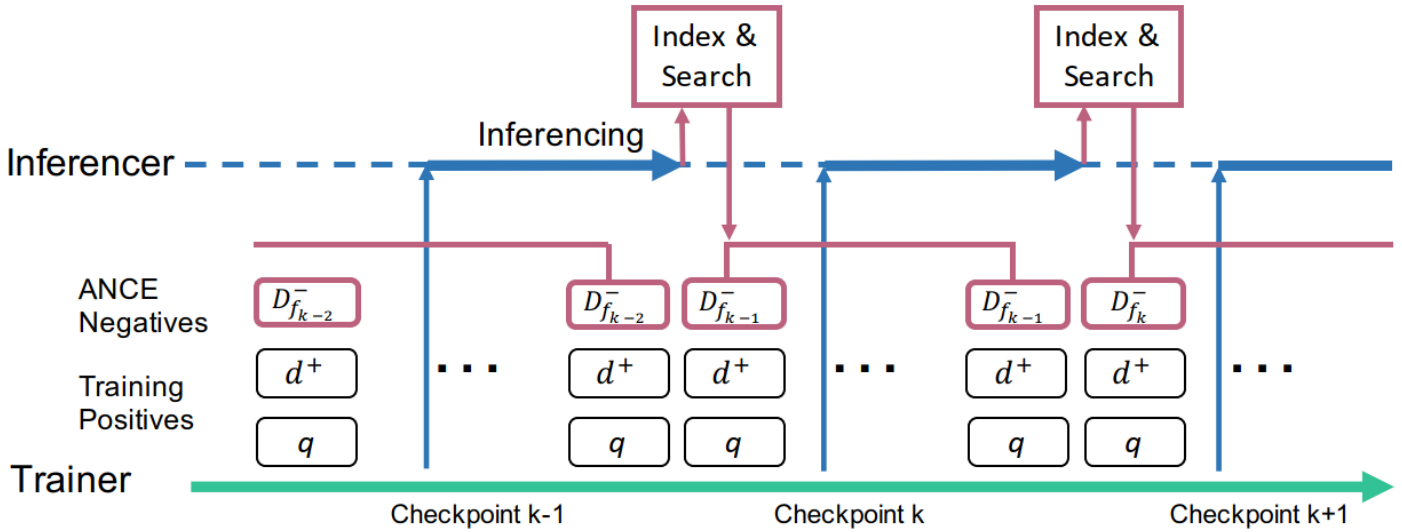






Negative-enhanced Fine-tuning

- ANCE (Approximate nearest neighbor Negative Contrastive Learning)
  - Asynchronous Index Refresh: document index goes stale after every gradient update → Refresh the index every *k* steps



Xiong et al. ICLR 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval.

Negative-enhanced Fine-tuning

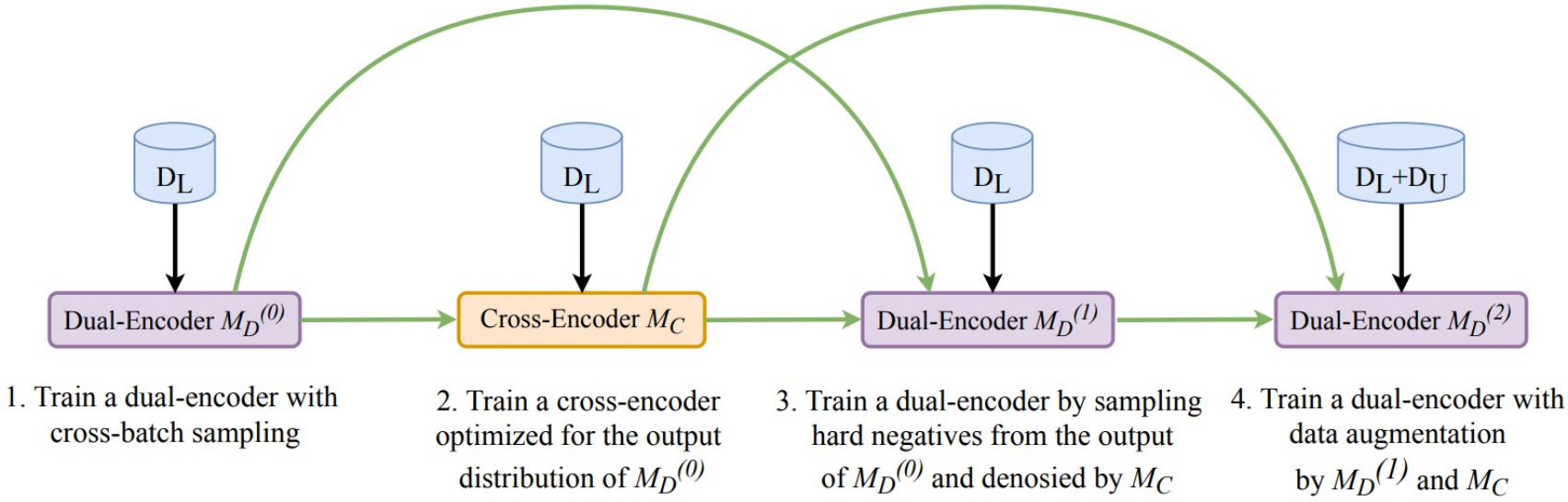
- ANCE (Approximate nearest neighbor Negative Contrastive Learning)
  - Performance Beat other dense retrieval

	MARCO Dev Passage Retrieval		TREC DL Passage NDCG@10		TREC DL Document NDCG@10	
	MRR@10	Recall@1k	Rerank	Retrieval	Rerank	Retrieval
Sparse & Cascade IR						
BM25	0.240	0.814	–	0.506	–	0.519
Best DeepCT	0.243	n.a.	–	n.a.	–	0.554
Best TREC Trad Retrieval	0.240	n.a.	–	0.554	–	0.549
BERT Reranker	–	–	**0.742**	–	0.646	–
Dense Retrieval						
Rand Neg	0.261	0.949	0.605	0.552	0.615	0.543
NCE Neg	0.256	0.943	0.602	0.539	0.618	0.542
BM25 Neg	0.299	0.928	0.664	0.591	0.626	0.529
DPR (BM25 + Rand Neg)	0.311	0.952	0.653	0.600	0.629	0.557
BM25 → Rand	0.280	0.948	0.609	0.576	0.637	0.566
BM25 → NCE Neg	0.279	0.942	0.608	0.571	0.638	0.564
BM25 → BM25 + Rand	0.306	0.939	0.648	0.591	0.626	0.540
ANCE (FirstP)	**0.330**	**0.959**	0.677	**0.648**	0.641	0.615
ANCE (MaxP)	–	–	–	–	**0.671**	**0.628**

Xiong et al. ICLR 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval.

Negative-enhanced Fine-tuning

- RocketQA (NAACL 2021)
  - Uses cross-encoder to filter hard negatives



Qu Yingqi, et al. NAACL 2021. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering.

Negative-enhanced Fine-tuning

- RocketQA (NAACL 2021)
  - Performance beats ANCE

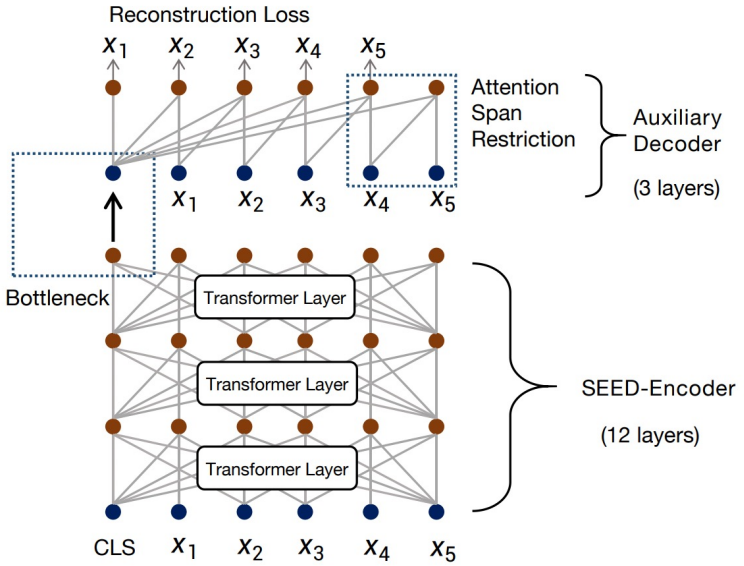
Methods	PLMs	MSMARCO Dev			Natural Questions Test		
		MRR@10	R@50	R@1000	R@5	R@20	R@100
BM25 (anserini) (Yang et al., 2017)	-	18.7	59.2	85.7	-	59.1	73.7
doc2query (Nogueira et al., 2019c)	-	21.5	64.4	89.1	-	-	-
DeepCT (Dai and Callan, 2019)	-	24.3	69.0	91.0	-	-	-
docTTTTTquery (Nogueira et al., 2019a)	-	27.7	75.6	94.7	-	-	-
GAR (Mao et al., 2020)	-	-	-	-	-	74.4	85.3
DPR (single) (Karpukhin et al., 2020)	$\text{BERT}_{\text{base}}$	-	-	-	-	78.4	85.4
ANCE (single) (Xiong et al., 2020)	$\text{RoBERTa}_{\text{base}}$	33.0	-	95.9	-	81.9	87.5
ME-BERT (Luan et al., 2020)	$\text{BERT}_{\text{large}}$	33.8	-	-	-	-	-
RocketQA	$\text{ERNIE}_{\text{base}}$	**37.0**	**85.5**	**97.9**	**74.0**	**82.7**	**88.5**

Table 2: The performance comparison on passage retrieval. Note that we directly copy the reported numbers from the original papers and leave the blanks if they were not reported.

Qu Yingqi, et al. NAACL 2021. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering.

IR-oriented Pretraining

- SEED-Encoder (EMNLP 2021)
  - pre-trains the autoencoder using a weak decoder to push the encoder to provide better text representations.
  - The encoder and decoder are connected only via [CLS]. The decoder is restricted in both param size and attention span.



$$\mathcal{L}(x, \theta_{enc}, \theta_{dec}^{weak}) = \mathcal{L}_{\text{MLM}}(x, \theta_{enc}) + \mathcal{L}_{dec}^{weak}(x, \theta_{dec}^{weak}).$$

$$\mathcal{L}_{dec}^{weak}(x, \theta_{dec}^{weak}) = -\sum_{t:1\sim n} \log P(x_t | x_{t-k:t-1}, \mathbf{h_0}; \theta_{\mathbf{dec}}^{\mathbf{weak}}),$$

$$(\text{CLS}, x_1, ..., x_n) \xrightarrow{\text{Transformer}} (\mathbf{h_0}, \mathbf{h_1}, ..., \mathbf{h_n}),$$

Lu Shuqi, et al. EMNLP 2021. Less is More: Pre-train a Strong Text Encoder for Dense Retrieval Using a Weak Decoder.

IR-oriented Pretraining

- SEED-Encoder (EMNLP 2021)
  - beats standard pretrained models

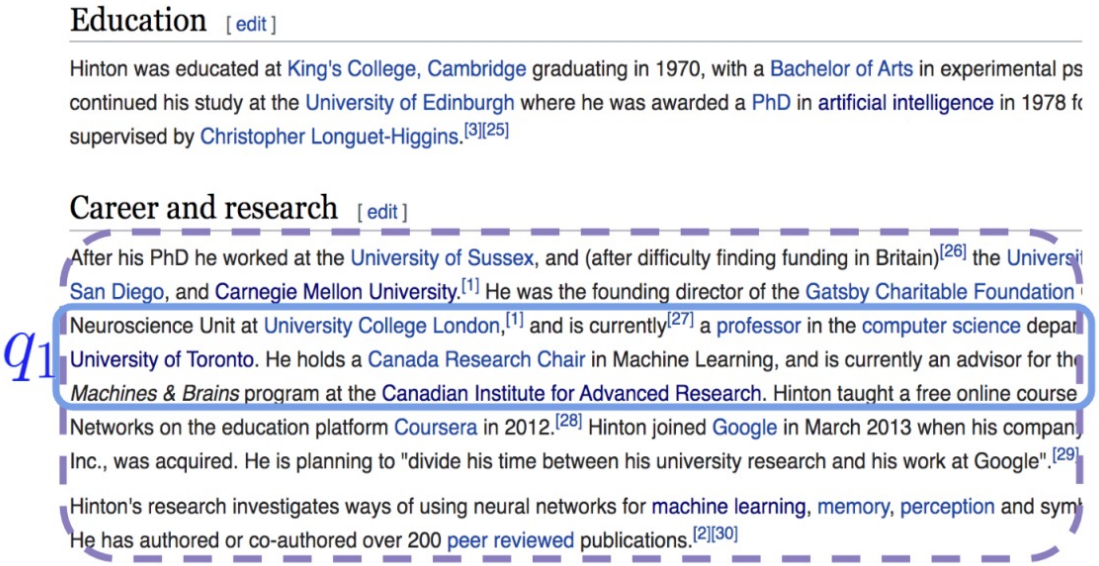
	Rerank	Retrieval	
Model	**MRR@10**	**MRR@10**	**Recall@1k**
BM25 (Craswell et al., 2020)	-	0.240	0.814
Best DeepCT (Dai and Callan, 2019)	-	0.243	n.a.
Best TREC Trad IR (Craswell et al., 2020)	-	0.240	n.a.
DPR (RoBERTa) (Karpukhin et al., 2020)	-	0.311	0.952
With DPR (BM25 Neg)			
BERT (Devlin et al., 2018)	0.317	0.310	0.929
Optimus (Li et al., 2020)	0.300	0.244	0.880
ELECTRA (Clark et al., 2020)	0.300	0.258	0.854
ERNIE2.0 (Sun et al., 2020)	0.324	0.321	0.942
RoBERTa (Liu et al., 2019)	-	0.299	0.928
BERT (Ours)	0.326	0.320	0.933
SEED-Encoder	**0.329**†	**0.329**†	**0.953**†
With ANCE (FirstP)			
RoBERTa (Liu et al., 2019)	-	0.330	0.959
BERT (Ours)	0.327	0.332	0.952
SEED-Encoder	**0.334**†	**0.339**†	**0.961**†

Table 1: First stage retrieval results on MS MARCO Passage ranking Dev set. Rerank MRR is for reference only. Statistically significant improvements over BERT (Ours) are marked by †.

Lu Shuqi, et al. EMNLP 2021. Less is More: Pre-train a Strong Text Encoder for Dense Retrieval Using a Weak Decoder.

IR-oriented Pretraining

- ICT (Inverse Cloze Task)
  - Given a passage consisting of n sentences
  - the query is a sentence randomly drawn from the passage, and the document is the rest of sentences



Education [edit]

Hinton was educated at King's College, Cambridge graduating in 1970, with a Bachelor of Arts in experimental ps
continued his study at the University of Edinburgh where he was awarded a PhD in artificial intelligence in 1978 f
supervised by Christopher Longuet-Higgins.[3][25]

Career and research [edit]

After his PhD he worked at the University of Sussex, and (after difficulty finding funding in Britain)[26] the Universit
San Diego, and Carnegie Mellon University.[1] He was the founding director of the Gatsby Charitable Foundation
Neuroscience Unit at University College London,[1] and is currently[27] a professor in the computer science depar
University of Toronto. He holds a Canada Research Chair in Machine Learning, and is currently an advisor for th
Machines & Brains program at the Canadian Institute for Advanced Research. Hinton taught a free online course
Networks on the education platform Coursera in 2012.[28] Hinton joined Google in March 2013 when his compan
Inc., was acquired. He is planning to "divide his time between his university research and his work at Google".[29]

Hinton's research investigates ways of using neural networks for machine learning, memory, perception and sym
He has authored or co-authored over 200 peer reviewed publications.[2][30]

q_1

$$p_\theta(\boldsymbol{d}|\boldsymbol{q}) = \frac{\exp\left(f_\theta(\boldsymbol{q},\boldsymbol{d})\right)}{\sum_{\boldsymbol{d}'\in\mathcal{D}}\exp\left(f_\theta(\boldsymbol{q},\boldsymbol{d}')\right)}$$

In-batch negative

Chang Wei-Cheng, et al. ICLR 2021. Pre-training Tasks for Embedding-based Large-scale Retrieval.

IR-oriented Pretraining

- ICT (Inverse Cloze Task)
  - ICT pre-training improves retrieval performance

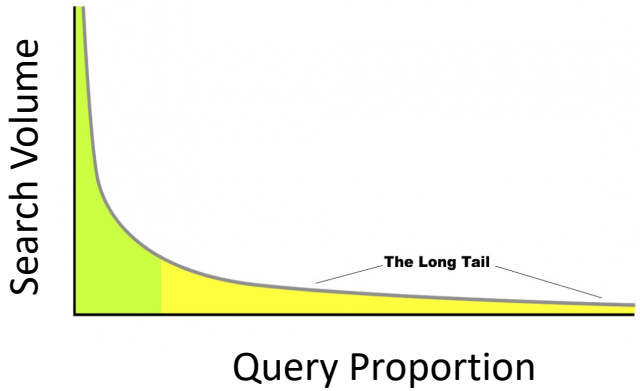
	R@1	R@5	R@20	R@100
DPR (BERT-Base)	-	-	78.4	85.3
ANCE (BERT-Base)	-	-	81.9	87.5
RocketQA (ERNIE-2.0-Base)	-	74.0	82.7	88.5
PAIR (ERNIE-2.0-Base)	-	74.9	83.5	89.1
AR2-G (ERNIE-2.0-Base)	57.2	76.6	85.3	89.8
AR2-G (BERT-2.0-Base)	56.7	76.1	85.0	89.3
AR2-G (ERNIE-Base w/ ICT)	58.7	77.9	86.0	90.1

Natural Question Results

Zhang Hang, et al. ICLR 2021. Adversarial Retriever-Ranker for Dense Text Retrieval.

Few-Shot IR

- Many real-world scenarios are **"few-shot"** where large supervision is hard to obtain



Long-Tail Web Search

Personal/Enterprise Search

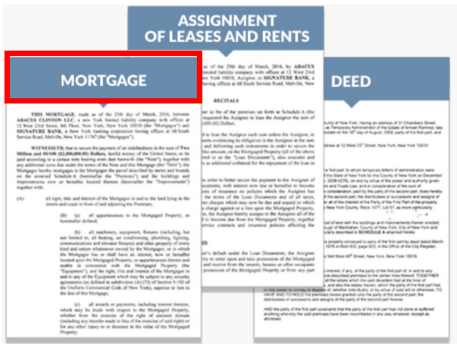
Biomedical/Legal Search

Few-Shot IR

- Weak supervision generation

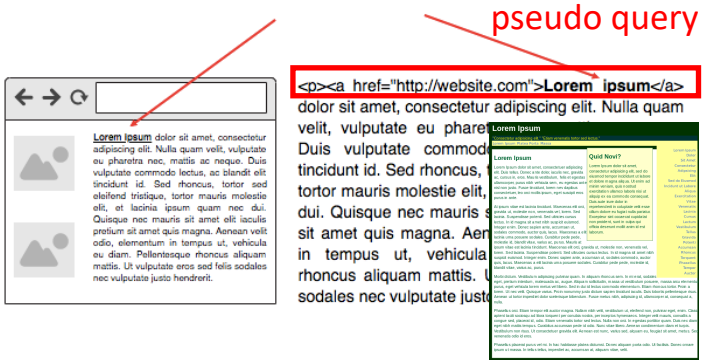
Title

Pseudo query

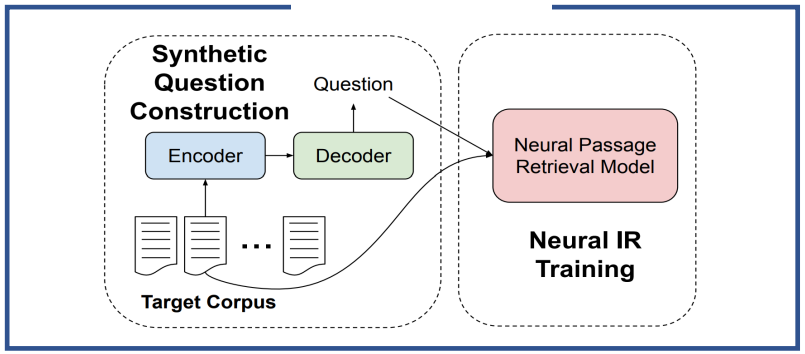


Anchor

pseudo query



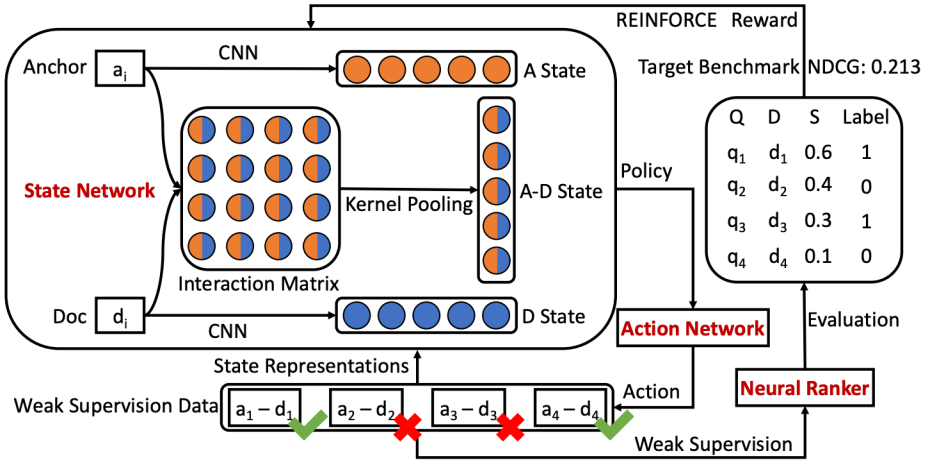
Generation

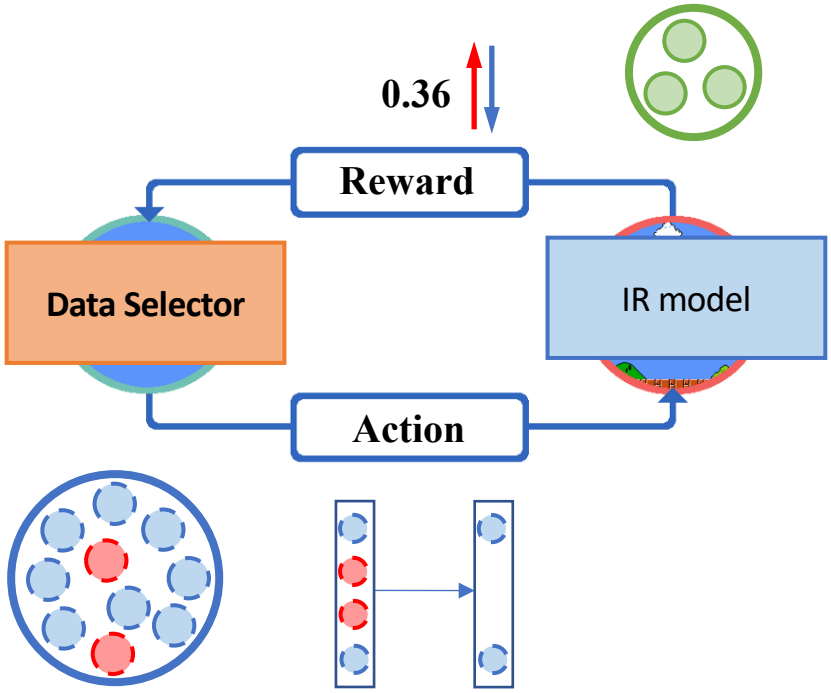


Sean MacAvaney, et al. SIGIR 2019. Content-based weak supervision for ad-hoc re-ranking.
Kaitao Zhang, et al. WWW 2020. Selective weak supervision for neural information retrieval.
Ji Ma, et al. EACL 2021. Zero-shot neural passage retrieval via domain-targeted synthetic question generation.

