Tencent AI Lab

Recent Advances in Retrieval-Augmented Text Generation

Deng Cai (蔡登)
The Chinese University of Hong Kong

Yan Wang (王琰)
Tencent AI Lab

Lemao Liu (刘乐茂)
Tencent AI Lab

Shuming Shi (史树明)
Tencent AI Lab

What is This Tutorial About?

- Integrating Information Retrieval (IR) Techniques in Text Generation

Information Retrieval + **Text Generation** = **Retrieval-Augmented Text Generation**

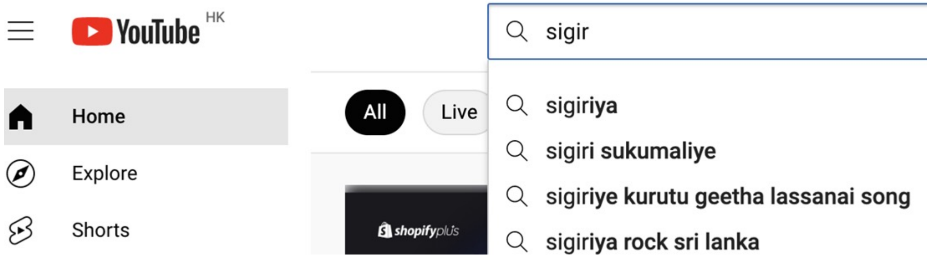
Close-book exam
(Hard mode)

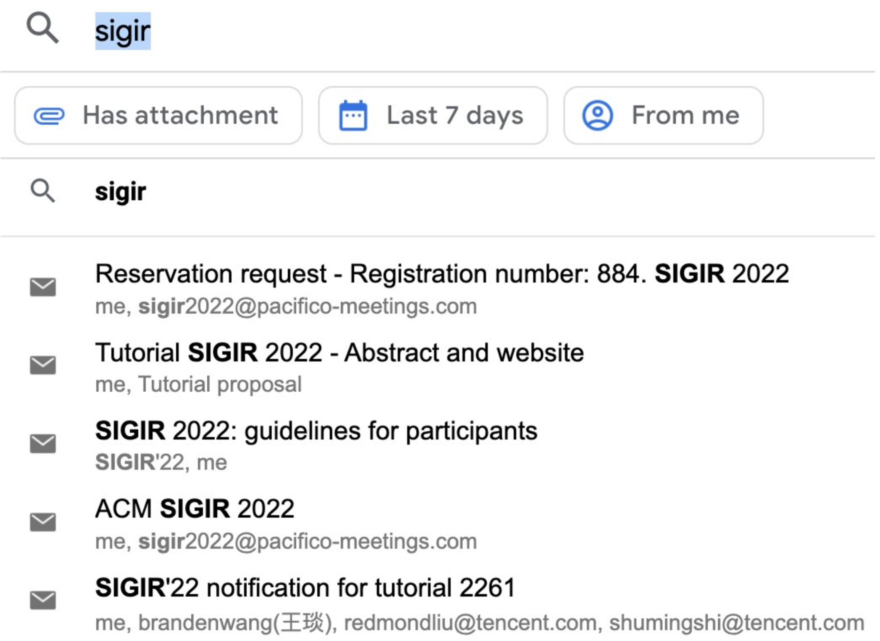
Open-book exam
(Easy mode)

Information Retrieval

- Information Retrieval (IR) is finding material of an unstructured nature (usually text) that satisfies an information need from large collections

- Web Search
- Video Search
- E-mail Search

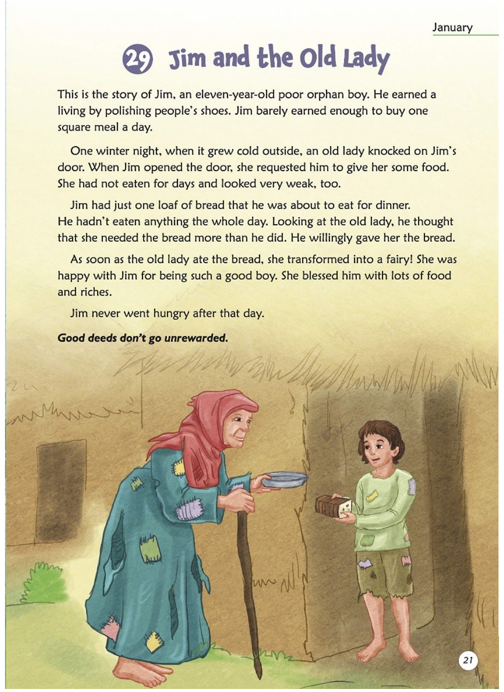




Text Generation

- Text generation, also known as natural language generation, is the task of generating text with the goal of appearing indistinguishable to human-written text

- Story Generation
- Dialogue Generation
- Machine Translation



January

29 Jim and the Old Lady

This is the story of Jim, an eleven-year-old poor orphan boy. He earned a living by polishing people's shoes. Jim barely earned enough to buy one square meal a day.

One winter night, when it grew cold outside, an old lady knocked on Jim's door. When Jim opened the door, she requested him to give her some food. She had not eaten for days and looked very weak, too.

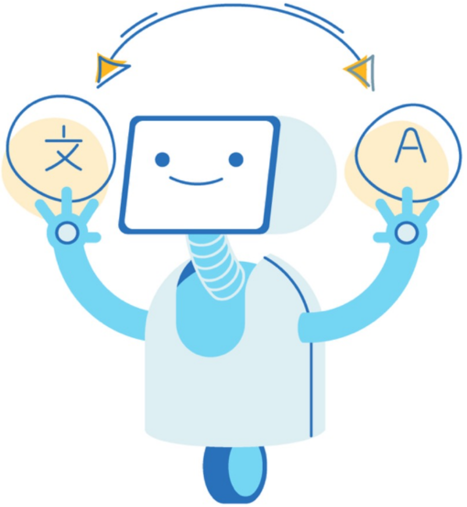
Jim had just one loaf of bread that he was about to eat for dinner. He hadn't eaten anything the whole day. Looking at the old lady, he thought that she needed the bread more than he did. He willingly gave her the bread.

As soon as the old lady ate the bread, she transformed into a fairy! She was happy with Jim for being such a good boy. She blessed him with lots of food and riches.

Jim never went hungry after that day.

Good deeds don't go unrewarded.

21



The Challenge

Tencent AI Lab

- Create is more difficult than judge!

Binary Classification

SIGIR 2022 will be held on July?

True

False

Multi-Class Classification

When will SIGIR 2022 be held?

June

July

August

September

Text Generation

Write about following topic

SIGIR 2022 will be held at Madrid, Spain. What do you think about this conference? Will you attend this conference?

Write at least 250 words.

Require strong background information about SIGIR 2022!