Few-Shot IR

- Weak supervision selection
  - Reinforcement data selection (ReinfoSelect)
  - Learn to select training pairs that best weakly supervise the neural ranker

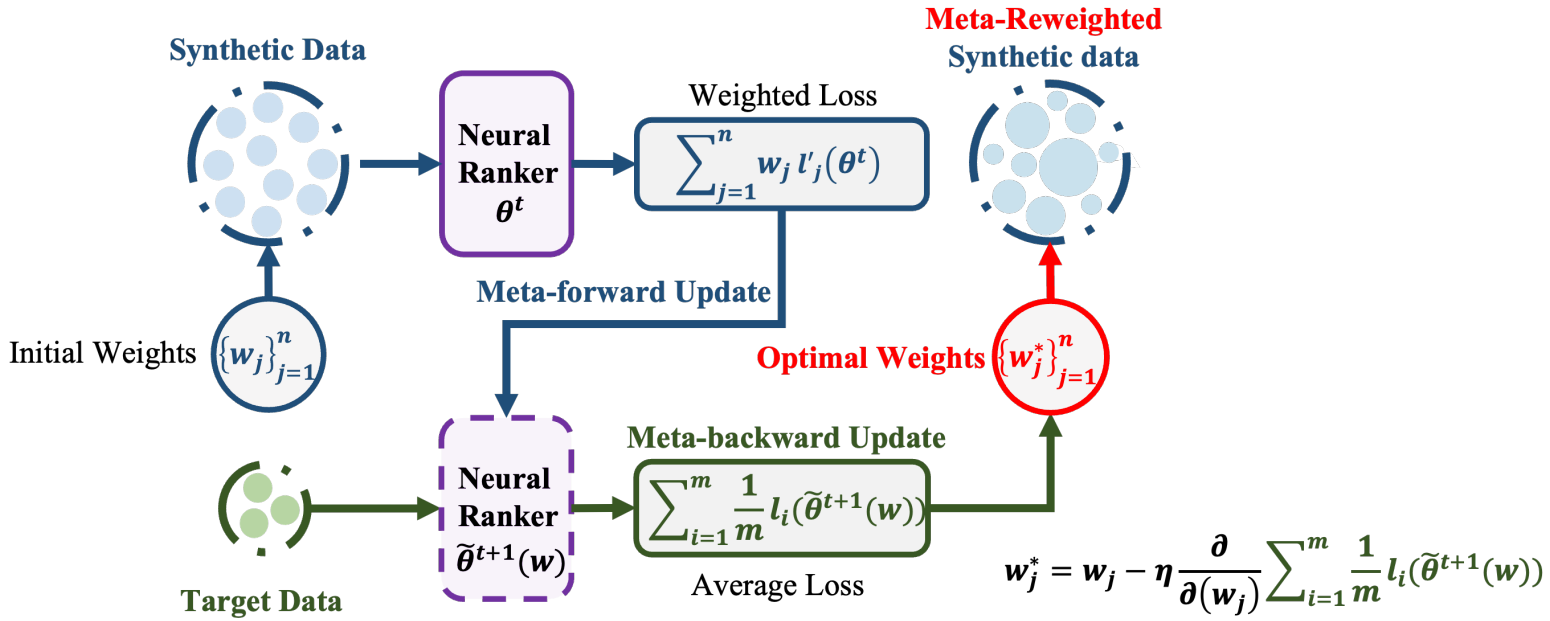




Kaitao Zhang, et al. WWW 2020. Selective weak supervision for neural information retrieval.

Few-Shot IR

- Weak supervision selection
  - Meta-learning data selection (MetaAdaptRank)
  - Learn to reweight training pairs that best weakly supervise the neural ranker



Si Sun, et al. ACL 2021. Few-Shot Text Ranking with Meta Adapted Synthetic Weak Supervision.

Few-Shot IR

- Weak supervision selection
  - MetaAdaptRank beats ReinfoSelect

	Methods (Supervision Sources)	ClueWeb09-B (Web) NDCG@20	ClueWeb09-B (Web) ERR@20	Robust04 (News) NDCG@20	Robust04 (News) ERR@20	TREC-COVID (BioMed) NDCG@20	TREC-COVID (BioMed) P@20
(a)	ReInfoSelect (MS MARCO)	0.3294	0.1760	0.4756	0.1291	$0.8229^{\ddagger}$	$0.8780^{\ddagger}$
(b)	ReInfoSelect (Anchor)	0.3261	0.1669	0.4703	0.1313	0.7891	0.8430
(c)	ReInfoSelect (CTSyncSup)	0.3243	0.1742	$0.4816^{\ddagger}$	0.1334	$0.8230^{\ddagger}$	$0.8800^{\ddagger}$
(d)	MetaAdaptRank (MS MARCO)	$0.3453^{\dagger\ddagger\flat}$	$\mathbf{0.2018}^{\dagger\ddagger\flat\sharp}$	$0.4853^{\ddagger}$	0.1331	$0.8354^{\ddagger\sharp}$	$0.8730^{\ddagger}$
(e)	MetaAdaptRank (Anchor)	0.3374	0.1730	0.4797	0.1314	0.8045	0.8650
(f)	MetaAdaptRank (CTSyncSup)	$0.3416^{\flat}$	$0.1893^{\ddagger\sharp}$	$0.4916^{\dagger\ddagger\sharp}$	$0.1362^{\dagger\sharp}$	$0.8378^{\ddagger\sharp}$	$0.8790^{\ddagger}$
(g)	MetaAdaptRank (MARCO + CTSyncSup)	$\mathbf{0.3498}^{\dagger\ddagger\sharp}$	$0.1926^{\ddagger\flat\sharp}$	$\mathbf{0.4989}^{\dagger\ddagger\flat\natural\sharp}$	$\mathbf{0.1366}^{\dagger\natural}$	$\mathbf{0.8488}^{\dagger\ddagger\flat\natural\sharp}$	$\mathbf{0.8910}^{\ddagger\natural\sharp}$

Table 5: Ranking accuracy of ReInfoSelect and MetaAdaptRank using different supervision sources. Superscripts $\dagger, \ddagger, \flat, \natural, \sharp, \S$ indicate statistically significant improvements over $(a)^{\dagger}$, $(b)^{\ddagger}$, $(c)^{\flat}$, $(d)^{\natural}$, $(e)^{\sharp}$ and $(f)^{\S}$.

Si Sun, et al. ACL 2021. Few-Shot Text Ranking with Meta Adapted Synthetic Weak Supervision.

Zero-Shot IR

- Generalizable T5-based dense Retrievers (GTR)

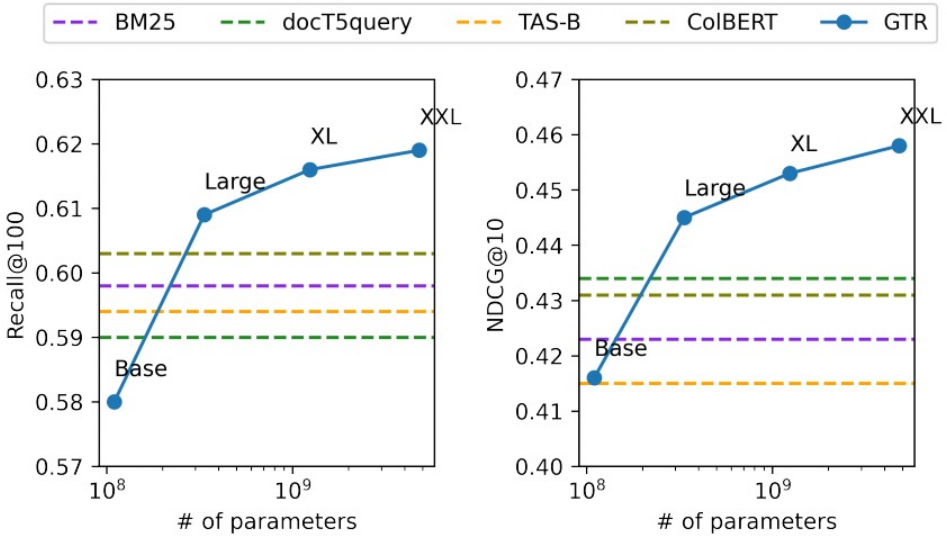
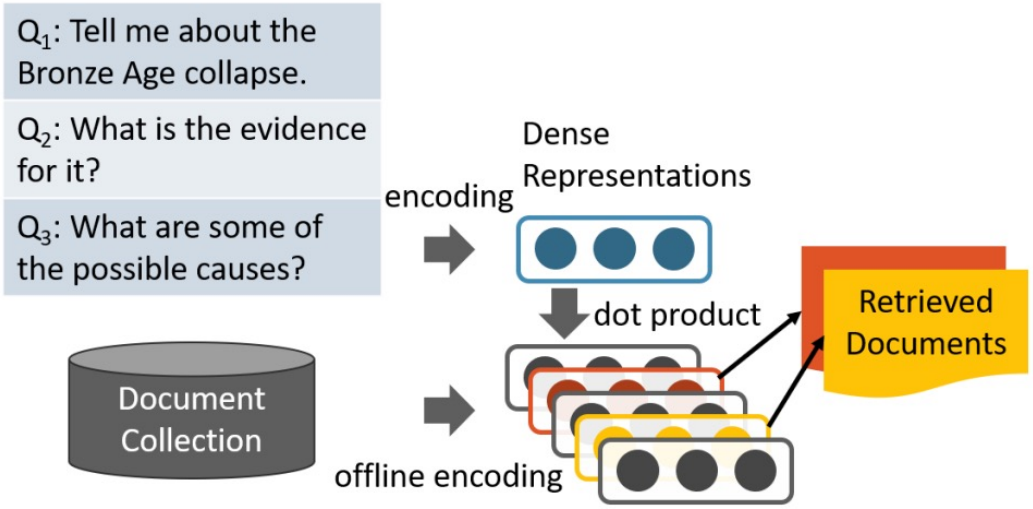


Figure 1: The average Recall@100 and NDCG@100 on all BEIR tasks excluding MS Marco. Scaling up consistently improves dual encoders' out-of-domain performance.

Ni Jianmo, et al. arXiv 2022. Large dual encoders are generalizable retrievers.

Other Topics

- Conversational IR
  - Models multiple rounds of query
- How to use big model to retrieve long documents?
  - Long-range dependency



Shi Yu, et al. SIGIR 2021. Few-Shot Conversational Dense Retrieval.

Demo

- Load Document Representations

```
## ****************************************
## (1) Load Document Representations
## ****************************************

index_files = glob.glob(args.passage_reps)
logger.info(f'Pattern match found {len(index_files)} files; loading them into index.')

p_reps_0, p_lookup_0 = pickle_load(index_files[0])
retriever = BaseFaissIPRetriever(p_reps_0, args.use_gpu)

shards = chain([(p_reps_0, p_lookup_0)], map(pickle_load, index_files[1:]))
if len(index_files) > 1:
    shards = tqdm(shards, desc='Loading shards into index', total=len(index_files))

assert len(index_files) == args.index_num

p_reps = []
look_up = []
for _p_reps, p_lookup in shards:
    p_reps.append(_p_reps)
    look_up += p_lookup

p_reps = np.concatenate(p_reps, axis=0)
retriever.add(p_reps)

## ****************************************
## (2) Visulize Doc Representations
## ****************************************
print("Doc Number & Dimenson: ", p_reps.shape)
print(p_reps)
```

```
Doc Number & Dimenson:  (8841823, 768)
[[-0.38321945  0.25906575  0.26489395 ...  0.15825956 -0.17769206
  -0.01632608]
 [-0.00561821  0.18144645  0.1604223  ... -0.12024047 -0.12155294
   0.1989834 ]
 [-0.54339373  0.1314484  -0.15276673 ... -0.05960921 -0.46373585
  -0.08406078]
 ...
 [-0.2320291   0.20622829 -0.04704667 ...  0.39412552  0.11358216
  -0.24110852]
 [-0.692405    0.21709926 -0.04632384 ...  0.7187064   0.08598054
   0.05040906]
 [ 0.23211516 -0.06568207 -0.06819141 ...  0.00336331  0.02062089
  -0.21549858]]
```

Demo

- Load Query Representations

```
    ## ****************************************
    ## (3) Visulize Query Representations
    ## ****************************************

    q_reps, q_lookup = pickle_load(args.query_reps)
    q_reps = q_reps

    print("Query Number & Dimenson: ", q_reps.shape)
    print(q_reps)
```

```
Query Number & Dimenson:  (6980, 768)
[[-0.21370466  0.12421735  0.31339    ...  0.18244497  0.51787394
  -0.2646188 ]
 [-0.2288768   0.13674128  0.01598361 ...  0.03635952  0.09578158
   0.12798253]
 [-0.11634977  0.16813089  0.13740303 ... -0.12385005 -0.08023782
   0.033319  ]
 ...
 [-0.05861889  0.1031251  -0.0871998  ... -0.23143262  0.0790739
  -0.06239492]
 [ 0.16780636  0.1445717   0.22115536 ...  0.09035147 -0.20445555
   0.01372622]
 [-0.23251729  0.02351602  0.22331919 ...  0.16926521  0.22054258
  -0.02174541]]
```

Demo

- Batch Search

```
## ********************************************
## (4) Batch Search
## ********************************************
all_scores, psg_indices = search_queries(retriever, q_reps, look_up, args)
```

- Visualize retrieved results

```
## ********************************************
## (5) Visulize Examples
## ********************************************
qid_index = 0
qid = q_lookup[qid_index]
query = queries[qid]
print("Queston: ", query)
```

```
Queston:  what is paula deen's brother
```

```
for rank, (score, docid) in enumerate(zip(all_scores[0], psg_indices[0])):
    print("Rank: ", rank+1)
    print("Score: ", score)
    print("Document: ", documents[int(docid)]["text"])
    print(" ")
```

```
Rank:  1
Score:  154.83257
Document:  Brian Killian/WireImage. Paula Deen and her brother Earl W. â€ Bubbaâ€  Hier
s are being sued by a former general manager at Uncle Bubbaâ€ s Seafood and Oyster Hous
e, a restaurant they co-own.

Rank:  2
Score:  154.45937
Document:  Paula Deen & Brother Bubba Sued for Harassment. Paula Deen and her brother Ea
rl W. â€ Bubbaâ€  Hiers are being sued by a former general manager at Uncle Bubbaâ€ s S
eafood and Oyster House, a restaurant they co-own. Lisa Jackson â€ who worked as genera
l manager at the Savannah, Ga., eatery for five years â€ alleges in court documents fil
ed Monday in the Superior Court of Chatham County, that she was subjected to sexual hara
ssment and violent behavior by Hiers.

Rank:  3
Score:  154.40625
Document:  The New York Times. U.S. | National Briefing | South. Georgia: Paula Deen and
Brother Shut a Restaurant. The celebrity chef Paula Deen and her younger brother, Bubba
Hiers, have closed a Savannah seafood restaurant that served as the backdrop to a workpl
ace discrimination lawsuit that stained her reputation.
```

Thank You!

Si Sun

s-sun17@mails.tsinghua.edu.cn

THUNLP

Question Answering

Ganqu Cui

cgq19@mails.tsinghua.edu.cn

THUNLP

Outline

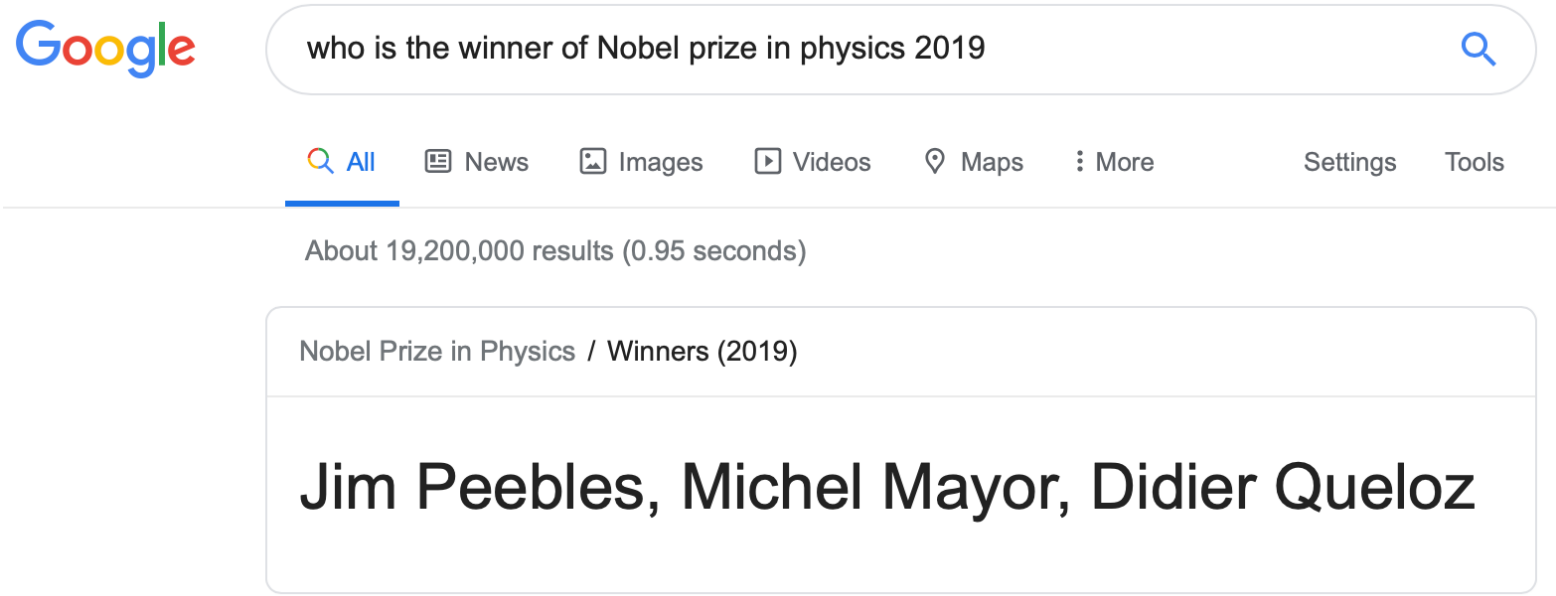
- Introduction to QA
- Reading Comprehension
- Open-domain QA

Background

- Why do we need question answering (QA) ?
  - When we search for something in Google, it's usually hard to find answers from the document list
  - With QA systems, answers are automatically found from large amount of data

Applications of QA

- Better search experience



Applications of QA

- IBM Watson: 2011 Winner in Jeopardy
- Defeat two human players (Ken and Brad)

Applications of QA

- Intelligent assistants

Apple Siri (2011)

Microsoft Cortana (2014)

Amazon Alexa (2014)

Google Home (2016)

History

Template-based QA
Expert System

IR-based QA

Community QA

Machine Reading
Comprehension
KBQA

1960 1990 2000 2010

BASEBALL
LUNAR
MACSYMA
SHRDLE

MASQUE
TREC

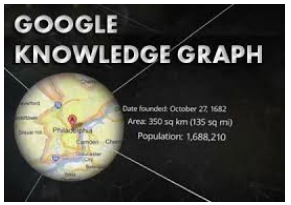
START
Natural Language Question Answering System

YAHOO! ANSWERS
BETA

ask. answer. discover.

知乎
与世界分享你的知识、经验和见解

WolframAlpha
computational knowledge engine



ProBase

Types of QA

- Machine Reading Comprehension:
  - Read specific documents and answer questions
- Open-domain QA:
  - Search and read relevant documents to answer questions
- Knowledge-based QA:
  - Answer questions based on knowledge graph
- Conversational QA and dialog:
  - Answer questions according to dialog history
- …

Outline

- Introduction to QA
- Reading Comprehension
  - Task Definition and Dataset
  - Traditional Pipeline
  - Big-model-based Methods
- Open-domain QA

Definition of RC

Documents →

One night I was at my friend's house where he threw a party. We were enjoying our dinner at night when all of a sudden, we heard a knock on the door. I opened the door and saw this guy who had scar on his face. As soon as I saw him, I ran inside the house and called the cops. The cops came and the guy ran away as soon as he heard the cop car coming. We never found out what happened to that guy after that day.

Questions →

1. What was the strange guy doing with the friend?

Candidate answers →

A) enjoying a meal
B) talking about his job
C) talking to him
D) trying to beat him

2. Why did the strange guy run away?

A) because he heard the cop car
B) because he saw his friend
C) because he didn't like the dinner
D) because it was getting late

Types of RC

- Cloze test
  - CNN/Daily Mail (93k CNN articles, 220k Daily Mail articles)

Original Version	Anonymised Version
Context	
The BBC producer allegedly struck by Jeremy Clarkson will not press charges against the "Top Gear" host, his lawyer said Friday. Clarkson, who hosted one of the most-watched television shows in the world, was dropped by the BBC Wednesday after an internal investigation by the British broadcaster found he had subjected producer Oisin Tymon "to an unprovoked physical and verbal attack." ...	the *ent381* producer allegedly struck by *ent212* will not press charges against the " *ent153* " host , his lawyer said friday . *ent212* , who hosted one of the most - watched television shows in the world , was dropped by the *ent381* wednesday after an internal investigation by the *ent180* broadcaster found he had subjected producer *ent193* " to an unprovoked physical and verbal attack . " ...
Query	
Producer **X** will not press charges against Jeremy Clarkson, his lawyer says.	producer **X** will not press charges against *ent212* , his lawyer says .
Answer	
Oisin Tymon	*ent193*

Types of RC

- Cloze test
  - CBT (Children's Book Test)
  - Context: 20 continuous sentences
  - Question: the 21st sentence, with an entity masked
  - Answer: the masked entity
  - 10 candidates

S: 1 Mr. Cropper was opposed to our hiring you .
2 Not , of course , that he had any personal objection to you , but he is set against female teachers , and when a Cropper is set there is nothing on earth can change him .
3 He says female teachers ca n't keep order .
4 He 's started in with a spite at you on general principles , and the boys know it .
5 They know he 'll back them up in secret , no matter what they do , just to prove his opinions .
6 Cropper is sly and slippery , and it is hard to corner him . ''
7 `` Are the boys big ? ''
8 queried Esther anxiously .
9 `` Yes .
10 Thirteen and fourteen and big for their age .
11 You ca n't whip 'em -- that is the trouble .
12 A man might , but they 'd twist you around their fingers .
13 You 'll have your hands full , I 'm afraid .
14 But maybe they 'll behave all right after all . ''
15 Mr. Baxter privately had no hope that they would , but Esther hoped for the best.
16 She could not believe that Mr. Cropper would carry his prejudices into a personal application .
17 This conviction was strengthened when he overtook her walking from school the next day and drove her home .
18 He was a big , handsome man with a very suave , polite manner .
19 He asked interestedly about her school and her work , hoped she was getting on well , and said he had two young rascals of his own to send soon .
20 Esther felt relieved .

q: She thought that Mr. _____ had exaggerated matters a little .

C: Baxter, Cropper, Esther, course, fingers, manner, objection, opinion, right, spite.

a: Baxter

Types of RC

- Multiple choice
  - RACE: 100k multiple choice questions collected from English exams in China.

Passage:
In a small village in England about 150 years ago, a mail coach was standing on the street. It didn't come to that village often. People had to pay a lot to get a letter. The person who sent the letter didn't have to pay the postage, while the receiver had to.
"Here's a letter for Miss Alice Brown," said the mailman.
" I'm Alice Brown," a girl of about 18 said in a low voice.
Alice looked at the envelope for a minute, and then handed it back to the mailman.
"I'm sorry I can't take it, I don't have enough money to pay it", she said.
A gentleman standing around were very sorry for her. Then he came up and paid the postage for her.
When the gentleman gave the letter to her, she said with a smile, " Thank you very much, This letter is from Tom. I'm going to marry him. He went to London to look for work. I've waited a long time for this letter, but now I don't need it, there is nothing in it."
"Really? How do you know that?" the gentleman said in surprise.
"He told me that he would put some signs on the envelope. Look, sir, this cross in the corner means that he is well and this circle means he has found work. That's good news."
The gentleman was Sir Rowland Hill. He didn't forgot Alice and her letter.
"The postage to be paid by the receiver has to be changed," he said to himself and had a good plan.
"The postage has to be much lower, what about a penny? And the person who sends the letter pays the postage. He has to buy a stamp and put it on the envelope." he said . The government accepted his plan. Then the first stamp was put out in 1840. It was called the "Penny Black". It had a picture of the Queen on it.

Questions:

1): The first postage stamp was made _.
A. in England B. in America C. by Alice D. in 1910

2): The girl handed the letter back to the mailman because _ .
A. she didn't know whose letter it was
B. she had no money to pay the postage
C. she received the letter but she didn't want to open it
D. she had already known what was written in the letter

3): We can know from Alice's words that _ .
A. Tom had told her what the signs meant before leaving
B. Alice was clever and could guess the meaning of the signs
C. Alice had put the signs on the envelope herself
D. Tom had put the signs as Alice had told him to

4): The idea of using stamps was thought of by _ .
A. the government
B. Sir Rowland Hill
C. Alice Brown
D. Tom

5): From the passage we know the high postage made _ .
A. people never send each other letters
B. lovers almost lose every touch with each other
C. people try their best to avoid paying it
D. receivers refuse to pay the coming letters

Answer: ADABC

Types of RC

- Extractive RC: Predict a span in documents
  - SQuAD: 10k human-annotated questions and 536 articles from Wikipedia. Every answer is a span in the article

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under gravity. The main forms of precipitation include drizzle, rain, sleet, snow, graupel and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals within a cloud. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall? Answer: gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail? Answer: graupel

Where do water droplets collide with ice crystals to form precipitation? Answer: within a cloud

Datasets

Multiple choice
Cloze test
Extractive

2015 bAbI CNN/Daily Mail

2016 SCT LAMBDA MSMARCO NEWSQA SQuAD CBT

2017 Who-Did-What SearchQA NarrativeQA DuReader RACE Quasar TriviaQA

2018 CoQA HOTPOTQA SQuAD2.0 CliCR DuoRC CLOTH MCScript ARC emrQA RecipeQA openBookQA ShARC QuAC ProPara TextWordsQA

2019 DREAM DROP Natural Questions RC-QED GeosQA CosmosQA QASC

2020 XCOPA COVID-QA LiveQA PIAF GrailQA

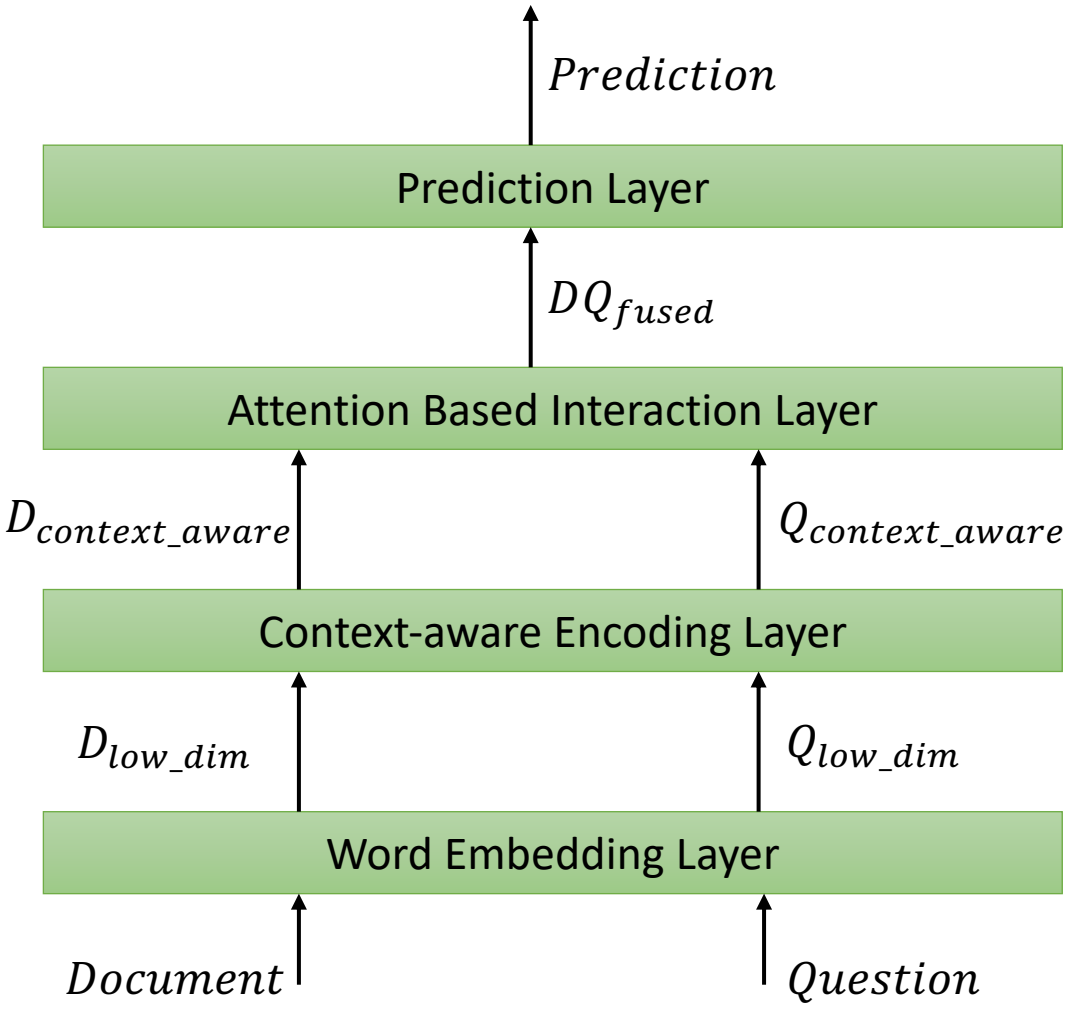
2021 CodeQA CCQA THQuAD MeDiaQA

2022 MedMCQA FeTaQA PQuAD JaQuAD

Outline

- Introduction to QA
- Reading Comprehension
  - Task Definition and Dataset
  - Traditional Pipeline
  - Big-model-based Methods
- Open-domain QA

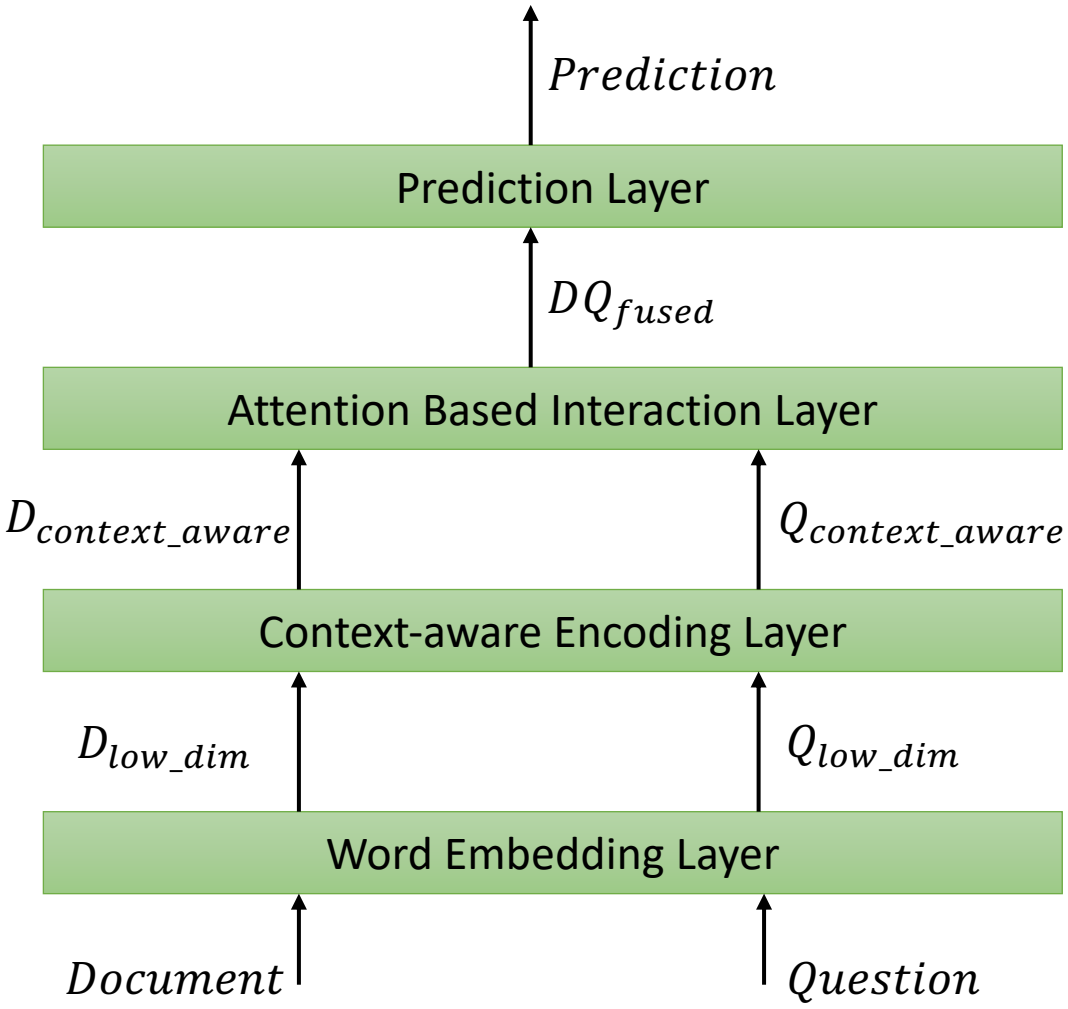
Model Framework



General framework in RC:
embed, encode, interact, and predict

Seo et al., Bidirectional Attention Flow for Machine Comprehension, In Proceedings of ICLR 2017

Model Framework



Bilinear, Pointer Network

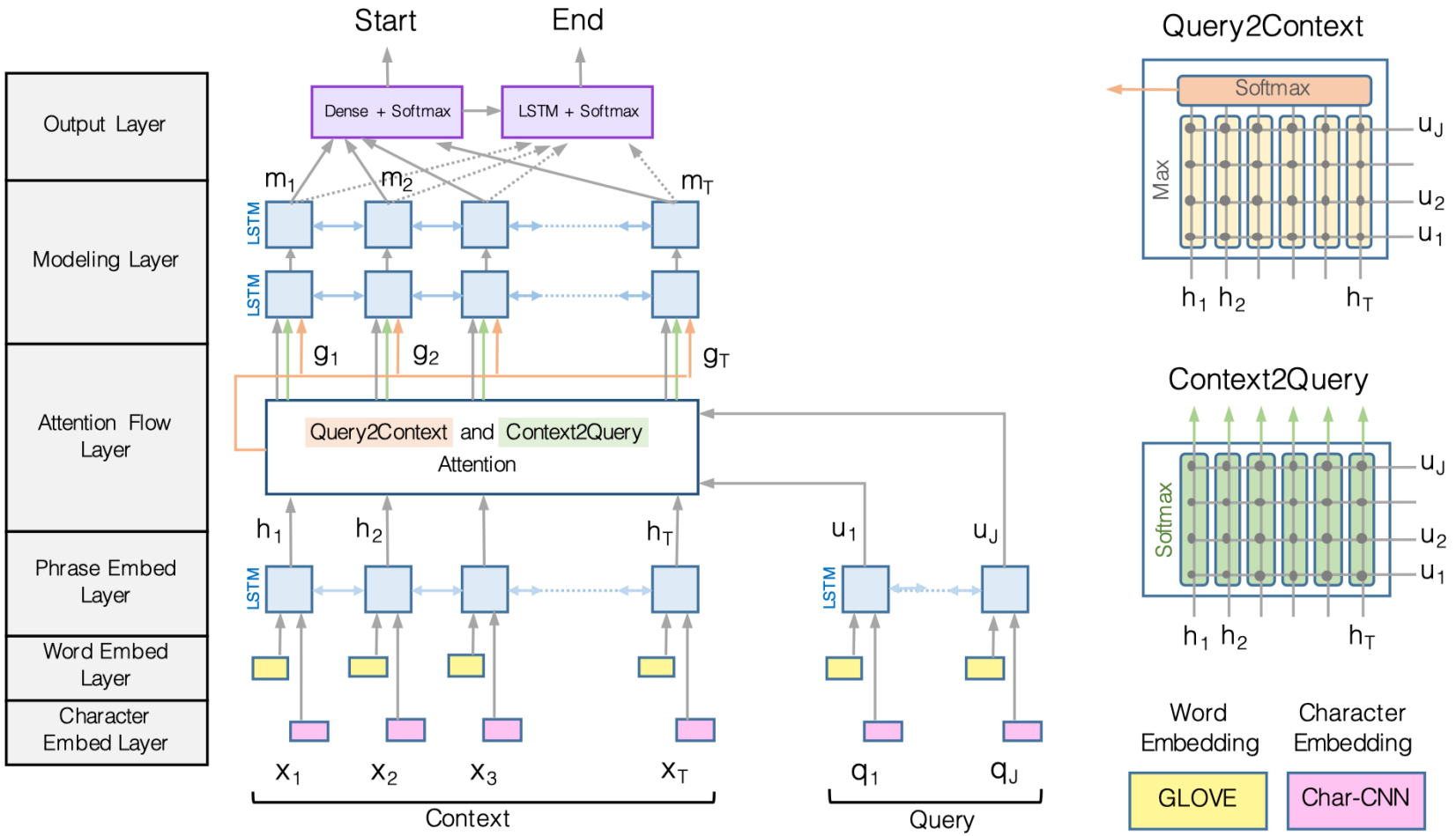
Attention: d2q, q2d

LSTM, GRU, Attention

GloVe, ELMo, Char Embedding

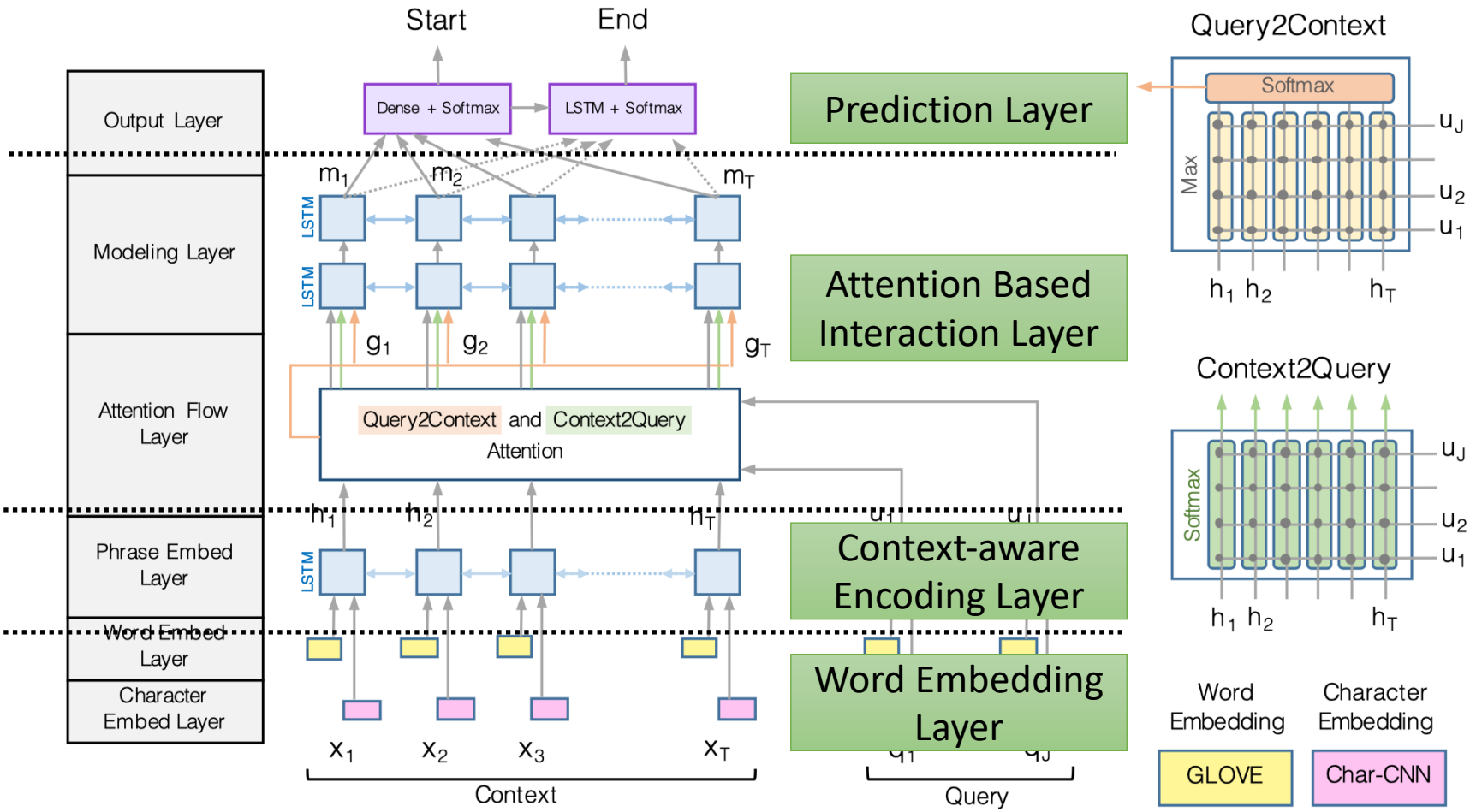
Seo et al., Bidirectional Attention Flow for Machine Comprehension, In Proceedings of ICLR 2017

An Example of RC Model: BiDAF



Seo et al., Bidirectional Attention Flow for Machine Comprehension, In Proceedings of ICLR 2017

An Example of RC Model: BiDAF



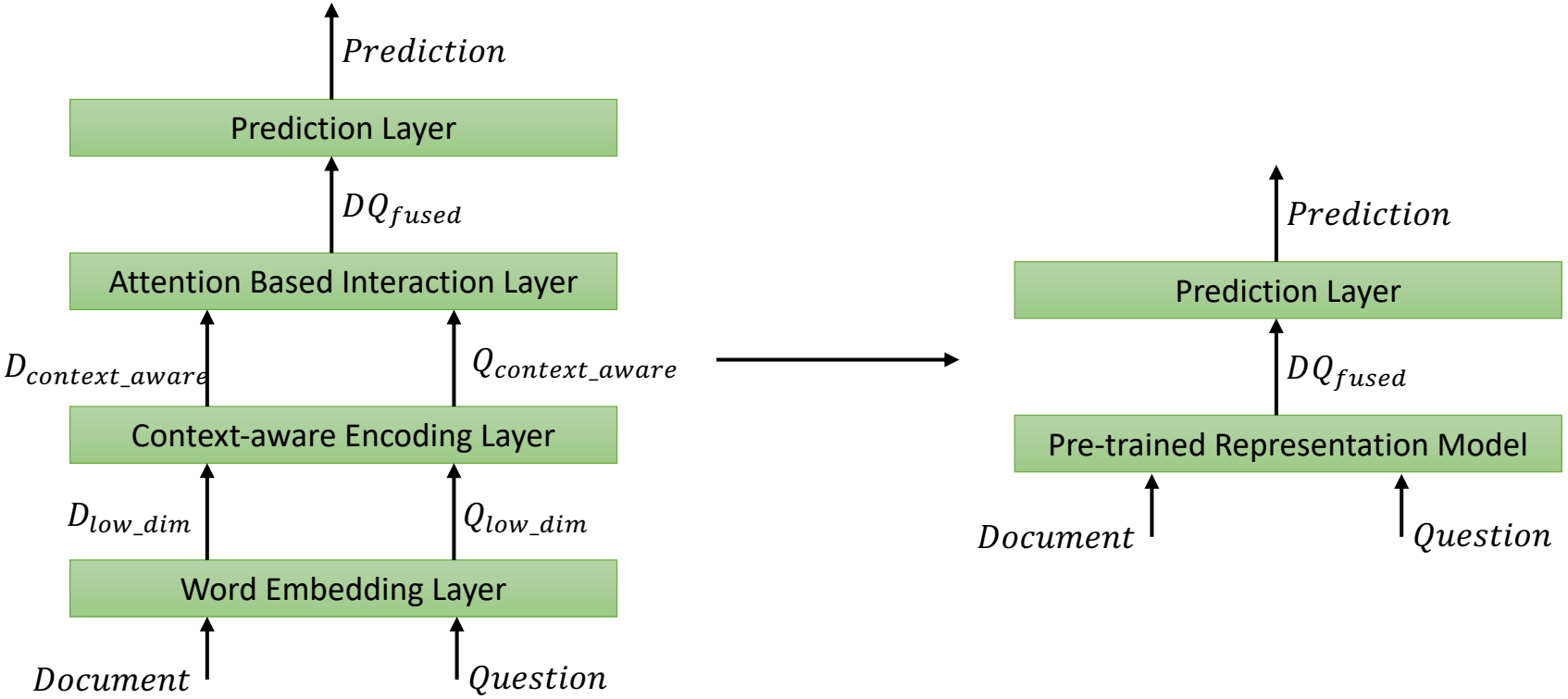
Seo et al., Bidirectional Attention Flow for Machine Comprehension, In Proceedings of ICLR 2017

Outline

- Introduction to QA
- Reading Comprehension
  - Task Definition and Dataset
  - Traditional Pipeline
  - Big-model-based Methods
- Open-domain QA

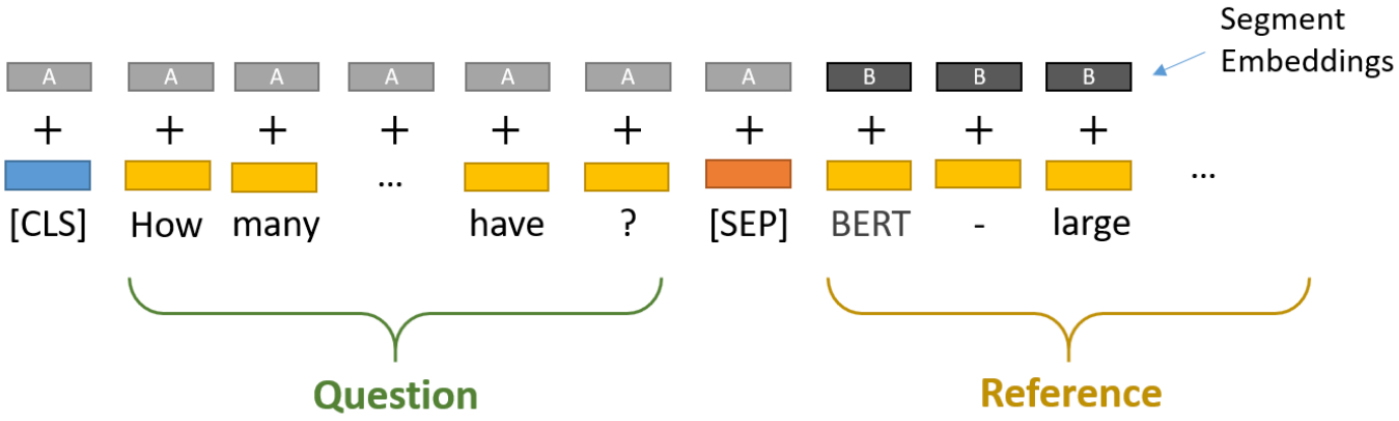
Using BERT for RC

- Use PLMs (like BERT) to replace the first three layers
  - BERT model has no RNN modules



Using BERT for RC

- Feed the concatenation of the question and the context to BERT. Get the question-aware context representation to predict the start/end of answers



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.