The information

- Where are these information?
  - In Training data
- How do we store these information
  - In Model parameters
  - This is why more data + bigger model always better in generation tasks
- Any alternative ways?
  - Endow model the capability to re-access its training data, or external resources

Close-book exam
(Hard mode)

Open-book exam
(Easy mode)

The Open-Book Paradigm

- **Core Questions**
  - Which book shall we open?**(Retrieval Sources)**
  - How to find needed information from the books? **(Retrieval Methods)**
  - How to use the found information? **(Integrating IR Results in Generation)**

The Open-Book Paradigm

- Which book shall we open?**(Retrieval Sources)**
  - Training Examples: re-access the examples we have already seen
  - External Examples:
    - Allow models accessing unseen examples
    - Beneficial for efficient domain adaptation and knowledge update
  - Unlabeled Data:
    - Retrieving any necessary knowledge from unlabeled corpus
    - Prevalent in Language Modeling and Question Answering

The Open-Book Paradigm

- How to find needed information from the books? **(Retrieval Methods)**
  - Sparse-Vector Retrieval
    - TF-IDF, BM25: Based on lexical-level similarity
    - Computed efficiently with an inverted index
  - Dense-Vector Retrieval
    - Embedding sentences in dense vectors via BERT-based encoders
    - computed via Maximum Inner Product Search (MIPS)
  - Task-Specific Retrieval
    - Intuition: **Nearest != Best**
    - Who is the best? End-to-End optimized in generation tasks

The Open-Book Paradigm

- How to use the found information? **(Integrating IR Results in Generation)**
  - Input Augmentation
    - Concatenating Retrieval samples with the original input
    - Simple, but do not support long text
  - Attention Mechanisms
    - Encoding memory via additional encoders, and integrate through cross-attention
  - Explicit Skeleton & Prototype
    - Intuition: remove the worthless and preserve the valuable

Successful Applications

- Language Modeling
- Open-Domain Dialogue Generation
- Machine Translation
- Question Answering
- Summarization
- Paraphrase Generation
- Text Style Transfer
- Data-to-Text Generation
- Image Caption
- Code Generation
- …

Outline

Language Modeling (45 Min)

Yan Wang (王琰)

Tencent AI Lab

Dialogue Generation (45 Min)

Deng Cai (蔡登)

The Chinese University of Hong Kong

Machine Translation (45 Min) + Conclusion (10 Min)

Lemao Liu (刘乐茂)

Tencent AI Lab

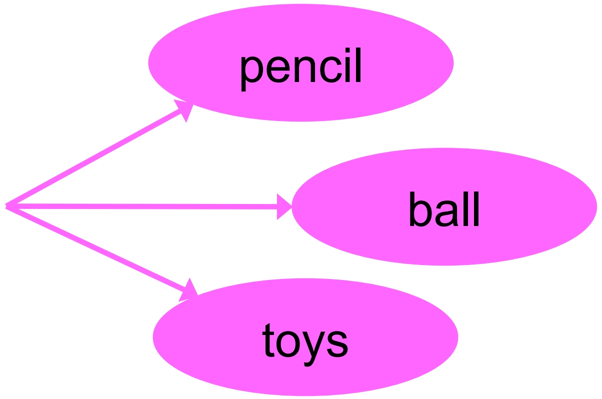
WARNING: this is a new research area, conclusions in this tutorial may be out-of-date soon!

Outline

Language Modeling

- Language Modeling is a fundamental NLP task that predicting what word comes next

A boy is looking at his ______



- Formally: given a sequence of words $x^1, x^2, \dots, x^t$, compute the probability distribution of the next word x^{t+1}:

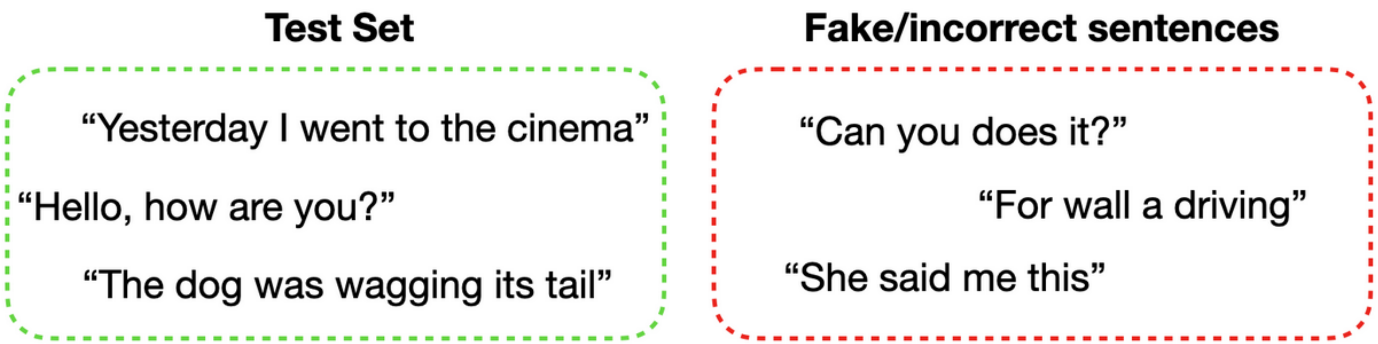
$$P(x^{t+1} | x^1, \dots, x^t)$$

Where x^{t+1} can be any word in the vocabulary $V = \{w_1, \dots, w_{|V|}\}$

- A system that does this is called a Language Model (LM)

Evaluation of Language Modeling

- Perplexity: an intrinsic evaluation method for LM
- Intuition: The probability of correct text (test set) should be high



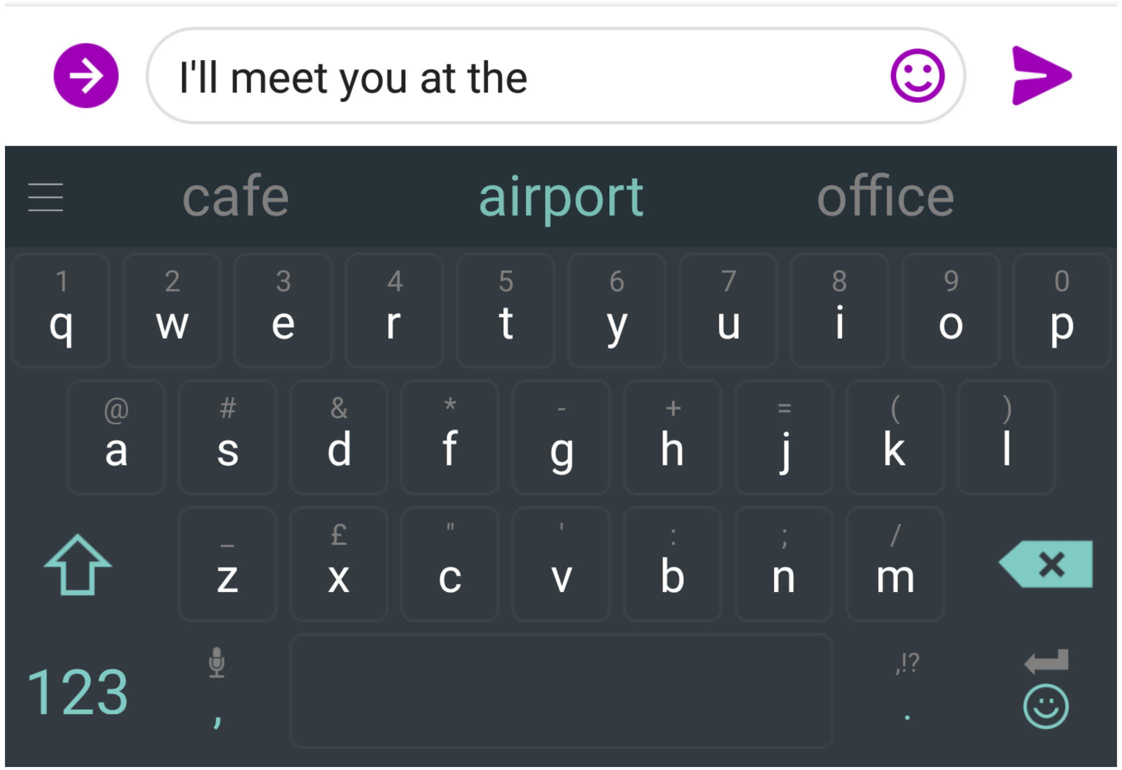
High probability
Low perplexity

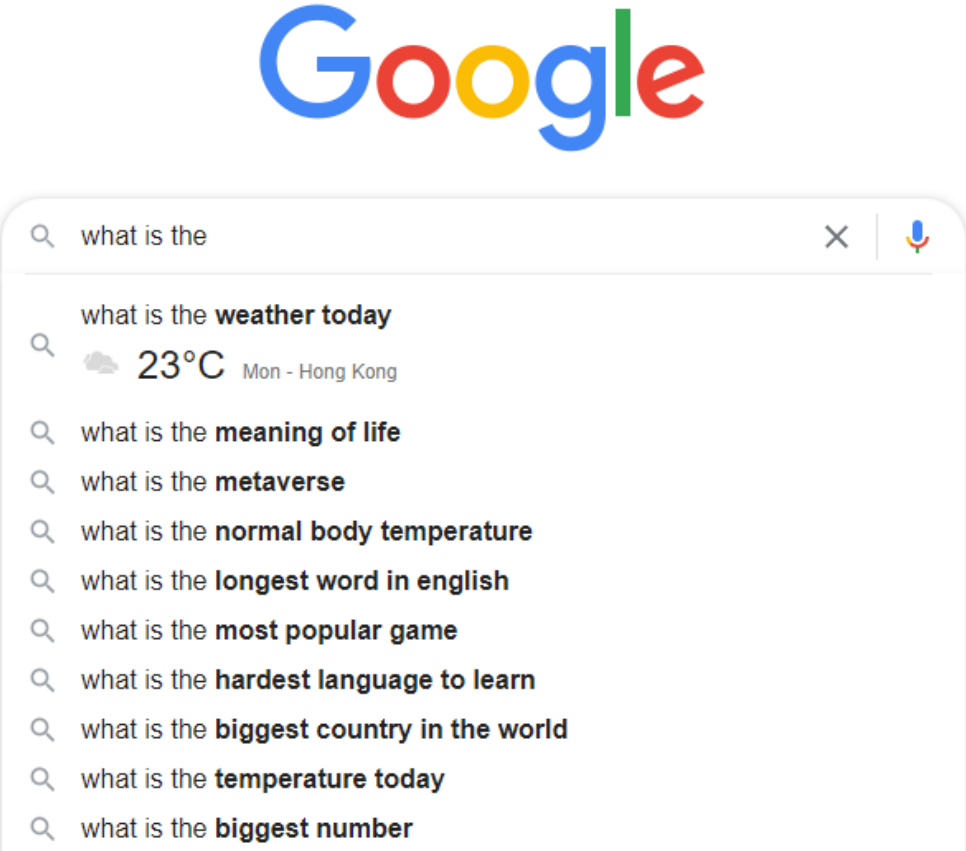
Low probability
High perplexity

- Formal definition:

$$PP(W) = \sqrt[N]{\frac{1}{P(w_1, w_2, \dots, w_N)}}$$

We use LM every day!





Traditional (Pre-Deep Learning) way: n-gram LM

A boy is looking at his ______

- N-gram Language Model
- Definition: A *n-gram* is a chunk of n consecutive words.
    - 1-gram: "a", "boy", "is", "looking", "at", "his"
    - 2-grams: "a boy", "boy is", "is looking", "looking at", "at his"
    - 3-grams:"a boy is", "boy is looking", "is looking at", "looking at his"
    - …
    - 6-grams: "a boy is looking at his "
- N-gram LM: Collect statistics about how frequent different n-grams are

$$P(x^{t+1}|x^t, \dots, x^1) = P(x^{t+1}|x^t, \dots, x^{t-n+2}) \approx \frac{count(x^{t+1}, x^t, \dots, x^{t-n+2})}{count(x^t, \dots, x^{t-n+2})}$$

Problems of n-gram LM

- Sparsity
  - Hard to compute the probability of unseen text
- Storage
  - Need to store count for all n-grams. Increasing n or corpus increases model size!

- Generating text with a 3-gram LM

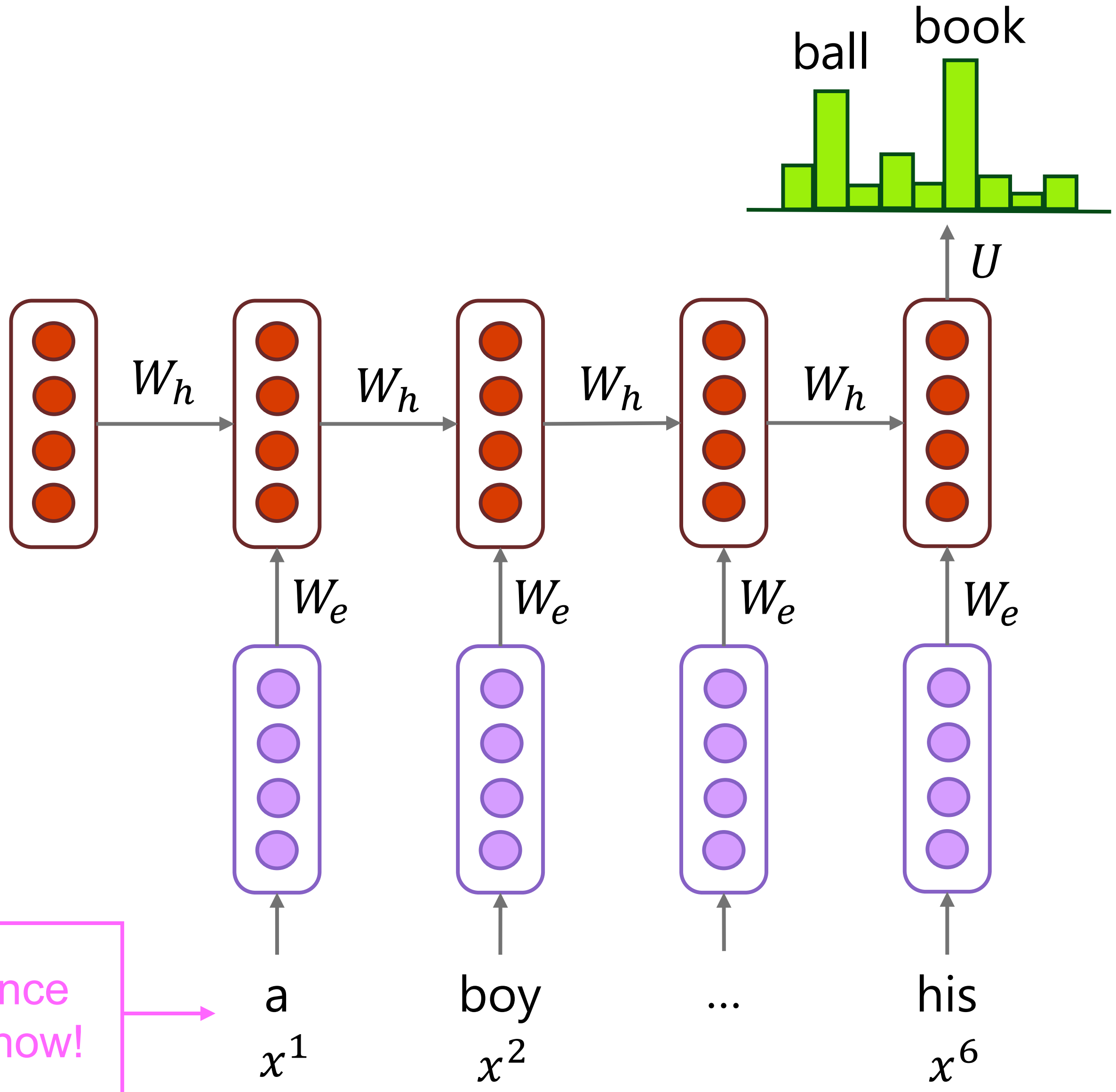
A boy is looking at his phone . A third possibility is that he was driving with his wife . I'm only thinking about my sexuality . The US wants the fight so he's starting to understand that no one could be expected to help get through a day .