Using BERT for RC

- Excellent performance on SQuAD

Leaderboard

SQuAD2.0 tests the ability of a system to not only answer reading comprehension questions, but also abstain when presented with a question that cannot be answered based on the provided paragraph. How will your system compare to humans on this task?

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar & Jia et al. '18)	86.831	89.452
1 Nov 08, 2018	BERT (single model) *Google AI Language*	**80.005**	**83.061**
2 Nov 06, 2018	SLQA+BERT (single model) *Alibaba DAMO NLP* http://www.aclweb.org/anthology/P18-1158	77.003	80.209
3 Nov 08, 2018	BERT_base_aug (ensemble) *GammaLab*	76.721	79.611
4 Nov 05, 2018	MIR-MRC(F-Net) (single model) *Kangwon National University, Natural Language Processing Lab. & ForceWin, KP Lab.*	74.803	77.988
5 Sep 13, 2018	nlnet (single model) *Microsoft Research Asia*	74.238	77.022

UnifiedQA

- Unifying different QA formats
  - Four types: extractive, abstractive, multiple-choice, yes/no
  - Text-to-text format

Extractive [SQuAD]

Question: At what speed did the turbine operate?
Context: (Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...
Gold answer: 16,000 rpm

Abstractive [NarrativeQA]

Question: What does a drink from narcissus's spring cause the drinker to do?
Context: Mercury has awakened Echo, who weeps for Narcissus, and states that a drink from Narcissus's spring causes the drinkers to "Grow dotingly enamored of themselves." ...
Gold answer: fall in love with themselves

Multiple-Choice [ARC-challenge]

Question: What does photosynthesis produce that helps plants grow?
Candidate Answers: (A) water (B) oxygen (C) protein (D) sugar
Gold answer: sugar

Yes/No [BoolQ]

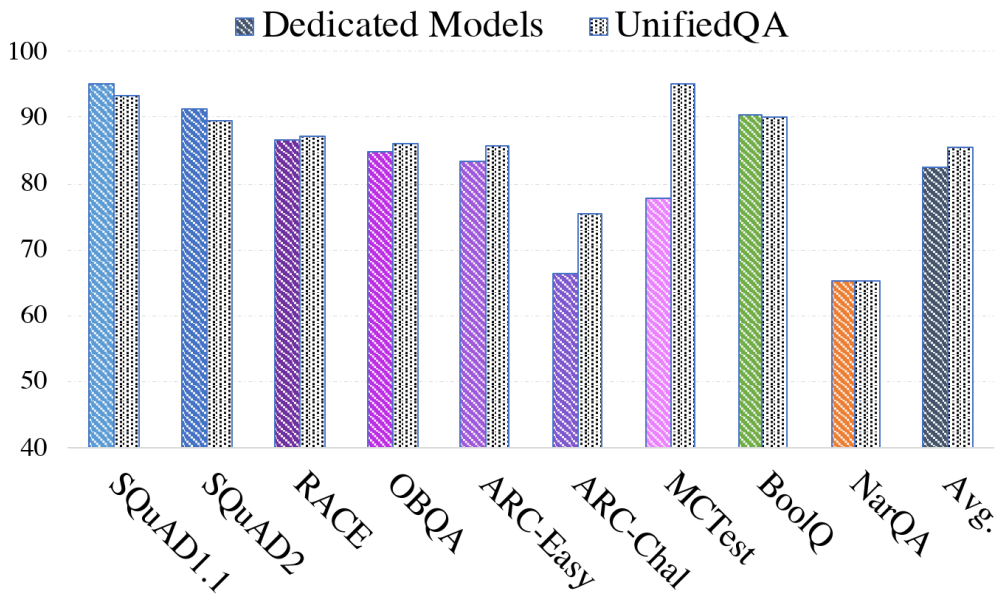
Question: Was America the first country to have a president?
Context: (President) The first usage of the word president to denote the highest official in a government was during the Commonwealth of England ...
Gold answer: no

EX	**Dataset**	SQuAD 1.1
	Input	`At what speed did the turbine operate? \n (Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...`
	Output	`16,000 rpm`
AB	**Dataset**	NarrativeQA
	Input	`What does a drink from narcissus's spring cause the drinker to do? \n Mercury has awakened Echo, who weeps for Narcissus, and states that a drink from Narcissus's spring causes the drinkers to ``Grow dotingly enamored of themselves.'' ...`
	Output	`fall in love with themselves`
MC	**Dataset**	ARC-challenge
	Input	`What does photosynthesis produce that helps plants grow? \n (A) water (B) oxygen (C) protein (D) sugar`
	Output	`sugar`
	Dataset	MCTest
	Input	`Who was Billy? \n (A) The skinny kid (B) A teacher (C) A little kid (D) The big kid \n Billy was like a king on the school yard. A king without a queen. He was the biggest kid in our grade, so he made all the rules during recess. ...`
	Output	`The big kid`
YN	**Dataset**	BoolQ
	Input	`Was America the first country to have a president? \n (President) The first usage of the word president to denote the highest official in a government was during the Commonwealth of England ...`
	Output	`no`

Khashabi et al., UNIFIEDQA: Crossing Format Boundaries with a Single QA System, Findings of EMNLP 2020

UnifiedQA

- Single QA system is on-par with, and often out-performs dedicated models
- Using prompt, we can do it easily!



Khashabi et al., UNIFIEDQA: Crossing Format Boundaries with a Single QA System, Findings of EMNLP 2020

Outline

- Introduction to QA
- Reading Comprehension
- Open-domain QA
  - Task Definition
  - Generation-based Methods
  - Retrieval-based Methods

Open-domain QA

- RC assumes that any question has a short piece of relevant text, which is not always true
- In open-domain QA, the model should be able to find relevant texts from a corpus and read them
  - Wikipedia can be viewed as a large-scale corpus for factoid question
- Goal: build an end-to-end QA system that can use full Wikipedia to answer any factoid question

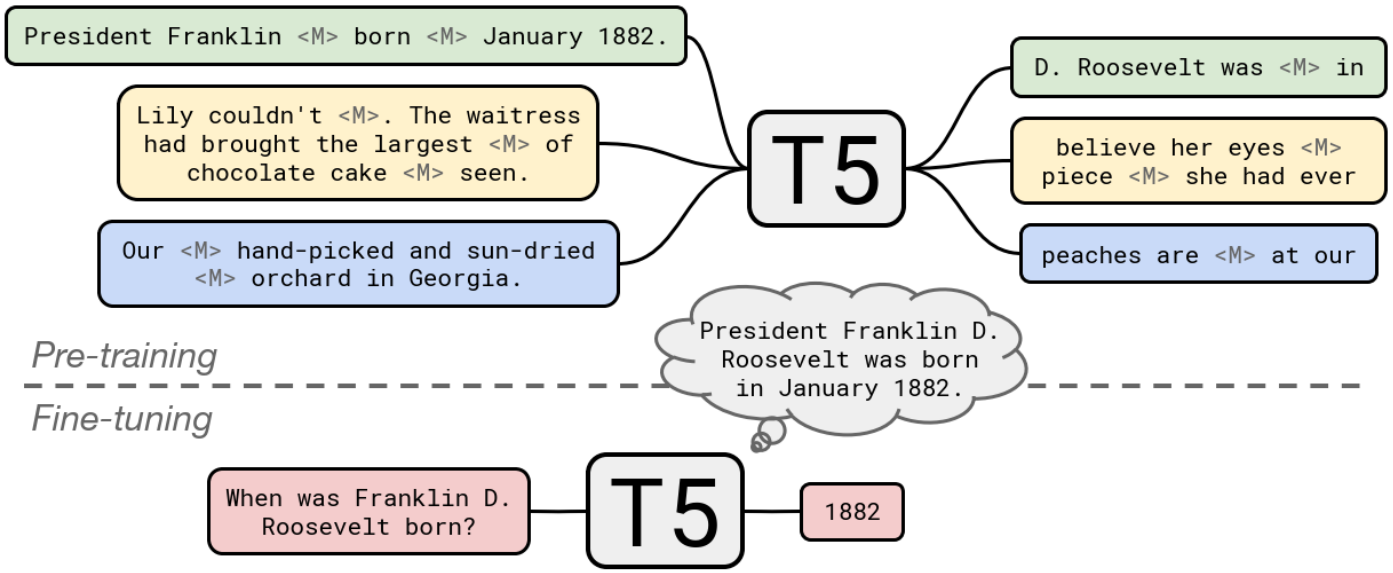
Chen et al., Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of ACL 2017

Outline

- Introduction to QA
- Reading Comprehension
- Open-domain QA
  - Task Definition
  - Generation-based Methods
  - Retrieval-based Methods

Answer Questions with Big Models

- GPT-3, T5, etc. can generate answers directly



Roberts et al., How Much Knowledge Can You Pack Into the Parameters of a Language Model?, EMNLP 2020

Answer Questions with Big Models

- Fine-tune T5 on open-domain QA
- Achieve competitive performance
- Bigger models perform better
- “Power of scale”

Table 1: Scores achieved by fine-tuning T5 on the open-domain Natural Questions (NQ), WebQuestions (WQ), and TriviaQA (TQA) tasks.

	NQ	WQ	TQA	
			dev	test
Chen et al. (2017)	–	20.7	–	–
Lee et al. (2019)	33.3	36.4	47.1	–
Min et al. (2019a)	28.1	–	50.9	–
Min et al. (2019b)	31.8	31.6	55.4	–
Asai et al. (2019)	32.6	–	–	–
Ling et al. (2020)	–	–	35.7	–
Guu et al. (2020)	40.4	40.7	–	–
Févry et al. (2020)	–	–	43.2	53.4
Karpukhin et al. (2020)	**41.5**	42.4	**57.9**	–
T5-Base	25.9	27.9	23.8	29.1
T5-Large	28.5	30.6	28.7	35.9
T5-3B	30.4	33.6	35.1	43.4
T5-11B	32.6	37.2	42.3	50.1
T5-11B + SSM	34.8	40.8	51.0	60.5
T5.1.1-Base	25.7	28.2	24.2	30.6
T5.1.1-Large	27.3	29.5	28.5	37.2
T5.1.1-XL	29.5	32.4	36.0	45.1
T5.1.1-XXL	32.8	35.6	42.9	52.5
T5.1.1-XXL + SSM	35.2	**42.8**	51.9	**61.6**

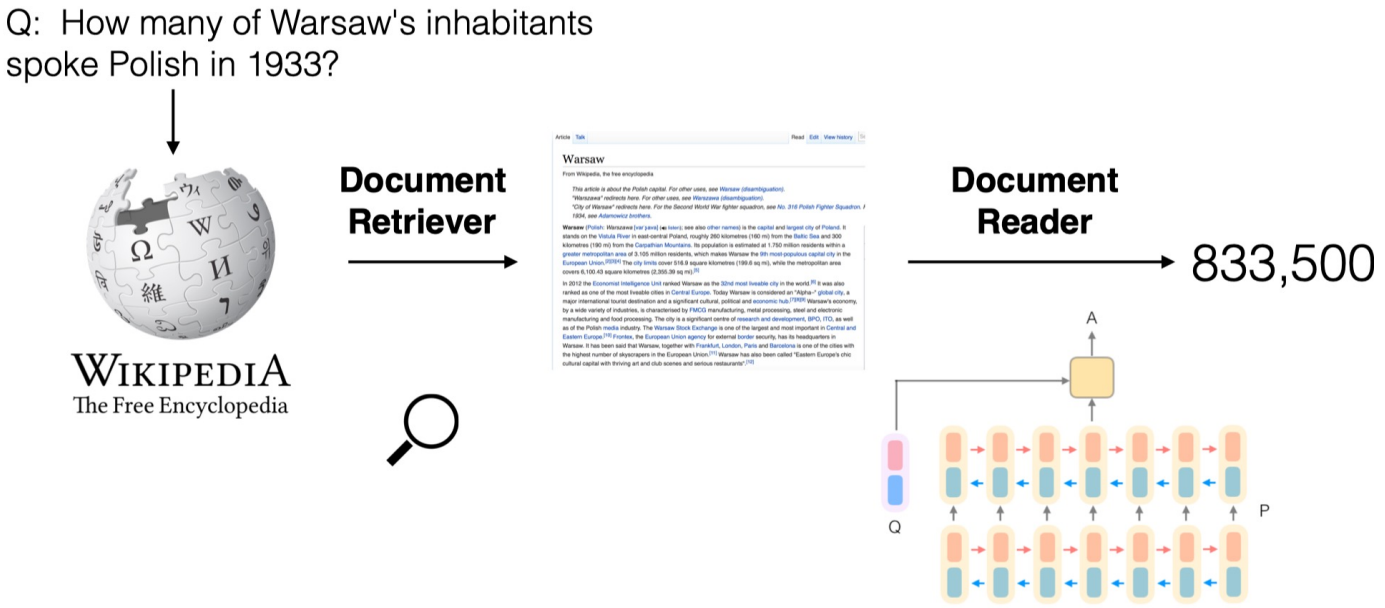
Roberts et al., How Much Knowledge Can You Pack Into the Parameters of a Language Model?, EMNLP 2020

Outline

- Introduction to QA
- Reading Comprehension
- Open-domain QA
  - Task Definition
  - Generation-based Methods
  - Retrieval-based Methods

Open-domain QA

- Document Retriever + Document Reader
  - Document retriever: finding relevant articles from 5 million Wikipedia articles
  - Document reader (reading comprehension system): identifying the answer spans from those articles



Chen et al., Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of ACL 2017

Document Retriever

- Return 5 Wikipedia articles given any question
- Features:
  - TF-IDF bag-of-words vectors
  - Efficient bigram hashing (Weinberger et al., 2009)
- Better performance than Wikipedia search: (hit@5)

Dataset	Wiki Search	Doc. Retriever	
		plain	+bigrams
SQuAD	62.7	76.1	**77.8**
CuratedTREC	81.0	85.2	**86.0**
WebQuestions	73.7	**75.5**	74.4
WikiMovies	61.7	54.4	**70.3**

Chen et al., Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of ACL 2017

Document Reader

- Simple reading comprehension model
- Features:
  - Word embeddings
  - Exact match features: whether the word appears in question
  - Token features: POS, NER, term frequency
  - Aligned question embedding
- Using Shared-Norm for multiple documents

Chen et al., Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of ACL 2017

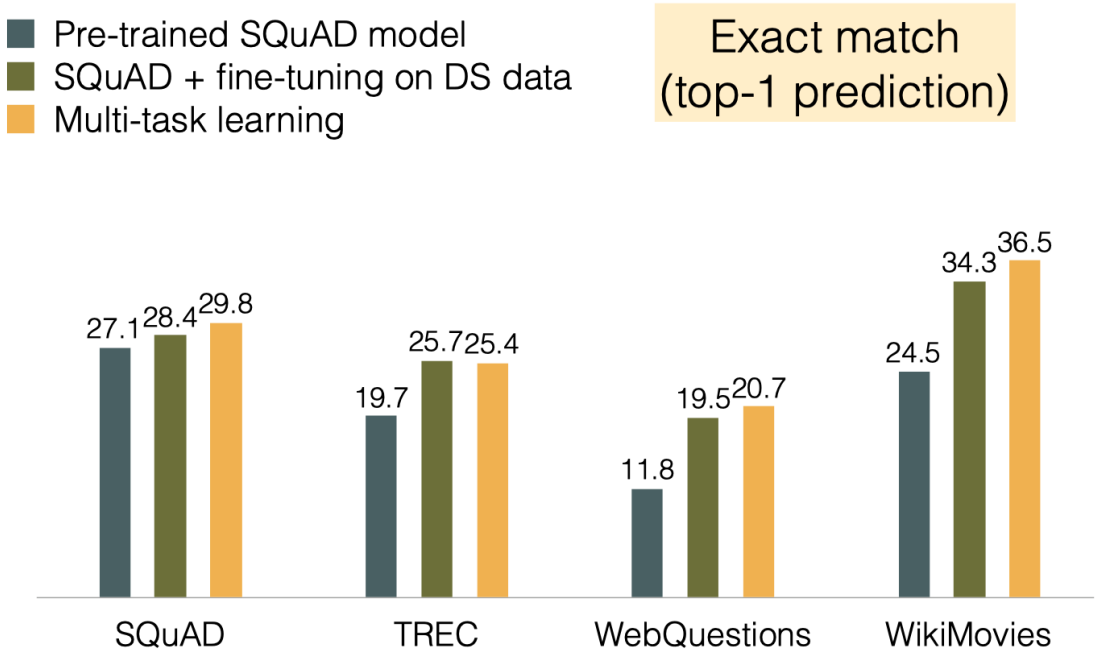
Distance Supervision

- For a given question, automatically associate paragraphs including the answer span to this question

Dataset	Example	Article / Paragraph
SQuAD	**Q**: How many provinces did the Ottoman empire contain in the 17th century? **A**: 32	**Article**: Ottoman Empire **Paragraph**: ... At the beginning of the 17th century the empire contained 32 provinces and numerous vassal states. Some of these were later absorbed into the Ottoman Empire, while others were granted various types of autonomy during the course of centuries.
CuratedTREC	**Q**: What U.S. state's motto is "Live free or Die"? **A**: New Hampshire	**Article**: Live Free or Die **Paragraph**: "Live Free or Die" is the official motto of the U.S. state of New Hampshire, adopted by the state in 1945. It is possibly the best-known of all state mottos, partly because it conveys an assertive independence historically found in American political philosophy and partly because of its contrast to the milder sentiments found in other state mottos.
WebQuestions	**Q**: What part of the atom did Chadwick discover?† **A**: neutron	**Article**: Atom **Paragraph**: ... The atomic mass of these isotopes varied by integer amounts, called the whole number rule. The explanation for these different isotopes awaited the discovery of the neutron, an uncharged particle with a mass similar to the proton, by the physicist James Chadwick in 1932. ...
WikiMovies	**Q**: Who wrote the film Gigli? **A**: Martin Brest	**Article**: Gigli **Paragraph**: Gigli is a 2003 American romantic comedy film written and directed by Martin Brest and starring Ben Affleck, Jennifer Lopez, Justin Bartha, Al Pacino, Christopher Walken, and Lainie Kazan.

Results

- Reasonable performance across all four datasets
- Models using DS outperform models trained on SQuAD
  - Multi-task: Training on SQuAD + DS data



Chen et al., Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of ACL 2017

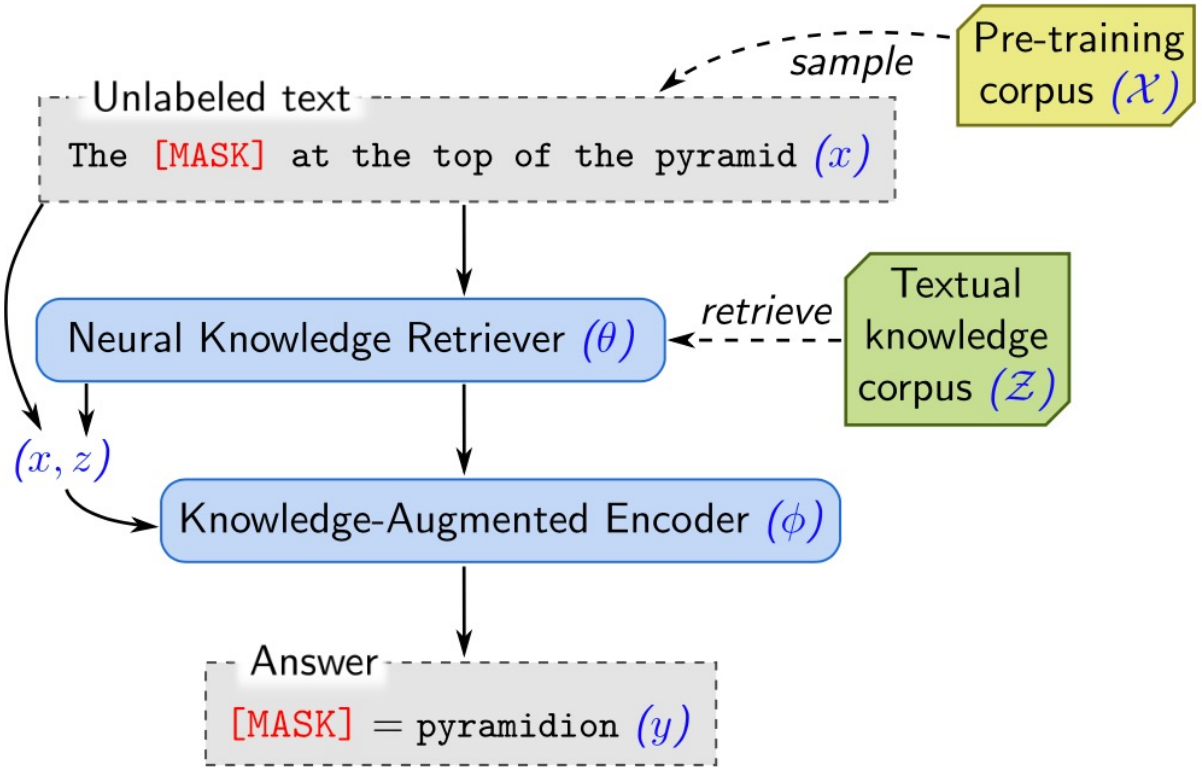
Retrieval-Augmented Language Model Pre-Training

- REALM
  - Augment language pre-training with a neural knowledge retriever that retrieves knowledge from a textual knowledge corpus (e.g., Wikipedia)
  - Allow the model to attend documents from a large corpus during pre-training, fine-tuning and inference

Guu, Kelvin, et al. "Realm: Retrieval-augmented language model pre-training." ICML (2020).