Surprisingly grammatical!

...but incoherent. We need to consider longer context, but increasing n worsens sparsity problem, and increases model size

RNN Language Model

- Advantages:
  - Can process any length input
  - Theoretically, can consider very long context
  - Model size doesn't increase for longer input context
- Disadvantage:
  - Recurrent computation is slow
  - Difficult to access very long context in practice



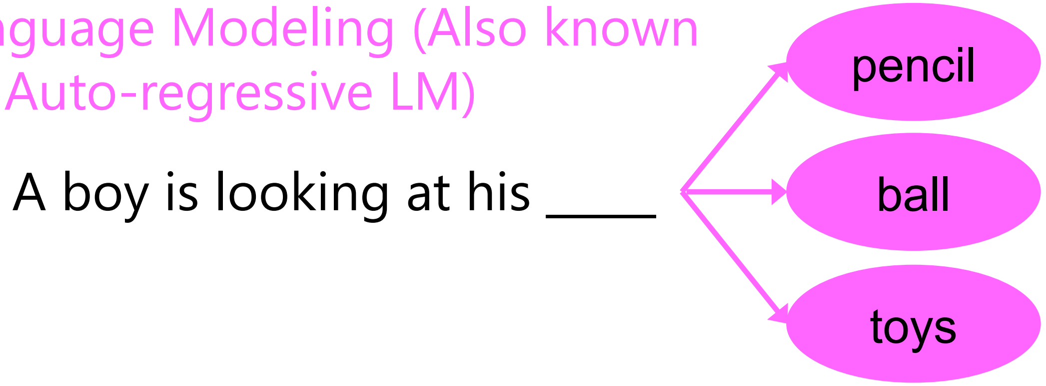
Note: this input sequence could be much longer now!

Pre-trained Language Model (PLM)

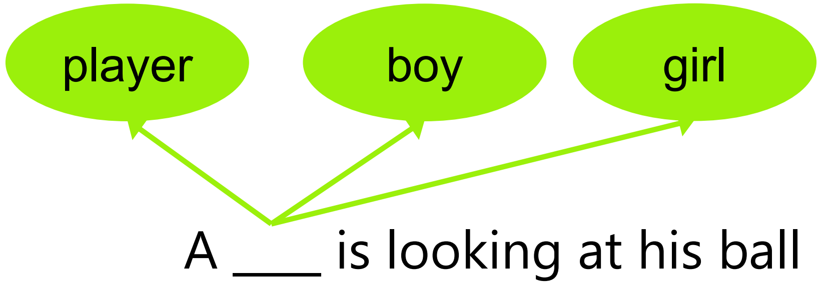
Tencent AI Lab

- Two pretraining objectives:

Language Modeling (Also known as Auto-regressive LM)



Masked Language Modeling



- Condition on the past only
- Representatives: GPT, GPT2, Retro
- It's helpful when the output is a sequence:
  - Dialogue (Condition on dialogue history)
  - Story Generation (Condition on story title)

- Condition on both the past and the future
- Representatives: BERT, and its variants
- It's helpful on Natural Language Understanding tasks
  - Sequence Labeling & Semantic Matching

PLM for Text Generation

- Open-Ended Text Generation: Fluent, informative, and coherent

Context (human-written): In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

GPT-2: The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

[Radford + 19]

Why So Good?

- Why so good?
  - Big: big model, big corpus
  - A way that teaches the model remembering knowledge in corpus

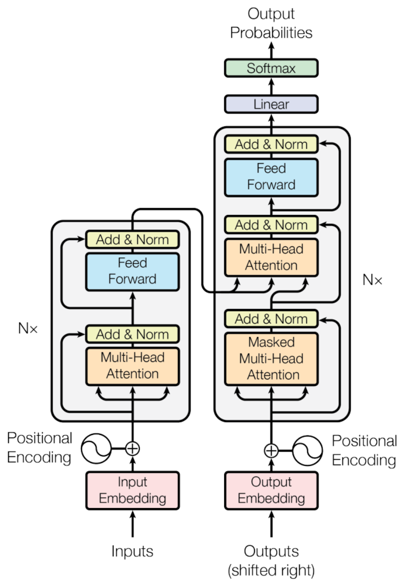
- What's bad?
  - Big->High cost on both time and space

Motivation of Retrieval-Augmented LM

Remember? This is the Expertise of IR

- Store knowledge in LM

Knowledge



- Store knowledge in non-parametric index

Knowledge

Full List of Retrieval-Augmented LM

- Interpolation-based LM
  - Improving neural language models with a continuous cache. ICLR 2017
  - Generalization through memorization: Nearest neighbor language models. ICLR 2020
  - Adaptive semiparametric language models. TACL 2021
- Masked LM and QA*
  - Dense passage retrieval for open-domain question answering. EMNLP 2020
  - Latent Retrieval for Weakly Supervised Open Domain Question Answering. ACL 2019
  - Retrieval augmented language model pre-training. ICML 2020
  - Retrieval-augmented generation for knowledge-intensive NLP tasks. NeuriPS 2020
  - Leveraging passage retrieval with generative models for open domain question answering. EACL 2021
- Huge-Index but Small-Size LM
  - Improving language models by retrieving from trillions of tokens. DeepMind 2022

*Retrieval-Augmented QA is not the core of this tutorial, one may refer to ACL tutorial "Knowledge-Augmented Methods for Natural Language Processing" for more details about this area

Full List of Retrieval-Augmented LM

- Interpolation-based LM
  - Improving neural language models with a continuous cache. ICLR 2017
  - Generalization through memorization: Nearest neighbor language models. ICLR 2020 ★
  - Adaptive semiparametric language models. TACL 2021
- Masked LM and QA*
  - Dense passage retrieval for open-domain question answering. EMNLP 2020
  - Latent Retrieval for Weakly Supervised Open Domain Question Answering. ACL 2019
  - Retrieval augmented language model pre-training. ICML 2020 ★
  - Retrieval-augmented generation for knowledge-intensive NLP tasks. NeuriPS 2020
  - Leveraging passage retrieval with generative models for open domain question answering. EACL 2021
- Huge-Index but Small-Size LM
  - Improving language models by retrieving from trillions of tokens. DeepMind 2022 ★