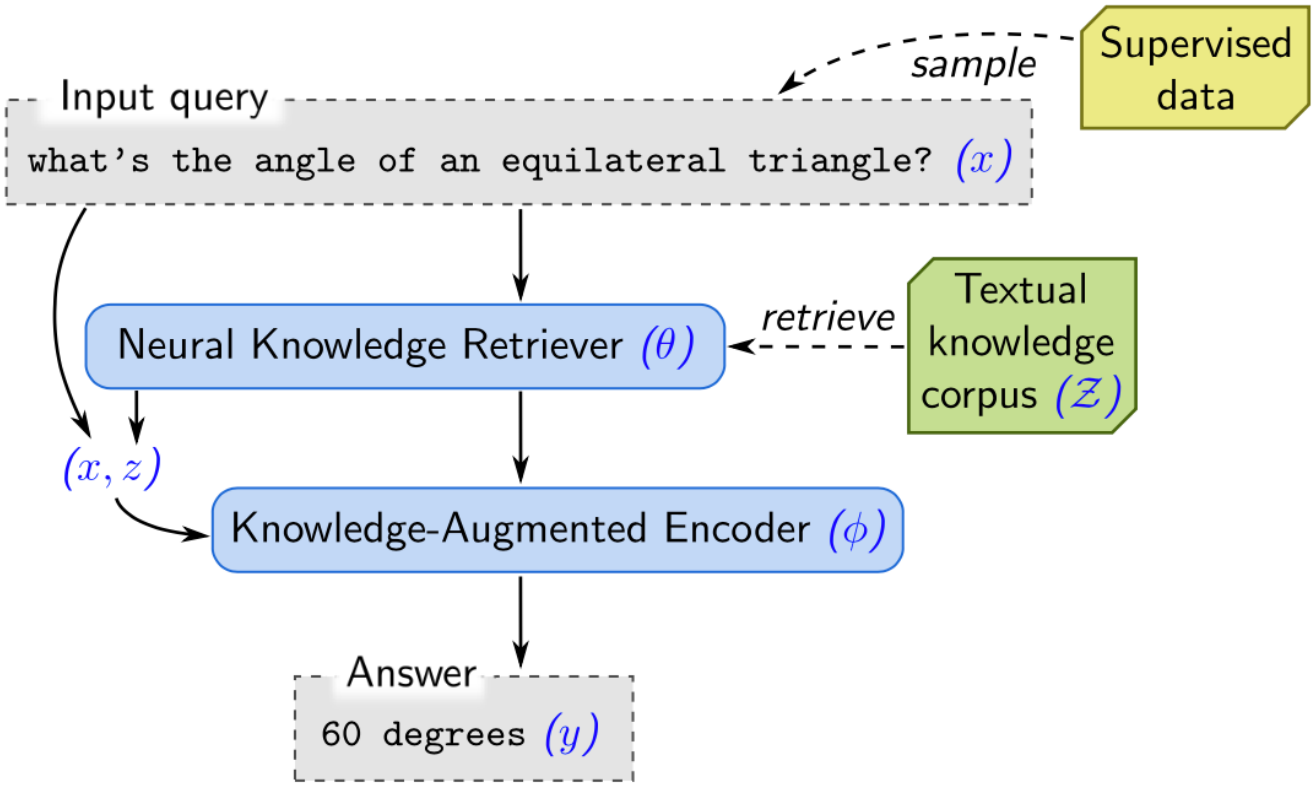
Retrieval-Augmented Language Model Pre-Training

- Pre-training of REALM
  - The knowledge retriever and knowledge-augmented encoder are jointly pre-trained on the unsupervised language modeling task



Retrieval-Augmented Language Model Pre-Training

- Fine-tuning of REALM
  - The pre-trained retriever (θ) and encoder (ϕ) are fine-tuned on a task of primary interest, in a supervised way



Retrieval-Augmented Language Model Pre-Training

- Excellent performance for open-domain QA

Name	Pre-training	NQ (79k/4k)	WQ (3k/2k)	CT (1k /1k)	# params
BERT-Baseline (Lee et al., 2019)	BERT	26.5	17.7	21.3	110m
T5 (base) (Roberts et al., 2020)	T5 (Multitask)	27.0	29.1	-	223m
T5 (large) (Roberts et al., 2020)	T5 (Multitask)	29.8	32.2	-	738m
T5 (11b) (Roberts et al., 2020)	T5 (Multitask)	34.5	37.4	-	11318m
DrQA (Chen et al., 2017)	N/A	-	20.7	25.7	34m
HardEM (Min et al., 2019a)	BERT	28.1	-	-	110m
GraphRetriever (Min et al., 2019b)	BERT	31.8	31.6	-	110m
PathRetriever (Asai et al., 2019)	MLM	32.6	-	-	110m
ORQA (Lee et al., 2019)	ICT+BERT	33.3	36.4	30.1	330m
Ours ($\mathcal{X}$ = Wikipedia, $\mathcal{Z}$ = Wikipedia)	REALM	39.2	40.2	**46.8**	330m
Ours ($\mathcal{X}$ = CC-News, $\mathcal{Z}$ = Wikipedia)	REALM	**40.4**	**40.7**	42.9	330m

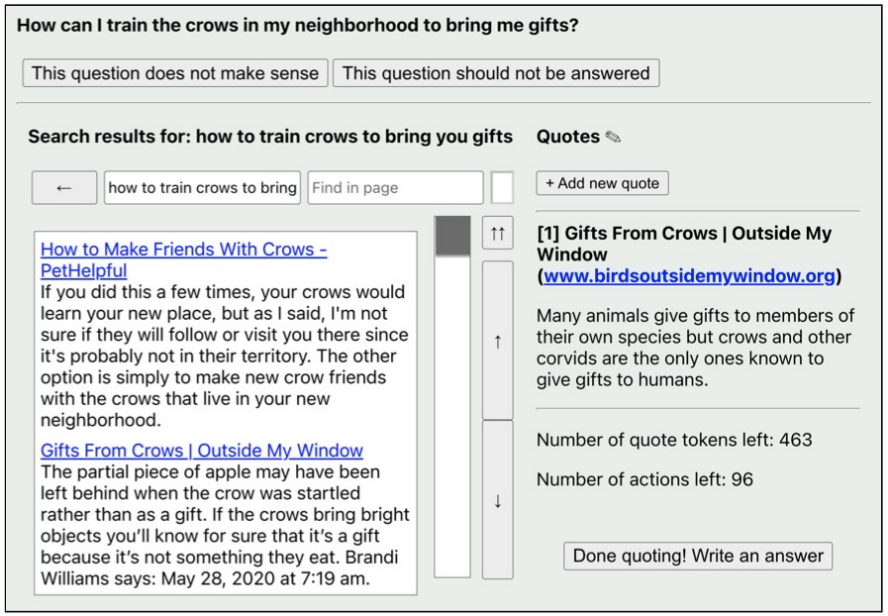
Document Retrieval and Synthesis with GPT-3

- WebGPT
  - Outsource document retrieval to the Microsoft Bing Web Search API
  - Utilize unsupervised pre-training to achieve high-quality document synthesis by fine-tuning GPT-3
  - Create a text-based web-browsing environment that both humans and language models can interact with

Nakano, Reiichiro, et al. "WebGPT: Browser-assisted question-answering with human feedback." arXiv preprint arXiv:2112.09332 (2021).

Document Retrieval and Synthesis with GPT-3

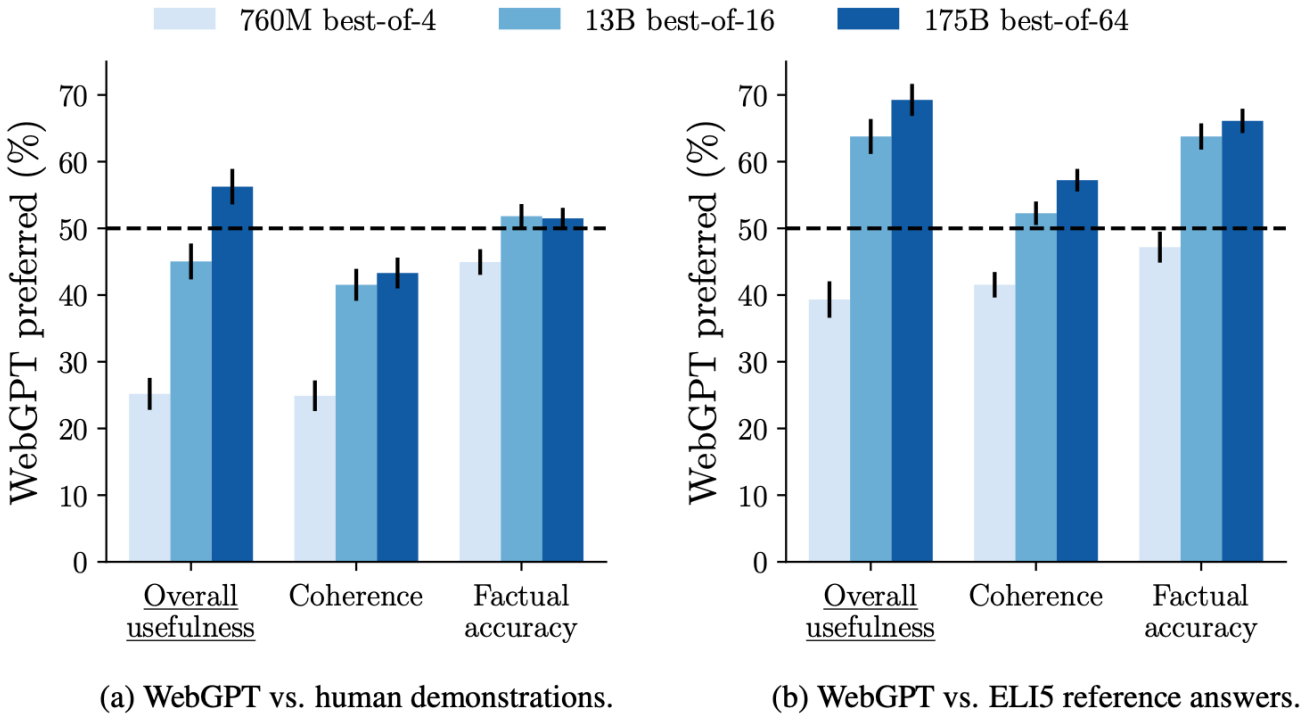
- Pipeline:
  - Fine-tune GPT-3 to imitate human behaviors when using the web-browser
  - Write down key references when browsing
  - After browsing, generate answers with references



Text-based web-browser

Document Retrieval and Synthesis with GPT-3

- WebGPT-produced answers are more preferred than human-generated ones
- Better coherence and factual accuracy



(a) WebGPT vs. human demonstrations.

(b) WebGPT vs. ELI5 reference answers.

Demo

- QA with T5 using OpenPrompt
  - Zero-shot inference

```
# Conduct zero-shot inference
from openprompt import PromptForClassification

use_cuda = True
prompt_model = PromptForClassification(plm=plm, template=mytemplate, verbalizer=myverbalizer)
if use_cuda:
    prompt_model = prompt_model.cuda()

example_dataloader = PromptDataLoader(dataset=[example], template=mytemplate, tokenizer=tokenizer,
    tokenizer_wrapper_class=WrapperClass, max_seq_length=256, decoder_max_length=3,
    batch_size=1,shuffle=True, teacher_forcing=False, predict_eos_token=False,
    truncate_method="head")

# Print the prediction results of examples
for step, input_example in enumerate(example_dataloader):
    if use_cuda:
        input_example = input_example.cuda()
    logits = prompt_model(input_example)
    labels = input_example['label']
    pred = myverbalizer.label_words[(torch.argmax(logits, dim=-1).cpu().tolist())[0]]
    print(pred)
```

[6]

```
tokenizing: 1it [00:00, 658.65it/s]

[' yes']
```

Demo

- QA with T5 using OpenPrompt and OpenDelta
  - Delta tuning

```
from opendelta import LoraModel
delta_model = LoraModel(backbone_model=plm, modified_modules=["SelfAttention.q", "SelfAttention.v"])
delta_model.freeze_module(exclude=["deltas"], set_state_dict=True)
delta_model.log()
```

```
optimizer = AdamW(optimizer_grouped_parameters, lr=1e-4)

for epoch in range(10):
    tot_loss = 0
    for step, inputs in enumerate(train_dataloader):
        if use_cuda:
            inputs = inputs.cuda()
        logits = prompt_model(inputs)
        labels = inputs['label']
        loss = loss_func(logits, labels)
        loss.backward()
        tot_loss += loss.item()
        optimizer.step()
        optimizer.zero_grad()
        if step %100 ==1:
            print("Epoch {}, average loss: {}".format(epoch, tot_loss/(step+1)), flush=True)
```

Text Generation

Fengyu Wang

wangfy20@mails.tsinghua.edu.cn

THUNLP

Outline

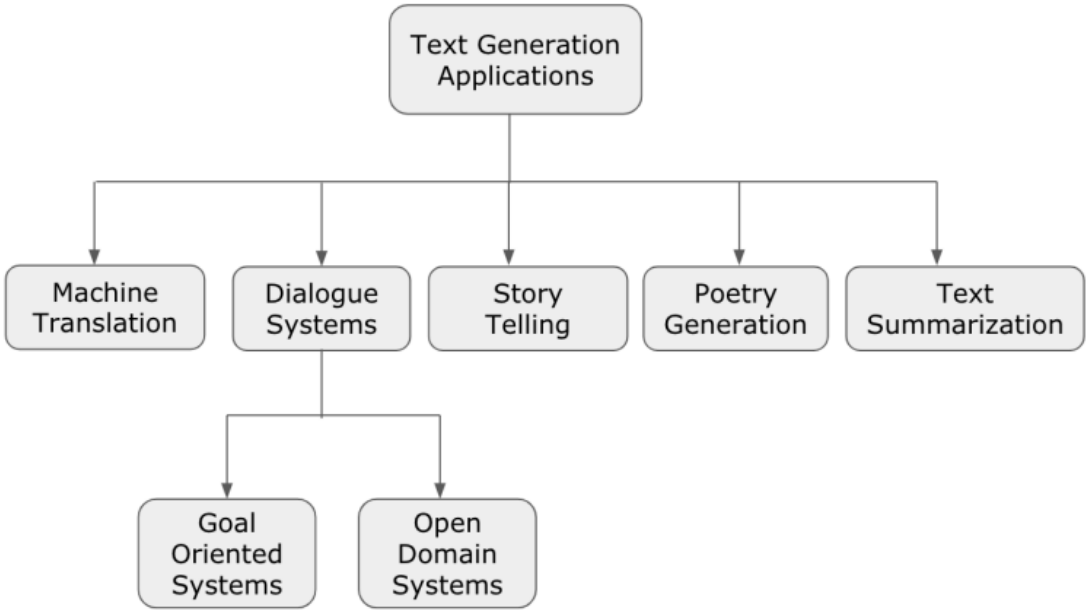
- Introduction to text generation
- Tasks of text generation
- Neural text generation
- Controllable text generation
- Text generation evaluation
- Challenges

Outline

- Introduction to text generation
- Tasks of text generation
- Neural text generation
- Controllable text generation
- Text generation evaluation
- Challenges

Introduction to Text Generation

- Formal Definition: Produce understandable texts in human languages from some underlying non-linguistic representation of information. [Reiter et al., 1997]
- Text-to-text generation and data-to-text generation are both instances of TG [Reiter et al., 1997]
- Applications under umbrella of text generation



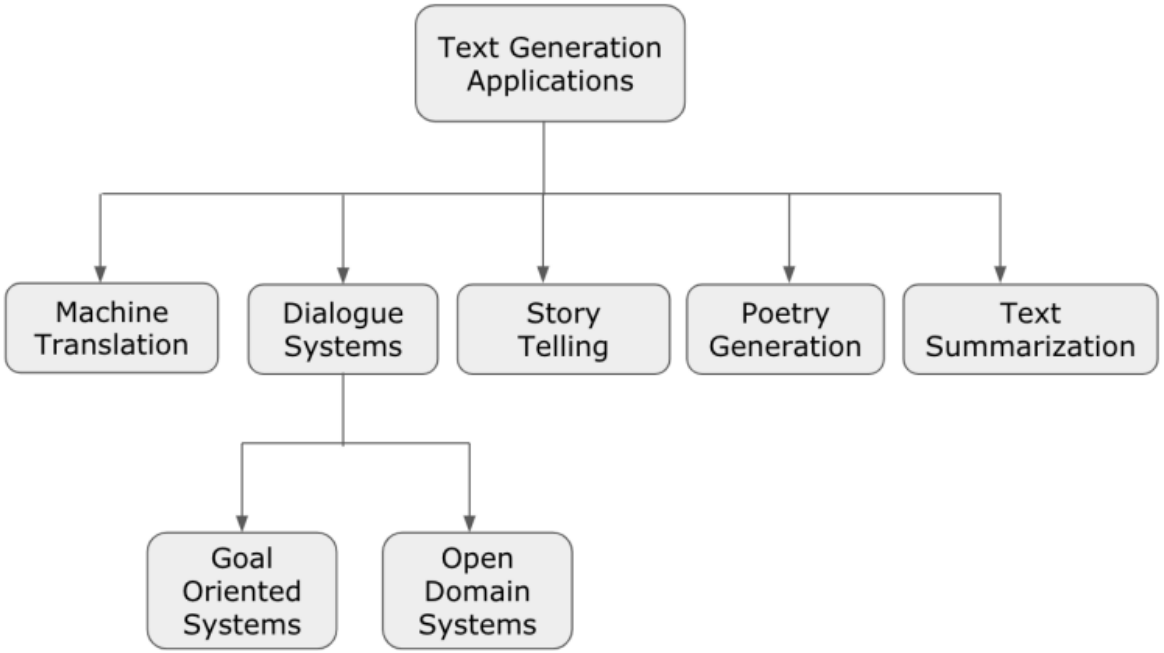
[Reiter at el. 1997, Building applied natural language generation systems.]
[Gatt at el., 2017, Survey of the State of the Art in Natural Language Generation- Core tasks, applications and evaluation]

Outline

- Introduction to text generation
- Tasks of text generation
- Neural text generation
- Controllable text generation
- Text generation evaluation
- Challenges

Text Generation Tasks

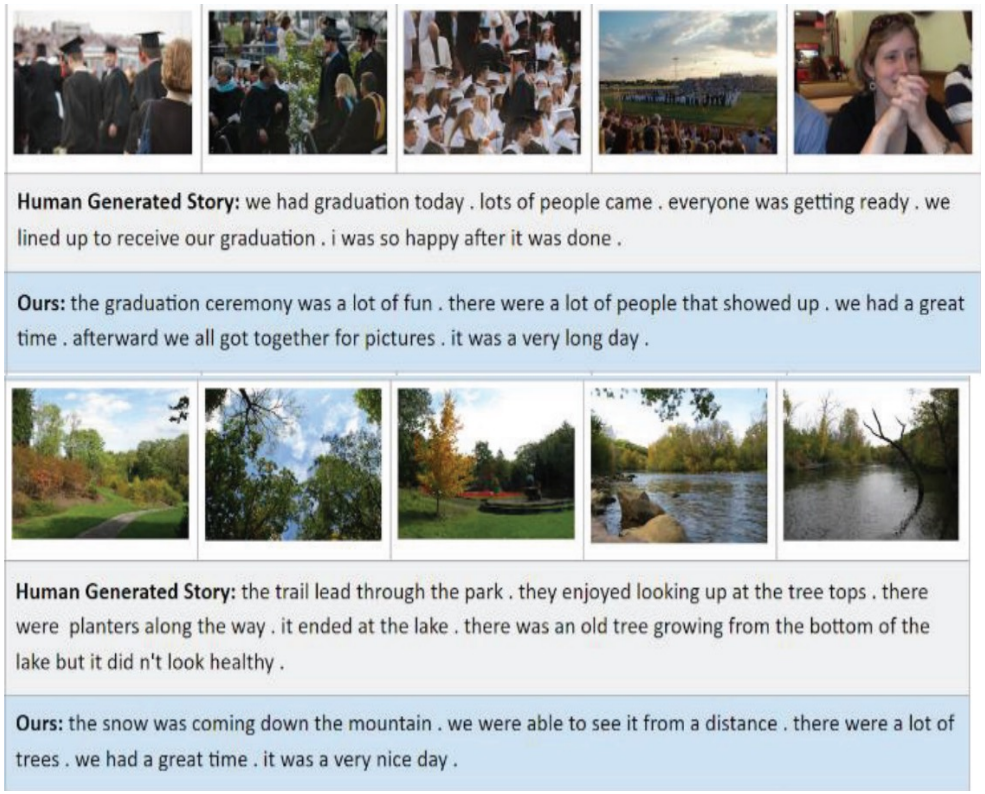
- Data-To-Text (image, table, graph)
- Dialogue
- Machine Translation
- Poetry Generation
- Style Transfer
- Storytelling
- Summarization



Gatt at el., 2017, Survey of the State of the Art in Natural Language Generation- Core tasks, applications and evaluation

TG Tasks: Data-to-Text

- Various of data forms: image; table; graph……



Human Generated Story: we had graduation today . lots of people came . everyone was getting ready . we lined up to receive our graduation . i was so happy after it was done .

Ours: the graduation ceremony was a lot of fun . there were a lot of people that showed up . we had a great time . afterward we all got together for pictures . it was a very long day .

Human Generated Story: the trail lead through the park . they enjoyed looking up at the tree tops . there were planters along the way . it ended at the lake . there was an old tree growing from the bottom of the lake but it did n't look healthy .

Ours: the snow was coming down the mountain . we were able to see it from a distance . there were a lot of trees . we had a great time . it was a very nice day .

Attribute	Value
name	james beattie
fullname	james scott beattie
birth	27 february 1978 lancaster , england
height	6 1
position	striker
currentclub	swansea city (assistant first team coach)
youthyears	1995 – 1996
youthclubs	blackburn rovers
totalcaps	443
totalgoals	131
nationalyears	1996 2003
nationalteam	england u21 england
nationalcaps	9 5
nationalgoals	4 0
manageryears	2013 – 2014
managerclubs	accrington stanley
article	james beattie (footballer)
...	...

Reference: james scott beattie (born 27 february 1978) is an english former professional footballer who played as a striker .

Base+switch+LM(R): james beattie (, born 10 july 1971) is an english former professional association footballer who played for , among others , England .

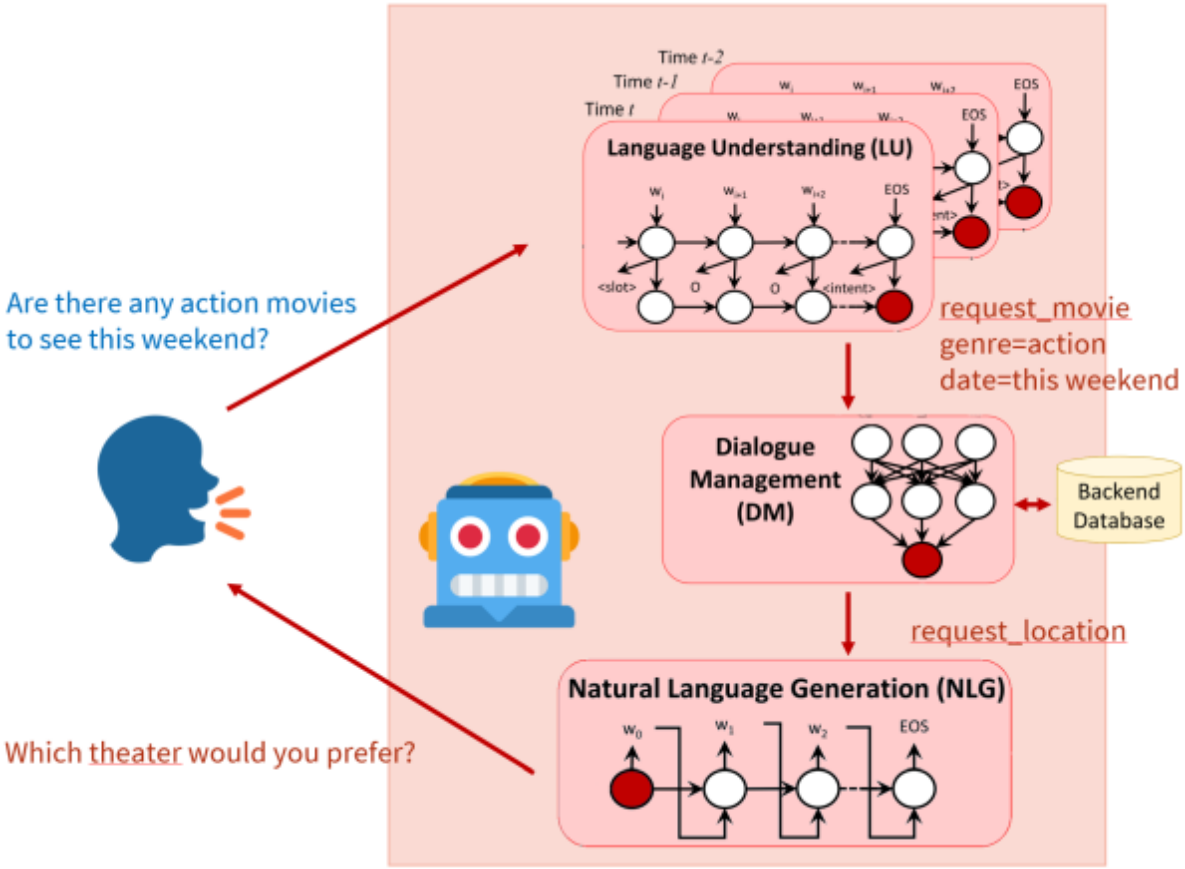
TableGPT: james beattie (born 27 february 1978 in lancaster) is a former english footballer who played as a striker .

Khushi et al, 2021, Picture Tales: An Approach for Story Generation Using a Series of Images

Heng et al, 2020, TableGPT: Few-shot Table-to-Text Generation with Table Structure Reconstruction and Content Matching

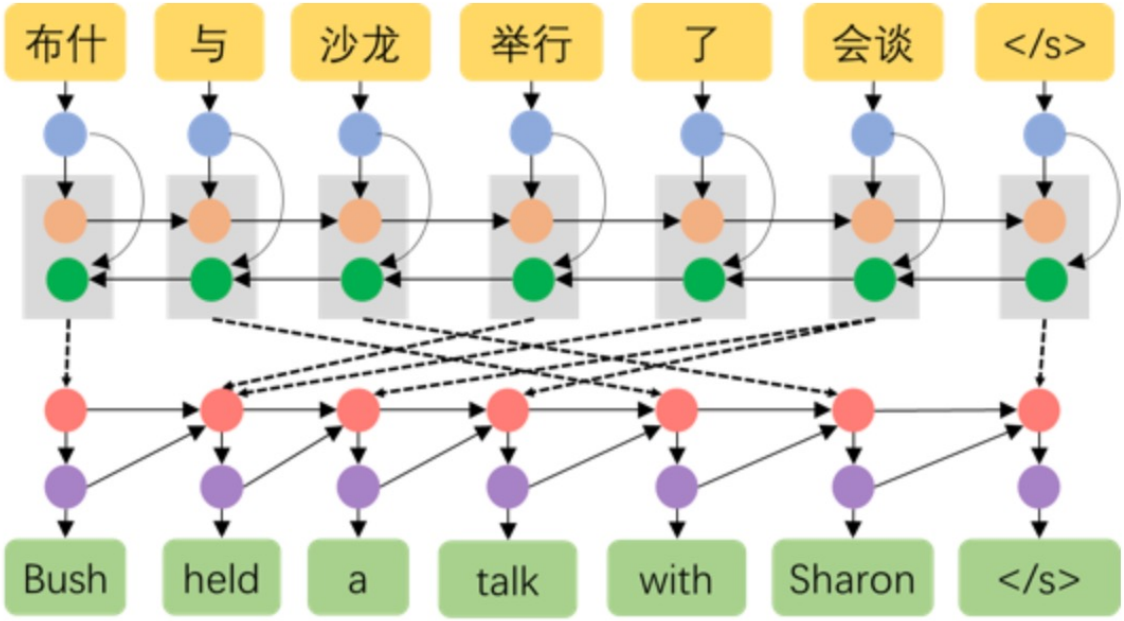
TG Tasks: Dialogue

- Generate conversations that meet the purpose in response to specific user input



TG Tasks: Machine Translation

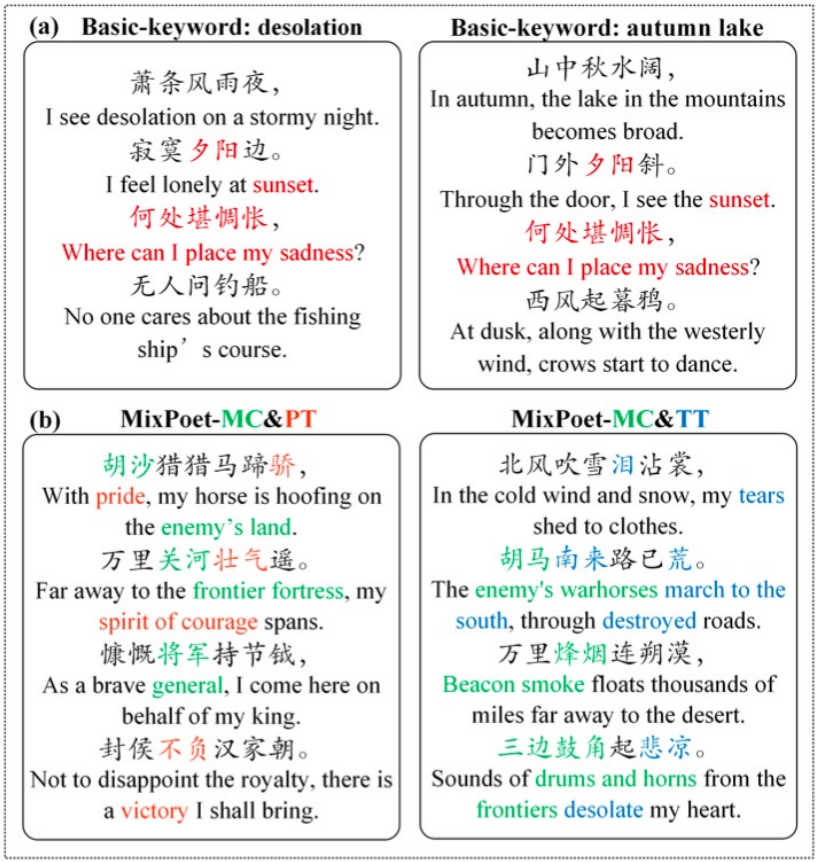
- Translate natural language sentences into a target language



https://nlp.csai.tsinghua.edu.cn/project/thumt/

TG Tasks: Poetry Generation

- Generate texts that meet the rhythmic requirements of the poem, based on keywords, or emotional control, etc.



Yi, Li et al, 2020, MixPoet: Diverse Poetry Generation via Learning Controllable Mixed Latent Space, Proceedings of The Thirty-Fourth AAAI Conference on Artificial Intelligence

TG Tasks: Style Transfer

- Control the style of input text while preserve the the meaning

Task	Attribute Values	Datasets	Size	Pa?
Style Features				
Formality	Informal↔Formal	GYAFC[3] (Rao and Tetreault 2018)	50K	✓
		XFORMAL[4] (Briakou et al. 2021b)	1K	✓
Politeness	Impolite→Polite	Politeness[5] (Madaan et al. 2020)	1M	✗
Gender	Masculine↔Feminine	Yelp Gender[6] (Prabhumoye et al. 2018)	2.5M	✗
Humor& Romance	Factual↔Humorous↔ Romantic	FlickrStyle[7] (Gan et al. 2017)	5K	✓
Biasedness	Biased→Neutral	Wiki Neutrality[8] (Pryzant et al. 2020)	181K	✓
Toxicity	Offensive→Non-offensive	Twitter (dos Santos, Melnyk, and Padhi 2018)	58K	
		Reddit (dos Santos, Melnyk, and Padhi 2018)	224K	✗
		Reddit Politics (Tran, Zhang, and Soleymani 2020)	350K	
Authorship	Shakespearean↔Modern	Shakespeare (Xu et al. 2012)	18K	✓
	Different Bible translators	Bible[9] (Carlson, Riddell, and Rockmore 2018)	28M	
Simplicity	Complicated→Simple	PWKP (Zhu, Bernhard, and Gurevych 2010)	108K	✓
		Expert (den Bercken, Sips, and Lofi 2019)	2.2K	✓
		MIMIC-III[10] (Weng, Chung, and Szolovits 2019)	59K	✗
		MSD[11] (Cao et al. 2020)	114K	✓
Engagingness	Plain→Attractive	Math[12] (Koncel-Kedziorski et al. 2016)	<1K	✓
		TitleStylist[13] (Jin et al. 2020a)	146K	✗
Content Preferences				
Sentiment	Positive↔Negative	Yelp[14] (Shen et al. 2017)	250K	✗
		Amazon[15] (He and McAuley 2016)	277K	
Topic	Entertainment↔Politics	Yahoo! Answers[16] (Huang et al. 2020)	153K	✗
Politics	Democratic↔Republican	Political[17] (Voigt et al. 2018)	540K	✗

Jin, Di, et al. "Deep learning for text style transfer: A survey." Computational Linguistics 48.1 (2022): 155-205.

TG Tasks: Storytelling

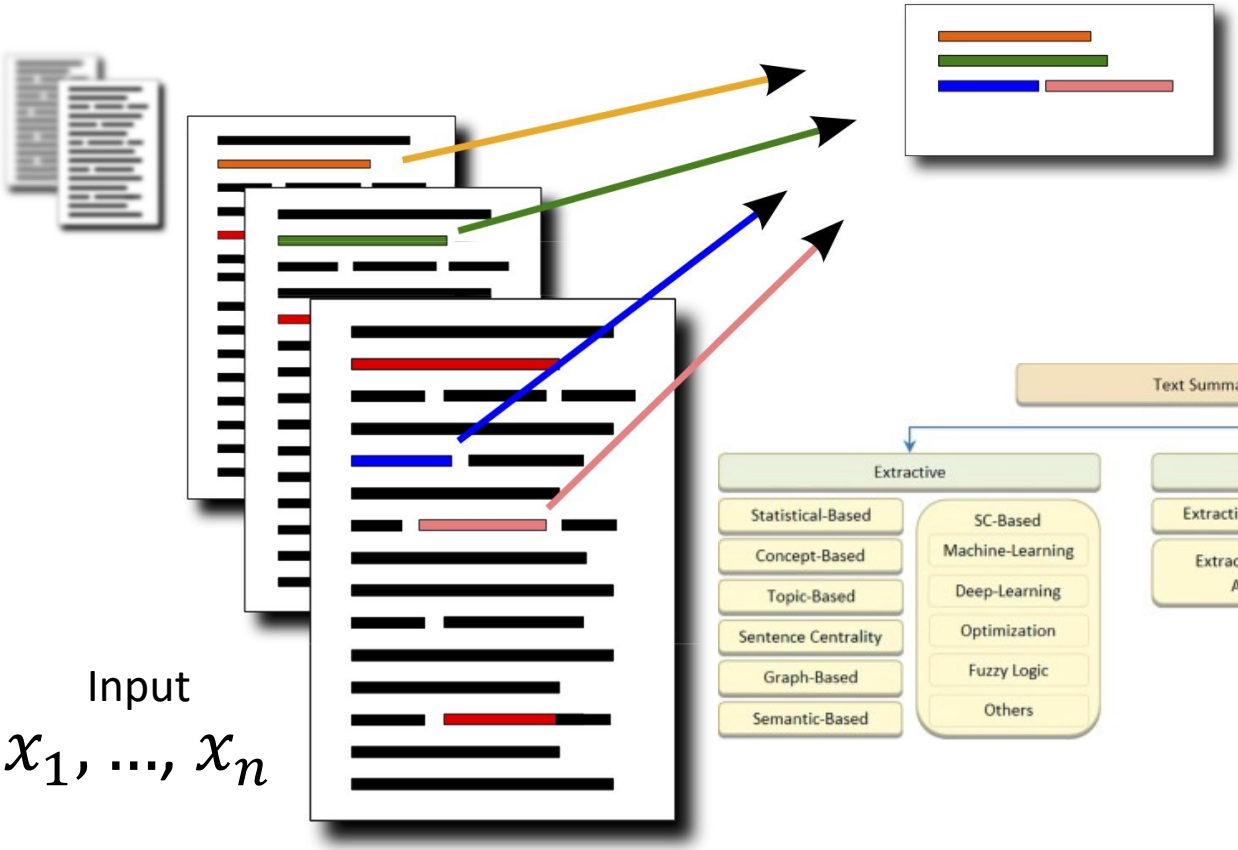
- Generate a story that meets the attribute requirements based on the given keywords, story line, etc.

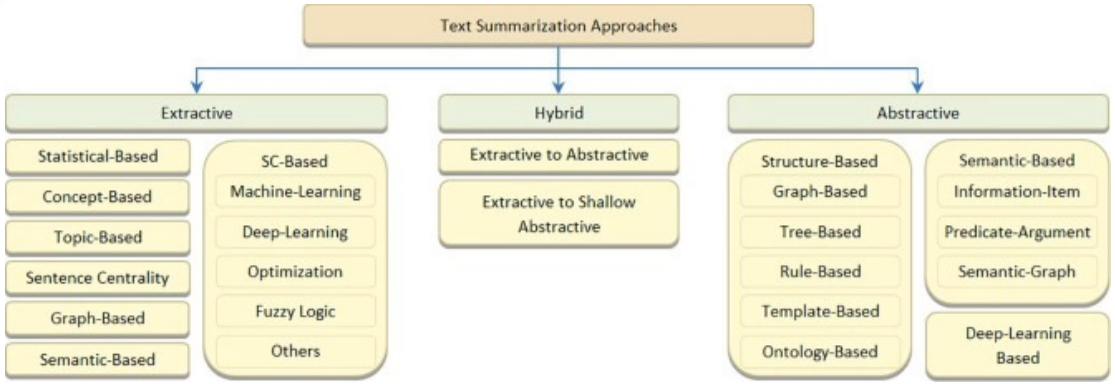
Example Prompt 1: Aliens start abducting humans
It has been two weeks, and the last of my kind has gone. It is only a matter of time until there will be nothing left. I'm not sure what the hell is going on... I can't think. I can hear a distant scream. I think of a strange, alien sound. I try to speak, but am interrupted by something, something that feels like a drum, I ca not tell. I mean I'm just a little bit older than an average human. But I can, and I can feel the vibrations . I hear the sound of a ship approaching. The ground quakes at the force of the impact, and a loud explosion shatters the silence.
Example Prompt 2: The scientists have discovered something terrible .
The scientist stood there, a little dazed as he stared. "What is it?" He asked. "This...this...Thing...This is a virus. A chemical that can destroy entire planet and it is a very small, complex, chemical that could destroy any planet." The scientist replied. His lab assistant looked down at the tablet. "I've just discovered it. I can't believe it. It looks like it's made of some sort of chemical that's very dangerous." " Well, there's a virus on the ground. It's very effective..." " I can't believe what it is." he said, pointing to the scientist . " We don't know what this thing is. We haven't seen anything like it . We can't even see anything like this. " Dr. Jones stared at the scientist for a moment. "What do you mean what does it do ?" "It...It 's a monster."

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical Neural Story Generation." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018.

TG Tasks: Summarization

- Summarize the input text with selected part of input text (extractive) or with generated text (abstractive)





El-Kassas, Wafaa S., et al. "Automatic text summarization: A comprehensive survey." Expert Systems with Applications 2021.

Outline

- Introduction to text generation
- Tasks of text generation
- Neural text generation
  - Language modeling
  - Sequence to sequence
  - Autoregressive models
  - Non-autoregressive models
  - Decoding strategy
- Controllable text generation
- Text generation evaluation
- Challenges

Language Modeling

- $P(y_t | y_1, y_2, \cdots, y_{t-1})$
- Predict next word given the words so far
- A system that produces this probability distribution is called a Language Model
- We use language models every day, such as …

Web search engine / …

Translation service / mail agent / …

Gatt at el., 2017, Survey of the State of the Art in Natural Language Generation- Core tasks, applications and evaluation
https://lena-voita.github.io/nlp_course/language_modeling.html#generation

Conditional Language Modeling

- Conditional Language Modeling: $P(y_t|y_1, y_2, \ldots, y_{t-1}, x)$
  - The task of predicting the next word, given the words so far, and also some other input
  - x input/source
  - y output/target sequence

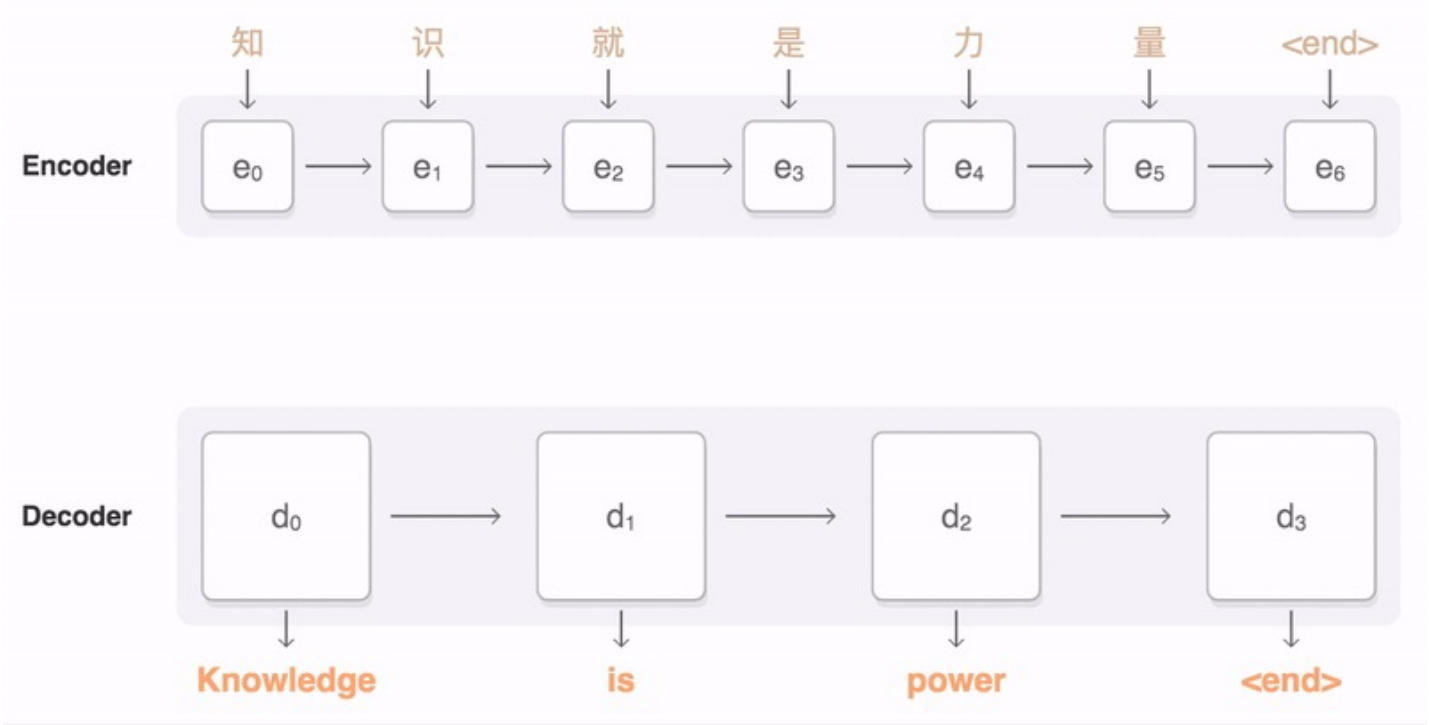
Task	X (example)	Y (example)
language modeling	none (empty sequence)	tokens from news corpus
machine translation	source sequence in English	target sequence in French
grammar correction	noisy, ungrammatical sentence	corrected sentence
summarization	body of news article	headline of article
dialogue	conversation history	next response in turn
	Related tasks (may be outside scope of this guide)	
speech transcription	audio / speech features	text transcript
image captioning	image	caption describing image
question answering	supporting text + knowledge base + question	answer