*Retrieval-Augmented QA is not the core of this tutorial, one may refer to ACL tutorial "Knowledge-Augmented Methods for Natural Language Processing" for more details about this area

Interpolation-based Method: KNN-LM

Tencent AI Lab

Generalization through Memorization: Nearest Neighbor Language Models

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, Mike Lewis
Stanford University, Facebook AI Research

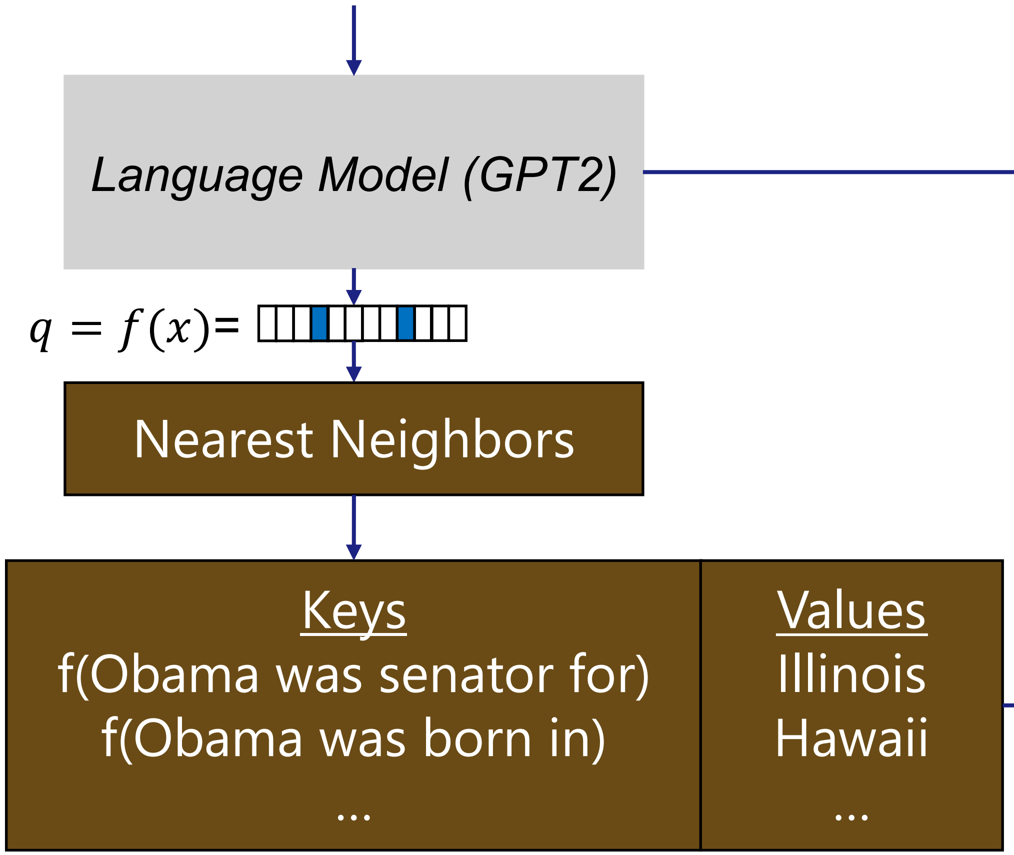
Stanford | NLP

facebook Artificial Intelligence

KNN-LM: Intuition

Tencent AI Lab

x = Obama's birthplace is ____



P_{LM} on vocabulary	
Hawaii	0.2
Illinois	0.2
…	…

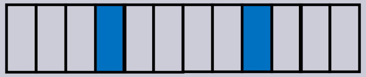
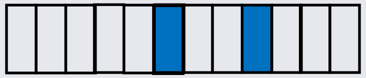
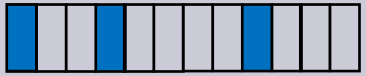
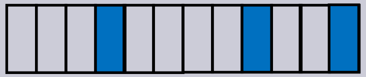
$$(1-\lambda)P_{LM}+\lambda P_{KNN}$$

P_{KNN} on vocabulary	
Hawaii	0.6
Illinois	0.2
…	…

Constructing the Index

Training Contexts c_i	Targets v_i
Obama was senator for	Illinois
Barack is married to	Michelle
Obama was born in	Hawaii
…	…
Obama is a native of	Hawaii

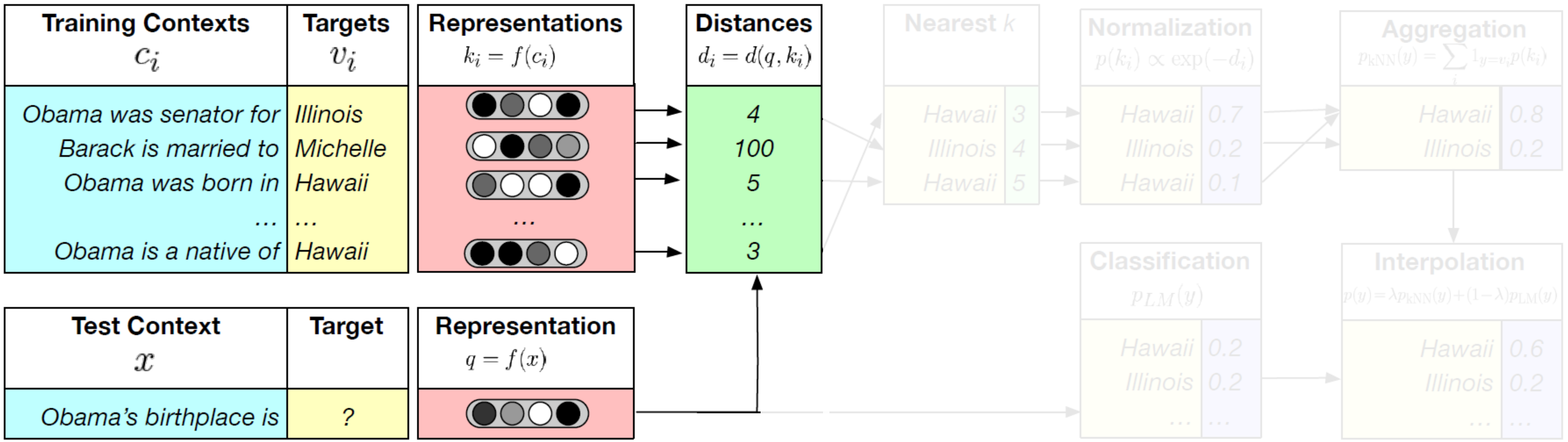
Constructing the Index

Training Contexts c_i	Representations $\mathbf{c}_i = f(c_i)$	Targets v_i
Obama was senator for		Illinois
Barack is married to		Michelle
Obama was born in		Hawaii
…	…	…
Obama is a native of		Hawaii