Xie at el., 2017, Neural Text Generation: A Practical Guide

Seq2seq

- Seq2seq is an example of conditional language model
- Encoder produces a representation of the source sentence
- Decoder is a language model that generates target sentence conditioned on encoding

$$P(y|x) = P(y_1|x)P(y_2|y_1, x) \dots P(y_T|y_1, \dots, y_{T-1}, x)$$



https://google.github.io/seq2seq/

Seq2seq

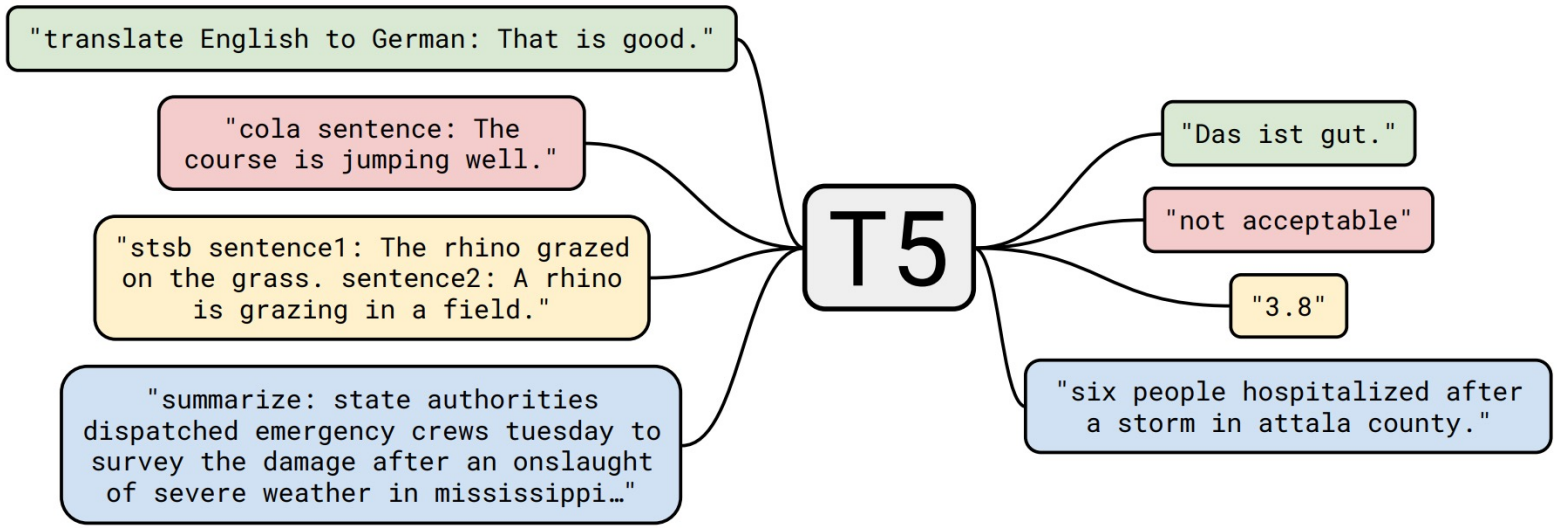
- seq2seq can be easily modeled using a single neural network

  and trained in an end-to-end fashion

- seq2seq training by teacher forcing
  - Training: predict next word based on previous ground-truth tokens, instead of predicted tokens
  - Testing: predict next word based on previous predicted tokens
- Exposure Bias: The gap between training & testing distribution

Text-to-Text-Transfer-Transformer (T5)

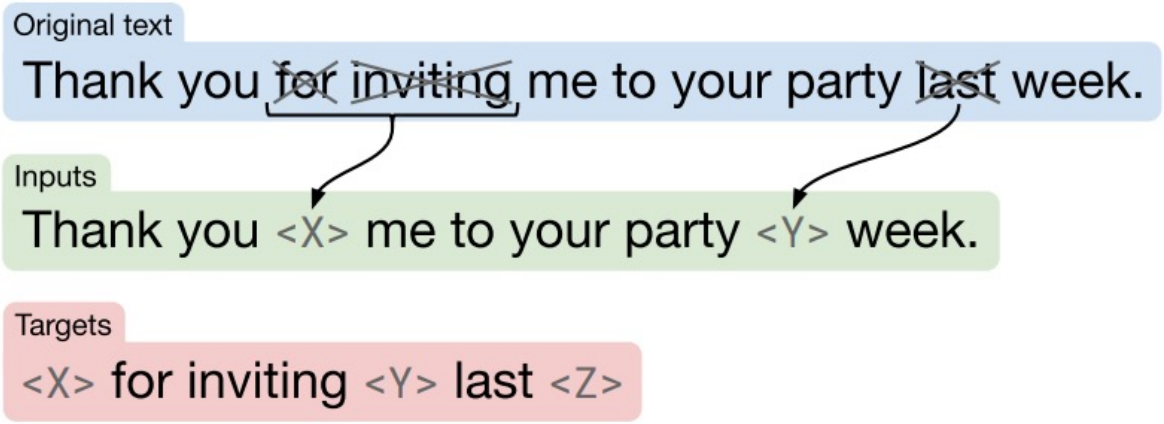
- A Shared Text-To-Text Framework: reframing all NLP tasks into a unified text-to-text-format where the input and output are always text strings



Raffel, Colin, et al. "Exploring the limits of transfer learning with a unified text-to-text transformer." J. Mach. Learn. Res. 21.140 (2020): 1-67.

Text-to-Text-Transfer-Transformer (T5)

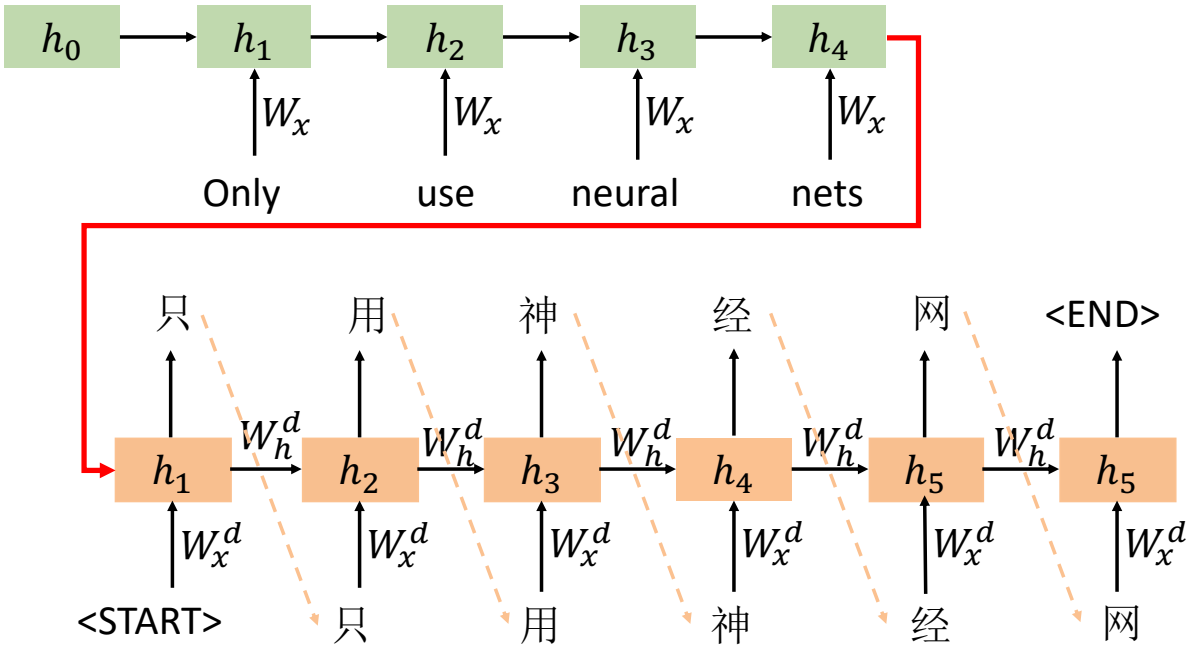
- Training objective
  - Colossal Clean Crawled Corpus (C4) dataset, a cleaned version of Common Crawl (deduplication, discarding incomplete sentences, and removing offensive or noisy content)



  - Unlabeled data

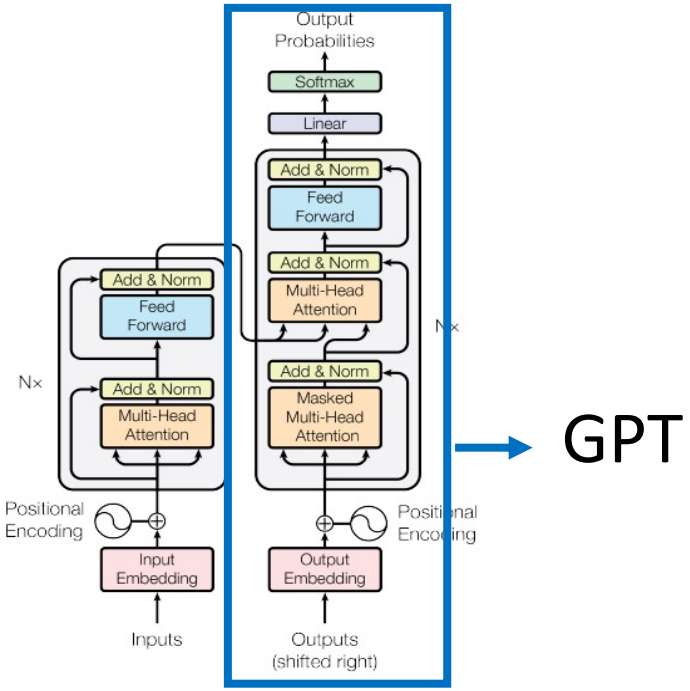
Autoregressive Generation

- Generate future values from past values
- Given source $x = (x_1, x_2, \ldots, x_n)$, target y $= (y_1, y_2, \ldots, y_m)$
- Generate Sequentially: $P(y|x) = \prod_{t=1}^{m} P(y_t|y_{<t}, x, \theta_{enc}, \theta_{dec})$



Generative Pre-Trained Transformer (GPT)

- GPT-1: Improving language understanding by generative pre-training
- GPT-2: Language models are unsupervised multitask learners
- GPT-3: Language models are few shot learners



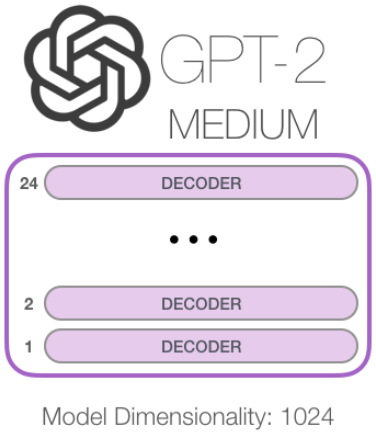
GPT

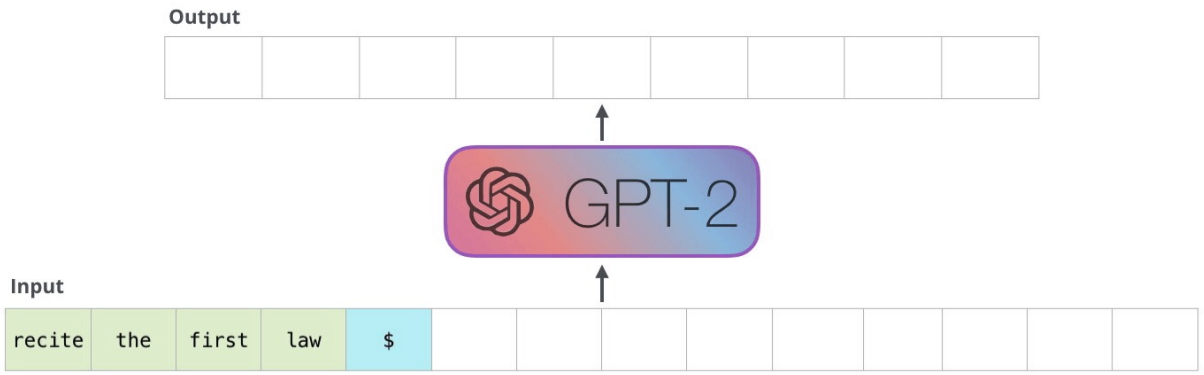
```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← examples
3  peppermint => menthe poivrée        ←
4  plush girafe => girafe peluche      ←
5  cheese =>                           ← prompt
```


Radford, Alec, et al. "Improving language understanding by generative pre-training." (2018).
Radford, Alec, et al. "Language models are unsupervised multitask learners." OpenAI blog 1.8 (2019): 9.
Brown, Tom, et al. "Language models are few-shot learners." Advances in neural information processing systems 33 (2020): 1877-1901.

GPT-2

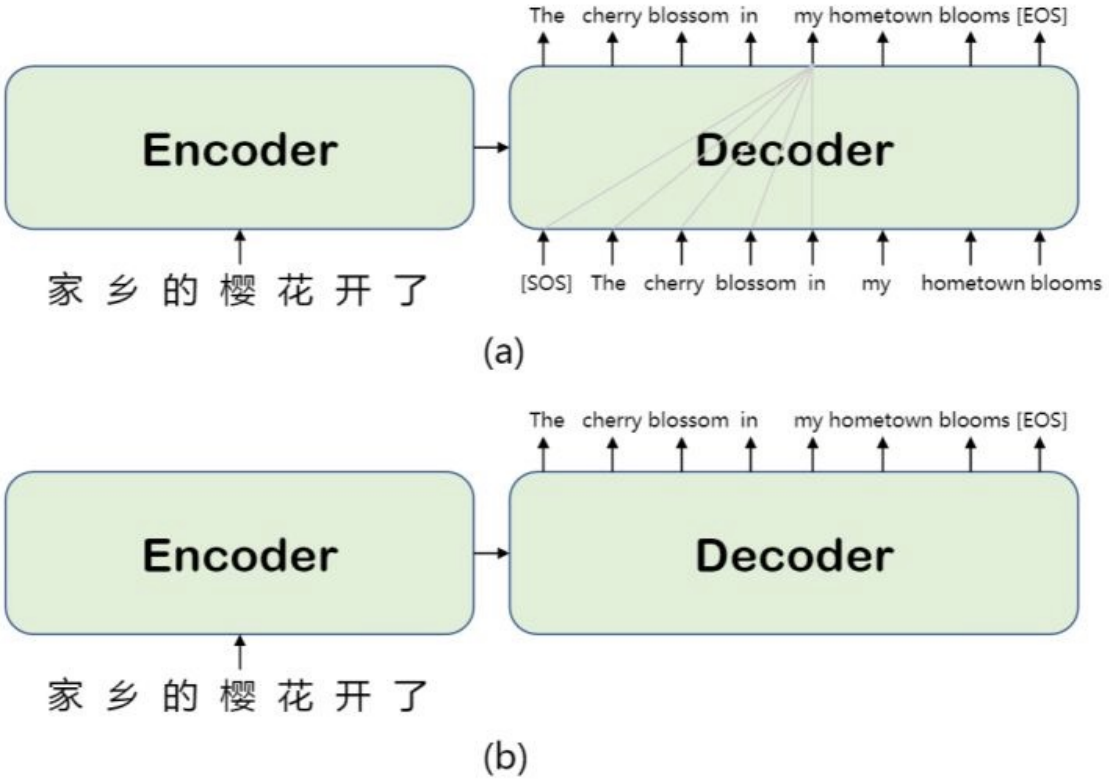
- GPT-2: Language models are unsupervised multitask learners
- Train the language model with unlabeled data, then fine-tune the model with labeled data according to corresponding tasks





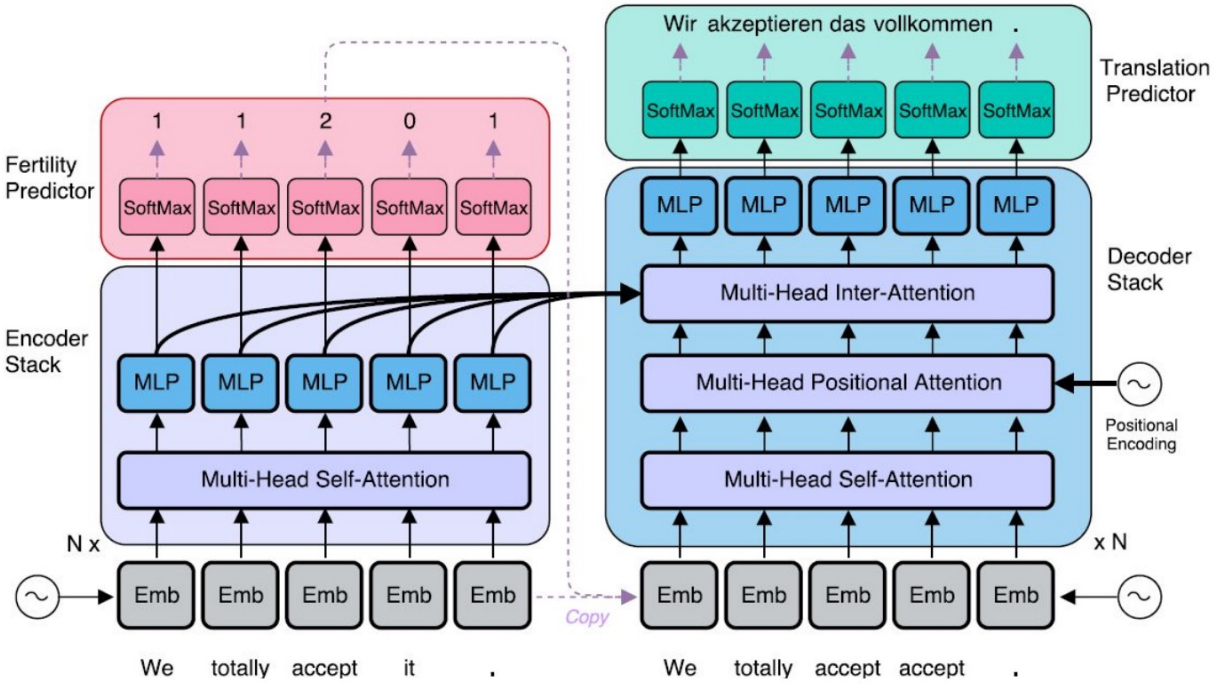
Non-Autoregressive Generation

- Given a source $x = (x_1, x_2, \ldots, x_n)$, target y $= (y_1, y_2, \ldots, y_m)$
- Generate in parallel: $P(y|x) = P(m \mid x) \prod_{t=1}^{m} P(y_t|z, x)$



Non-Autoregressive Generation

- $P(y|x) = P(m \mid x) \prod_{t=1}^{m} P(y_t|z, x)$
- $P(m \mid x)$ decide the length of the target sequence
- $z = f(x; \theta_{enc})$ captures dependencies among output tokens
- Generate target sequence in parallel



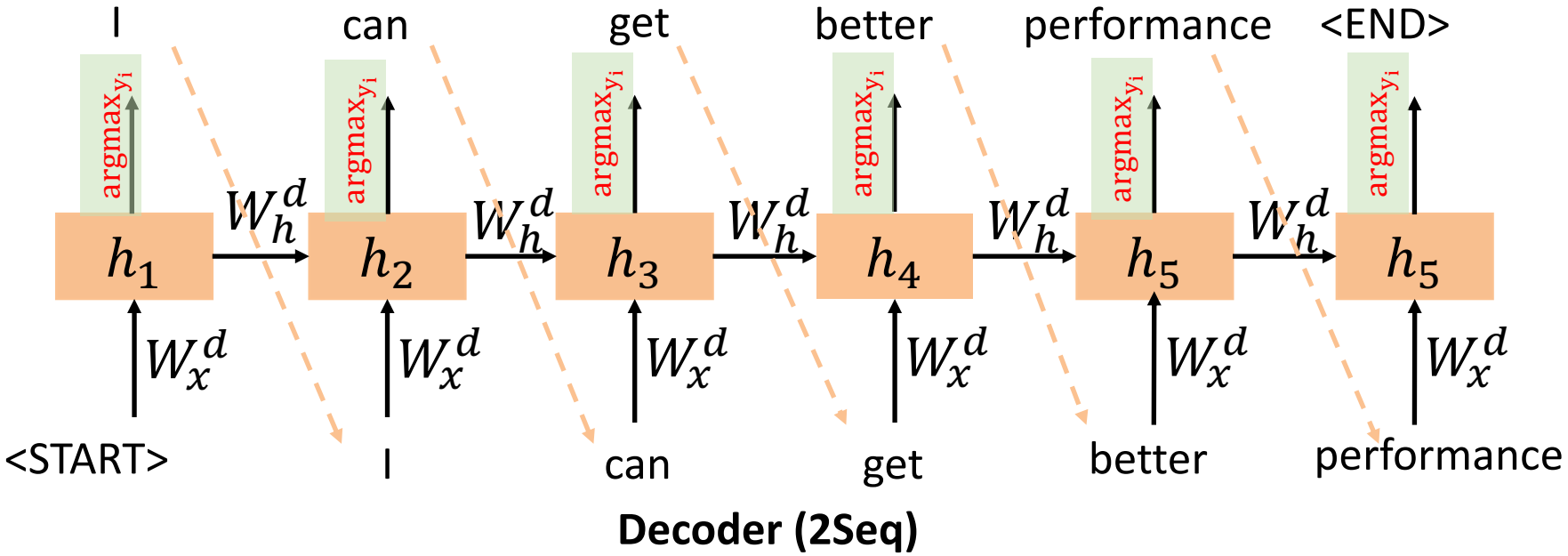
[Gu+ at el., 2018, Non-autoregressive neural machine translation]

Decoding

- Greedy decoding
- Beam search
- Sampling methods
  - Pure sampling
  - Top-n sampling
  - Nucleus sampling

Greedy Decoding

- Generate the target sentence by taking argmax on each step of the decoder
  - $\text{argmax}_{y_i} P(y_i | y_1, \ldots, y_{i-1}, x)$



Decoder (2Seq)

- Due to lack of backtracking, output can be poor (e.g. ungrammatical, unnatural, nonsensical)