The size of the datastore = The number of tokens in training corpus

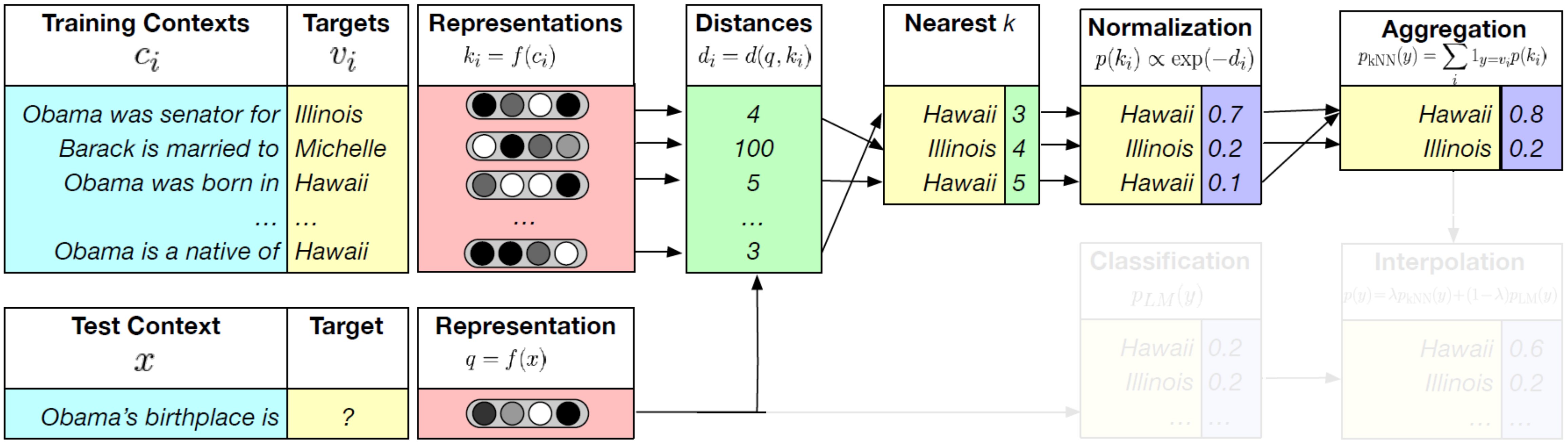
Retrieval nearest contexts to current context c

Back to Inference



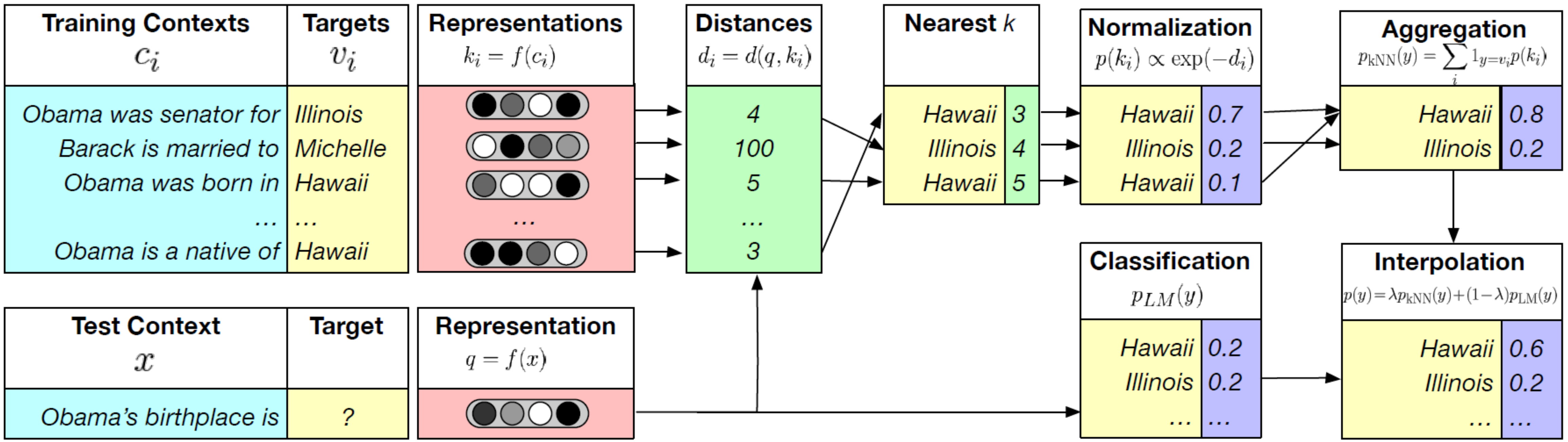
[Khandelwal+ 19]

Back to Inference



[Khandelwal+ 19]

Back to Inference



[Khandelwal+ 19]

Key Results

Tencent AI Lab

Explicitly memorizing the training data helps generation

LMs can scale to larger text collections without the added cost of training, by simply adding the data to the index

A single LM can adapt to multiple domains without the in-domain training, by adding domain-specific data to the index

Key Results

Tencent AI Lab

Memorizing with Wikitext-103: 103M tokens, $\lambda = 0.25$

Model	Perplexity↓
Previous Best (Luo et al., 2019)	17.40
Base LM	18.65
KNN-LM	16.12 ★
KNN-LM + Cont. Cache*	15.79 ★

*Edouard Grave, Armand Joulin, and Nicolas Usunier. Improving neural language models with a continuous cache. In ICLR, 2017

Key Results

Tencent AI Lab

Explicitly memorizing the training data helps generation

LMs can scale to larger text collections without the added cost of training, by simply adding the data to the index

A single LM can adapt to multiple domains without the in-domain training, by adding domain-specific data to the index

Key Results

From Wikitext-103 (100M tokens) to En-Wiki (3B tokens)

LM Training Data	Index	Perplexity↓
En-Wiki-3B	-	15.17
Wiki-100M	-	19.59
Wiki-100M	En-Wiki	13.73

Retrieving from corpus VS training on corpus

Key Results

Explicitly memorizing the training data helps generation

LMs can scale to larger text collections without the added cost of training, by simply adding the data to the index

A single LM can adapt to multiple domains without the in-domain training, by adding domain-specific data to the index

Key Results

Tencent AI Lab

Domain Adaptation from Wiki to Books

LM Training Data	Index	Perplexity↓
Books	-	11.89
Wiki-3B	-	34.84
Wiki-3B	Books	20.47

Domain adaptation in a plug-and-play manner!

Summary

Explicitly memorizing the training data helps generation

LMs can scale to larger text collections without the added cost of training, by simply adding the data to the index

A single LM can adapt to multiple domains without the in-domain training, by adding domain-specific data to the index

Limitations of KNN-LM

High index cost: Index size = Token number!

High inference cost: times of retrieval = generation length

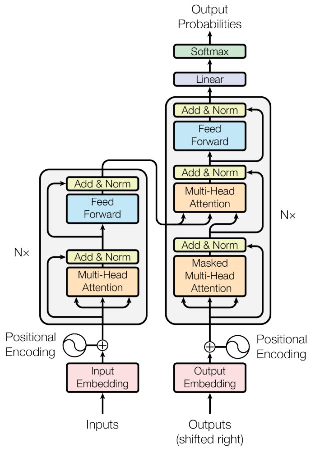
Gap between training and inference: No retrieval in training