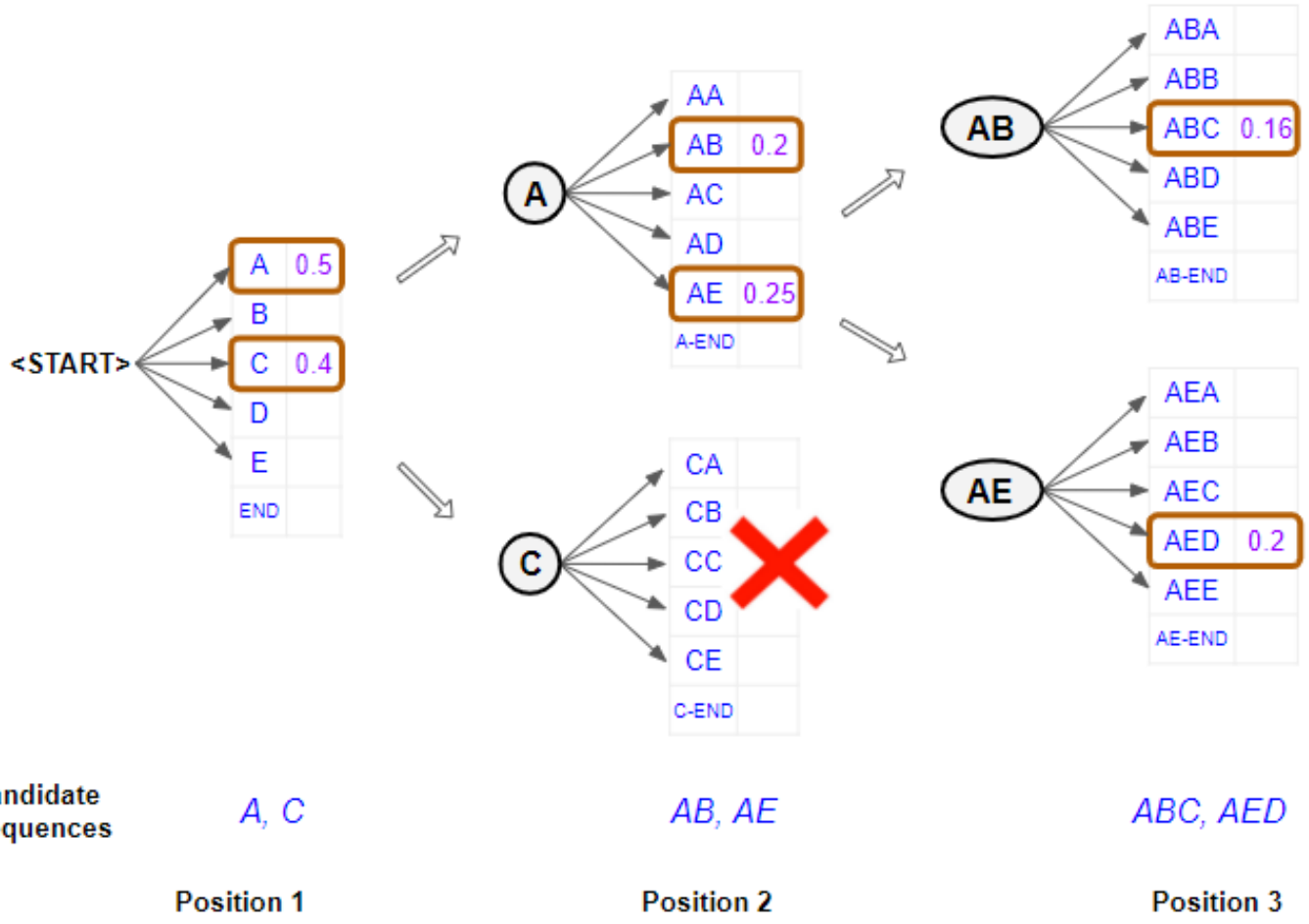
Beam Search Decoding

- We want to find y that maximizes
  - $P(y|x) = P(y_1|x)P(y_2|y_1, x) \ldots P(y_T|y_1, \ldots, y_{T-1}, x)$
  - Find a high-probability sequence

- Beam search
  - On each step of decoder, keep track of the k most probable partial sequences
  - After you reach some stopping criterion, choose the sequence with the highest probability
  - Not necessarily the optimal sequence

Beam Search Decoding

- Example (beam size = 2)



https://towardsdatascience.com/foundations-of-nlp-explained-visually-beam-search-how-it-works-1586b9849a24

Beam Search Decoding

- What's the effect of changing beam size k?
  - Small k has similar problems to greedy decoding
    - Ungrammatical, unnatural, nonsensical, incorrect
  - Larger k means you consider more hypotheses
    - Reduces some of the problems above
    - More computationally expensive
  - But increasing k can introduce other problems
    - For neural machine translation (NMT): Increasing k too much decreases BLEU score (Tu et al., Koehn et al.)
    - chit-chat dialogue: Large k can make output more generic

[Tu et al., 2017, Neural Machine Translation with Reconstruction]
[Koehn et al., 2017, Six Challenges for Neural Machine Translation]

Beam Search Decoding

- Effect of beam size in chitchat dialogue

I mostly eat a fresh and raw diet, so I save on groceries

Human chit-chat partner

Beam size	Model response
1	*I love to eat healthy and eat healthy*
2	*That is a good thing to have*
3	*I am a nurse so I do not eat raw food*
4	*I am a nurse so I am a nurse*
5	*Do you have any hobbies?*
6	*What do you do for a living?*
7	*What do you do for a living?*
8	*What do you do for a living?*

Low beam size:
More on-topic but nonsensical; bad English

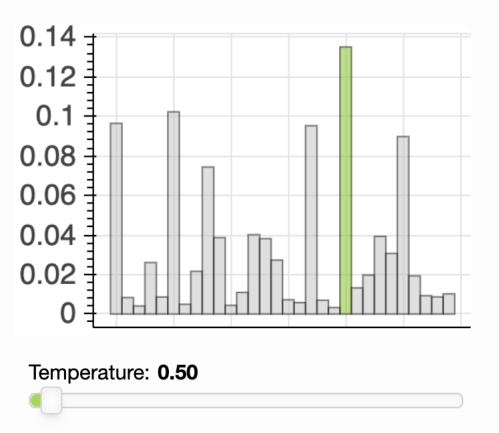
High beam size:
Converges to safe, "correct" response, but it's generic and less relevant

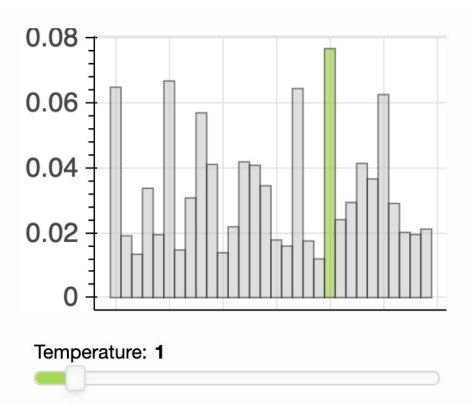
Sampling-based Decoding

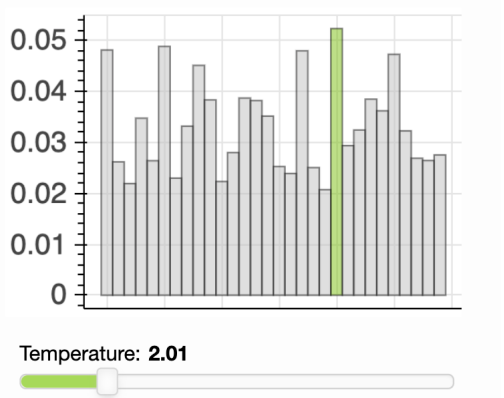
- Sampling-based decoding
  - Pure sampling
    - On each step t, randomly sample from the probability distribution P_t to obtain your next word
  - Top-n sampling
    - On each step t, randomly sample from P_t, restricted to just the top-n most probable words
    - $n = 1$ is greedy search, $n = V$ is pure sampling
  - Nucleus sampling (Top-p sampling)
    - On each step t, randomly sample from P_t, restricted to the top words that cover probability ≥ p
    - $p = 1$ is pure sampling

Sampling-based Decoding

- Sampling-based decoding
  - Sample with temperature
    - Before applying the final softmax, its inputs are divided by the temperature τ







- Increase n/p/temperature to get more diverse/risky output
- Decrease n/p/temperature to get more generic/safe output
- Both of these are more efficient than Beam search

https://lena-voita.github.io/nlp_course/language_modeling.html#generation_strategies

Decoding

- In summary
  - Greedy decoding
    - A simple method
    - Gives low quality output
  - Beam search
    - Delivers better quality than greedy
    - If beam size is too high, it will return unsuitable output (e.g. Generic, short)
  - Sampling methods
    - Get more diversity and randomness
    - Good for open-ended / creative generation (poetry, stories)
    - Top-n/p/temperature sampling allows you to control diversity

Neural text generation

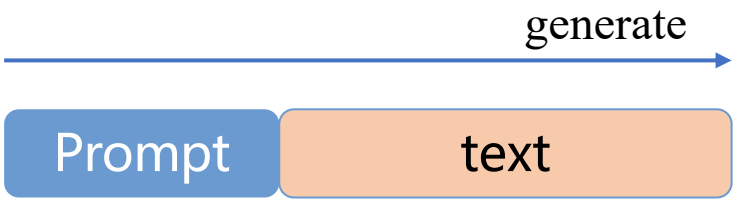
- In summary
  - Language modeling
    - $P(y_t|y_1, y_2, \dots, y_{t-1})$
  - Sequence to sequence
    - $P(y|x) = P(y_1|x)P(y_2|y_1, x) \dots P(y_T|y_1, \dots, y_{T-1}, x)$
  - Autoregressive models
    - $P(y|x) = \prod_{t=1}^{m} P(y_t|y_{<t}, x, \theta_{enc}, \theta_{dec})$
  - Non-autoregressive models
    - $P(y|x) = P(m \mid x) \prod_{t=1}^{m} P(y_t|z, x)$
  - Decoding strategy

Outline

- Introduction to text generation
- Tasks of text generation
- Neural text generation
- Controllable text generation
  - Prompt methods
  - Modifying probability distribution
  - Reconstructing model architecture
- Text Generation Evaluation
- Challenges

Control Text Generation

- Control text generation: avoid repeating, more diverse, …
- Prompt methods
- Modifying probability distribution
- Reconstructing model architecture



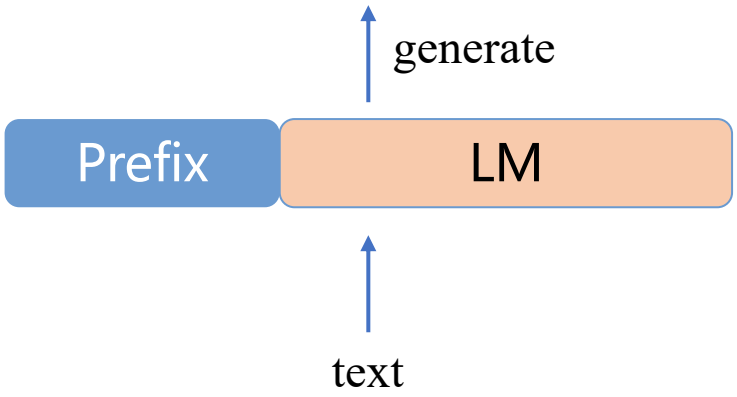
Horror *A knife* handle pulled through the open hole in the front. I jumped when the knife hit.\n\nEyes widened in horror. Her scream was the only sound I heard besides her sobs.\n\nThe spider touched her feet as it started to dig into the top of her arch. The creature's tears began to flow.\n\nThe spider looked up to her and looked back at me with eyes filled with burning tears. My heart started to race. . .

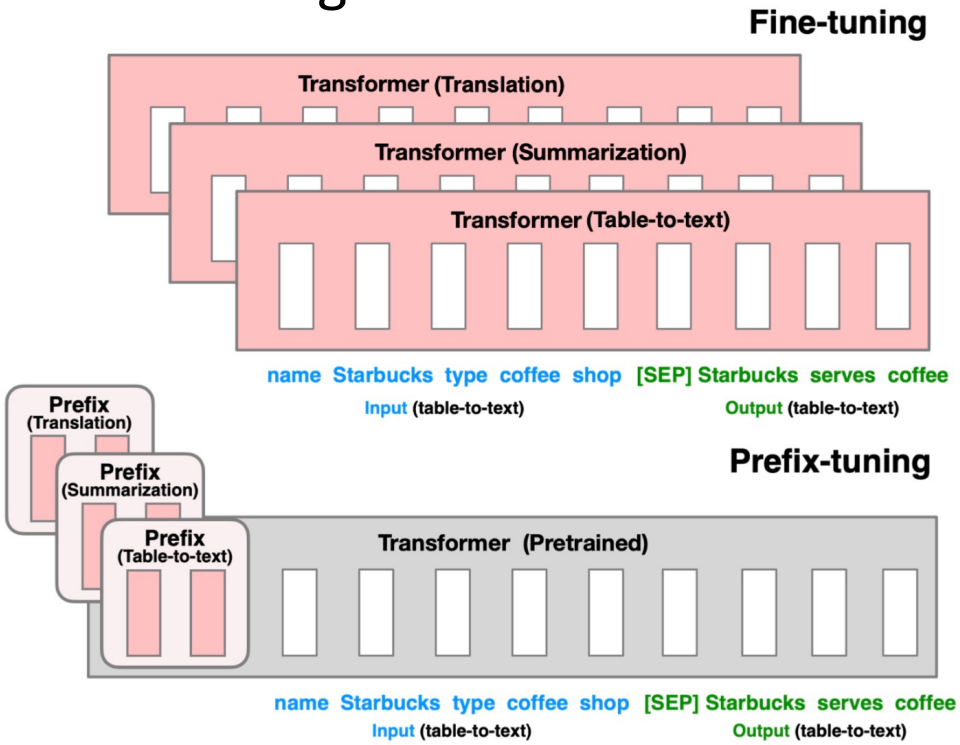
Reviews *A knife* is a tool and this one does the job well.\n\nRating: 4.0\n\nI bought these for my husband who has been using them to cut up his own meat since he got them. He says they are very sharp so be careful when you use them, but that doesn't seem like much of an issue because he's used it on everything from chicken breasts to beef tenderloin. . .

Keskar, 2019, Ctrl: A conditional transformer language model for controllable generation

Control Text Generation

- Control text generation: avoid repeating, more diverse, …
- Prompt methods
- Modifying probability distribution
- Reconstructing model architecture





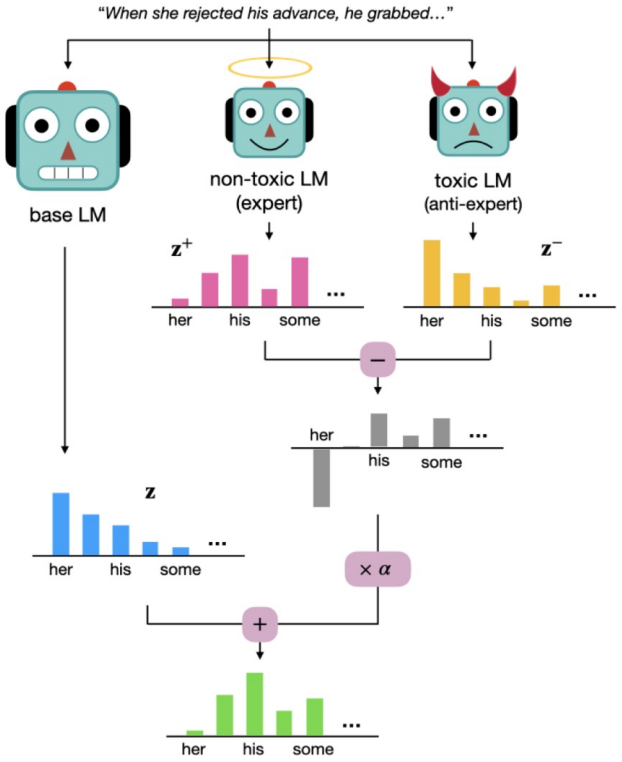
Li, 2021, Prefix-tuning: Optimizing continuous prompts for generation

Control Text Generation

- Control text generation: avoid repeating, more diverse, …
- Prompt methods
- Modifying probability distribution
- Reconstructing model architecture

$$\tilde{P}(X_t \mid \boldsymbol{x}_{<t}) = \text{softmax}\left(\mathbf{z}_t + \alpha\left(\mathbf{z}_t^+ - \mathbf{z}_t^-\right)\right)$$

Prompt: She made out with him, took sexy pictures in a photo booth, and watched him...	
GPT-2	having sex with a still-pregnant woman in the bathroom. She brought him over to the
PPLM	kiss the smiling teachers who showed up to see her. ↳ Her is the story of Kimberly Tur
DAPT	do it — with his brothers, neighbors, female employees at the bank and other bank customers who'd
GeDi	admire her enormous breasts for the first time. She laughed when she found out Bruce kept them in to
DEXPERTS ($\alpha = 1.0$)	go about his daily chores like a good girlfriend should. A neat piece of work who wanted her to
DEXPERTS ($\alpha = 2.0$)	play drums on the rooftop patio with his band while another time his girlfriend came over and woke him up



Liu, et al. 2021, DEXPERTS: Decoding-Time Controlled Text Generation with Experts and Anti-Experts.

Control Text Generation

- Control text generation: avoid repeating, more diverse, …
- Prompt methods
- Modifying probability distribution
- Reconstructing model architecture

Specialized encoder for guidance signal

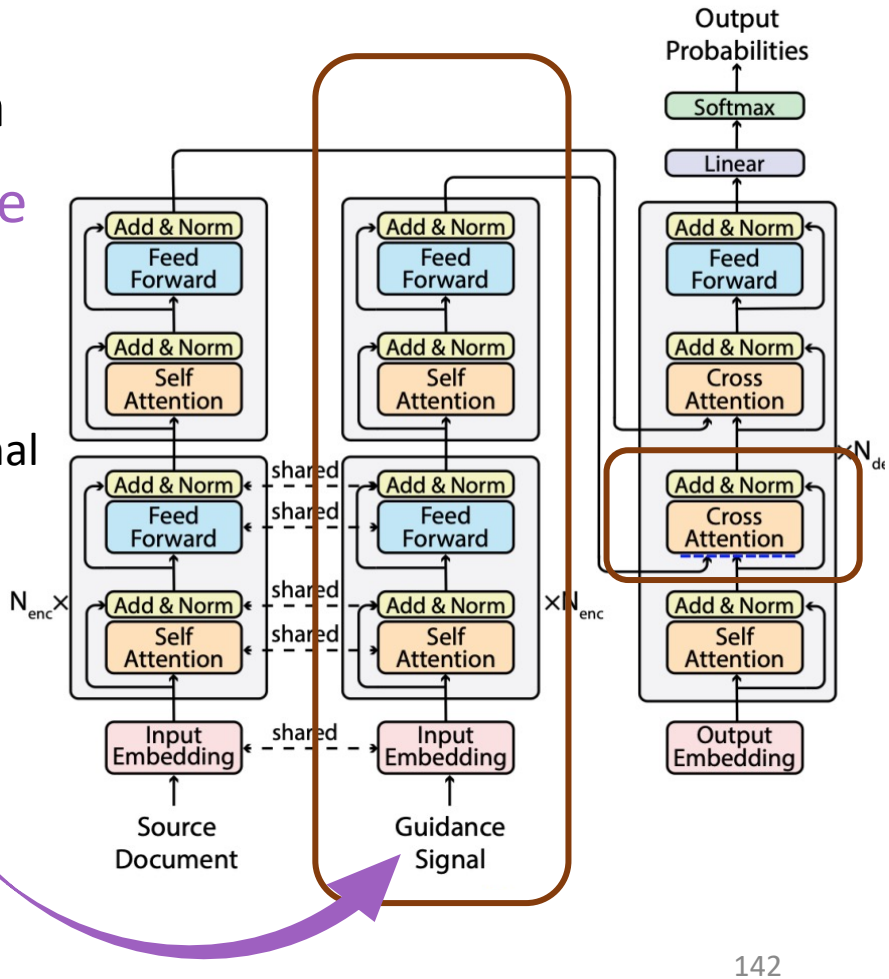
- Decoder:

  self-attention ->

  (+guidance signal)cross-attention ->

  (+source document)cross-attention ->

  FFN



Dou, Zi-Yi, et al. "Gsum: A general framework for guided neural abstractive summarization." arXiv preprint arXiv:2010.08014 (2020).

Outline

- Introduction to text generation
- Tasks of text generation
- Neural text generation
- Controllable text generation
- Text generation evaluation
  - Overlap-based Metric
  - Other Metric
- Challenges

Overlap-based Metric

- Common metrics
  - BLEU (Bilingual evaluation understudy)
    - easy to compute
    - doesn't consider semantics & sentence structure

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N} W_n \times \log P_n\right), \qquad BP = \begin{cases} 1 & lc > lr \\ \exp\left(1 - \frac{lr}{lc}\right). & lc \leq lr \end{cases}$$

  - PPL (perplexity)
    - Evaluate how well a probability model predicts a sample.

$$perplexity(S) = p(w_1, w_2, w_3, \ldots, w_m)^{-1/m}$$

$$= \sqrt[m]{\prod_{i=1}^{m} \frac{1}{p(w_i | w_1, w_2, \ldots, w_{i-1})}}$$

Overlap-based Metric

- Common metrics for Translation & Summarization
  - ROUGE (Recall-Oriented Understudy for Gisting Evaluation)
    - solve the problem of missed flipping (low recall rate)

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_N \in S} Count_{match}(gram_N)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_N \in S} Count(gram_N)}$$

  - NIST
    - consider the amount of n-gram information
  - METEOR
    - based on the harmonic mean of precision and recall

Other Metrics

- Distance-based Metrics
  - Edit Dist (cosine similarity); SMD (embedding distance); YISI (weighted similarity)
- Diversity Metrics
  - Distinct (n-gram diversity); Entropy; KL_divergence
- Task-oriented Metrics
  - SPICE (Semantic propositional image caption evaluation)
- Human Evaluation
  - Intrinsic (fluency, internal relevance, correctness)
  - Extrinsic (performance on the downstream subtasks)

Outline

- Introduction to text generation
- Tasks of text generation
- Neural text generation
- Controllable text generation
- Text generation evaluation
- Challenges

TG Tasks: Challenges

- Training & model strategy
  - Always generate repeated words
  - Exposure bias
- Commonsense
  - Lack of logical consistency
- Controllability
  - Difficult to ensure both language quality and control quality
- Evaluation: Reasonable metrics and datasets

Demo: GPT-2

- WebNLG dataset
- Task
  - The WebNLG challenge consists in mapping data to text
  - The training data consists of Data/Text pairs where the data is a set of triples extracted from DBpedia and the text is a verbalization of these triples.
  - Example:
    - a. (John_E_Blaha birthDate 1942_08_26) (John_E_Blaha birthPlace San_Antonio) (John_E_Blaha occupation Fighter_pilot)
      b. *John E Blaha, born in San Antonio on 1942-08-26, worked as a fighter pilot*

https://webnlg-challenge.loria.fr/challenge_2017/

Demo: GPT-2

- Text generated with untuned GPT-2

```
input:   | Abilene_Regional_Airport : cityServed : Abilene,_Texas
target: Abilene, Texas is served by the Abilene regional airport.
Abilene Regional Airport serves the city of Abilene in Texas.
generated: <pad> To <pad> To <pad> To <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad>
<pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad> <pad>
```

- Loss

```
Epoch 0, global_step 20 average loss: 11.240423202514648 lr: 4.4871794871794874e-05
Epoch 0, global_step 40 average loss: 2.357112014770508 lr: 3.974358974358974e-05
Epoch 0, global_step 60 average loss: 1.3735915184020997 lr: 3.461538461538462e-05
Epoch 1, global_step 80 average loss: 1.0153145580291747 lr: 2.948717948717949e-05
Epoch 1, global_step 100 average loss: 0.7632075481414795 lr: 2.435897435897436e-05
Epoch 1, global_step 120 average loss: 0.658911714553833 lr: 1.923076923076923e-05
Epoch 2, global_step 140 average loss: 0.7315932235717774 lr: 1.4102564102564104e-05
Epoch 2, global_step 160 average loss: 0.5259016437530517 lr: 8.974358974358976e-06
Epoch 2, global_step 180 average loss: 0.5857448959350586 lr: 3.846153846153847e-06
```

- Text generated with tuned GPT-2

```
input:   | Abilene_Regional_Airport : cityServed : Abilene,_Texas
target: Abilene, Texas is served by the Abilene regional airport.
Abilene Regional Airport serves the city of Abilene in Texas.
generated: Abilene Regional Airport serves Abilene, Texas.
```