Retrieval-Augmented MLM Pretraining

Introducing Explicit World Knowledge

Typical encoder: $p(y|x)$

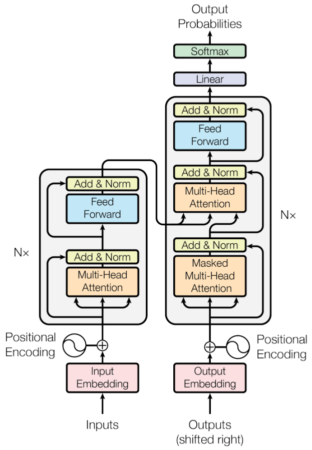
y= pounds



x: we paid 20 __ at the Buckingham Palace gift shop

Knowledge-augmented encoder: $p(y|x,z)$

y= pounds



x: we paid 20 __ at the Buckingham Palace gift shop

Linguistic knowledge

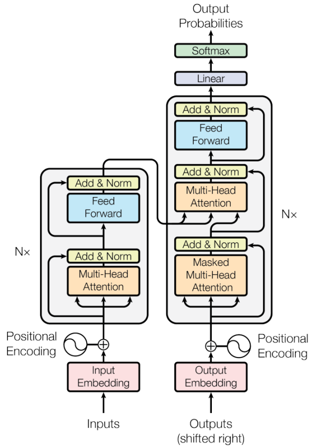
z: Buckingham Palace is home to the British monarchy

World knowledge

Problem: How to Select Right Knowledge

Knowledge-augmented encoder: $p(y|x,z)$

y= pounds



x: we paid 20 __ at the Buckingham Palace gift shop

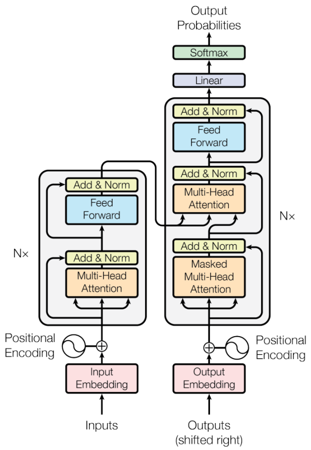
z: ???

No golden labels

Solution: try different documents

High

$$p(y = 'pounds' | x, z_1)$$

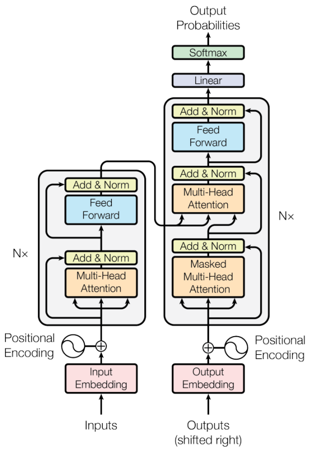


z_1: Buckingham Palace is home to…

x: we paid 20 __ at the Buckingham…

Low

$$p(y = 'pounds' | x, z_2)$$

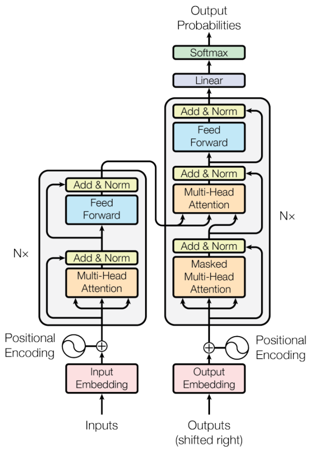


z_2: The Wall Street …

Solution: try different documents

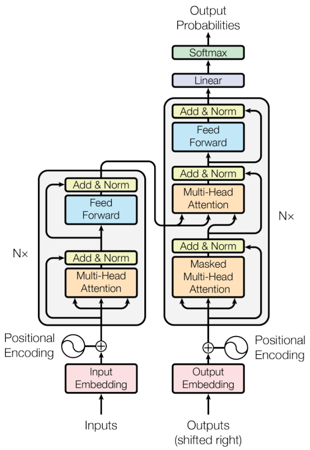
High

$$p(y = \text{'pounds'} | x, z_1)$$



Low

$$p(y = \text{'pounds'} | x, z_2)$$



z_1: Buckingham Palace is home to…

Neural Retriever: $p(z|x)$

z_2: The Wall Street …

x: we paid 20 __ at the Buckingham…

The Model

Tencent AI Lab

$$p(\textcolor{magenta}{y}|\textcolor{blue}{x}) = \sum_{\textcolor{purple}{z}} p(\textcolor{magenta}{y}|\textcolor{blue}{x},\textcolor{purple}{z})\,p(\textcolor{purple}{z}|\textcolor{blue}{x})$$

Knowledge-Augmented Encoder

Neural Retriever

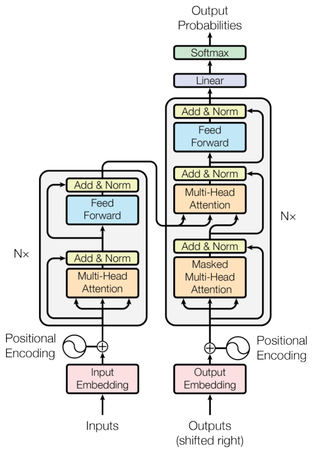
Challenge: Summation over millions of documents!
(for every sample, ever gradient step)

Approximation: Dual-Encoder + MIPS

Tencent AI Lab

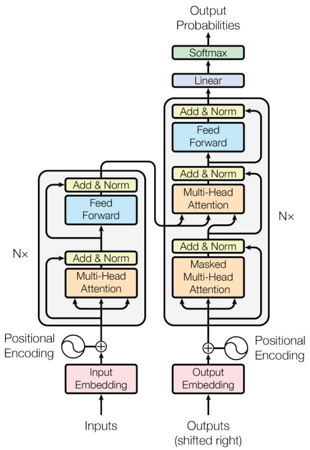
Retriever: $p(z|x) \propto h(x)^T h(z)$

$h(x)$



x: we paid 20 __ at the Buckingham…

$h(z)$



z: Buckingham Palace is home to…

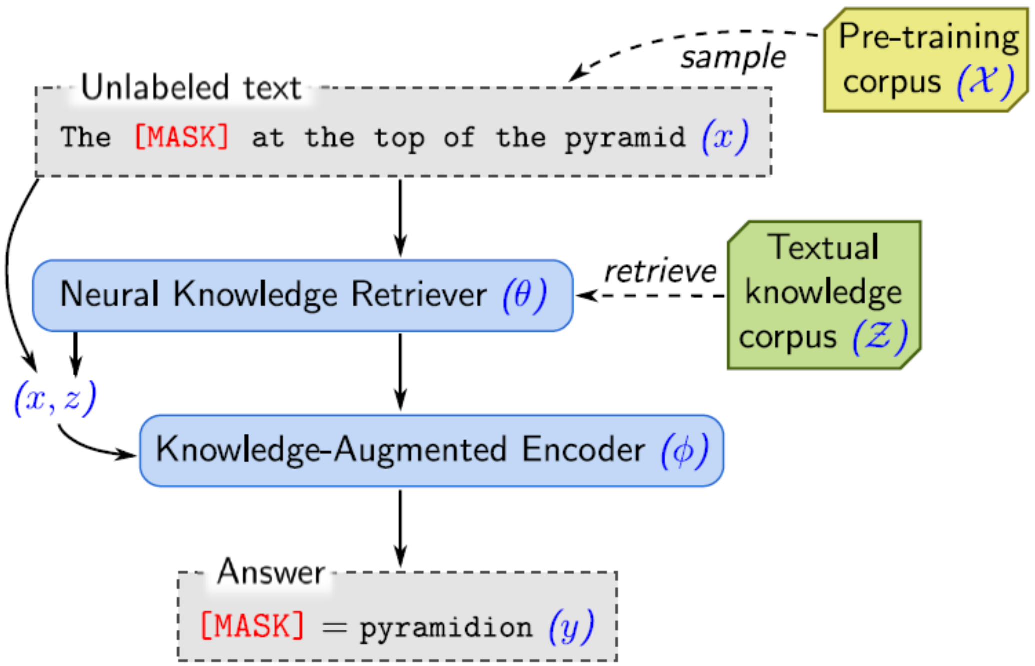
- Search top-k candidates via MIPS tool:

$$p(y|x) = \sum_z p(y|x,z)p(z|x)$$

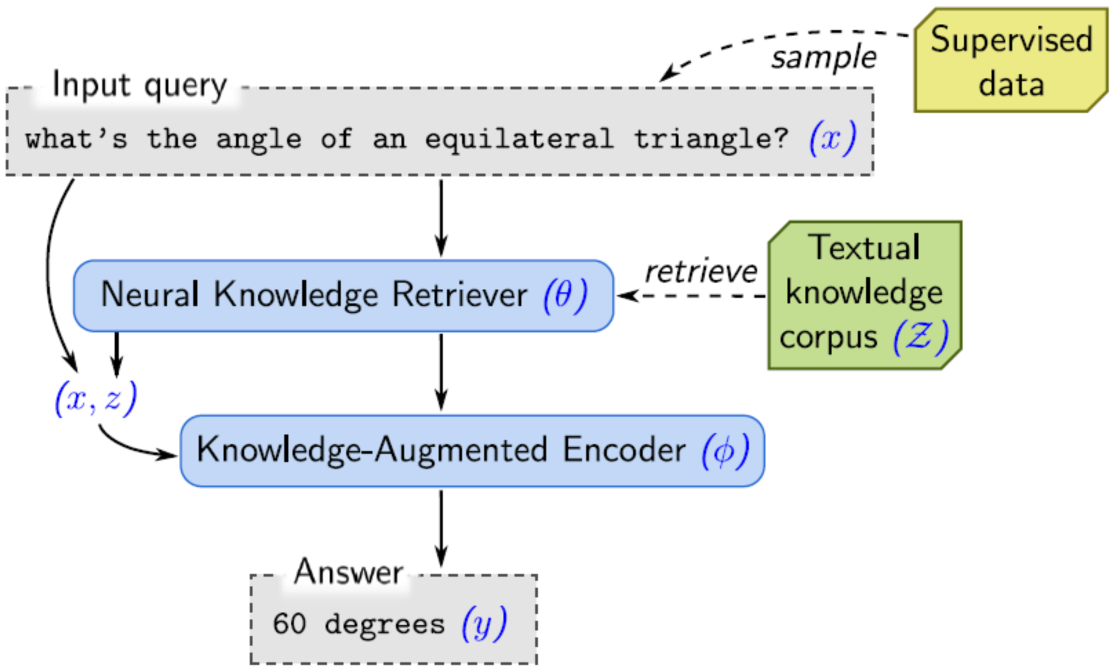
$$= \sum_{z \in MIPS(x)} p(y|x,z)p(z|x)$$

Pretrain and Fine-tune

Pre-training (REALM):



Fine-tuning (Open-domain QA):



[Guu+ 20]

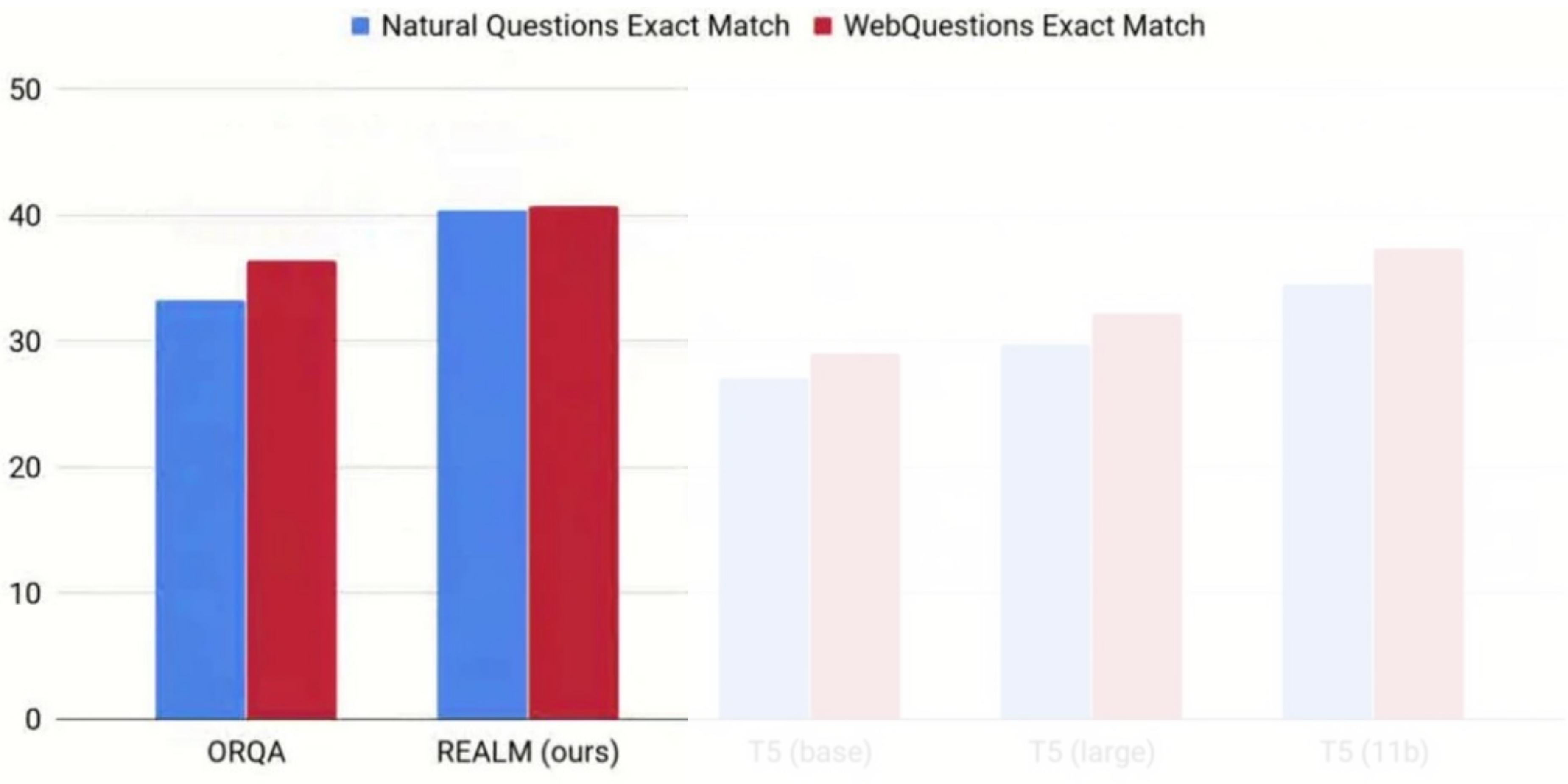
Key Results

Tencent AI Lab

- 3 open-domain QA datasets:
  - Natural Questions, WebQuestions, CuratedTrec

- Baselines
  - ORQA (Lee et al. 2019) – 330M paras
    - Equivalent to REALM without joint training
  - T5-base (220M), L (770M), XL (11B) (Raffel et al. 2019)

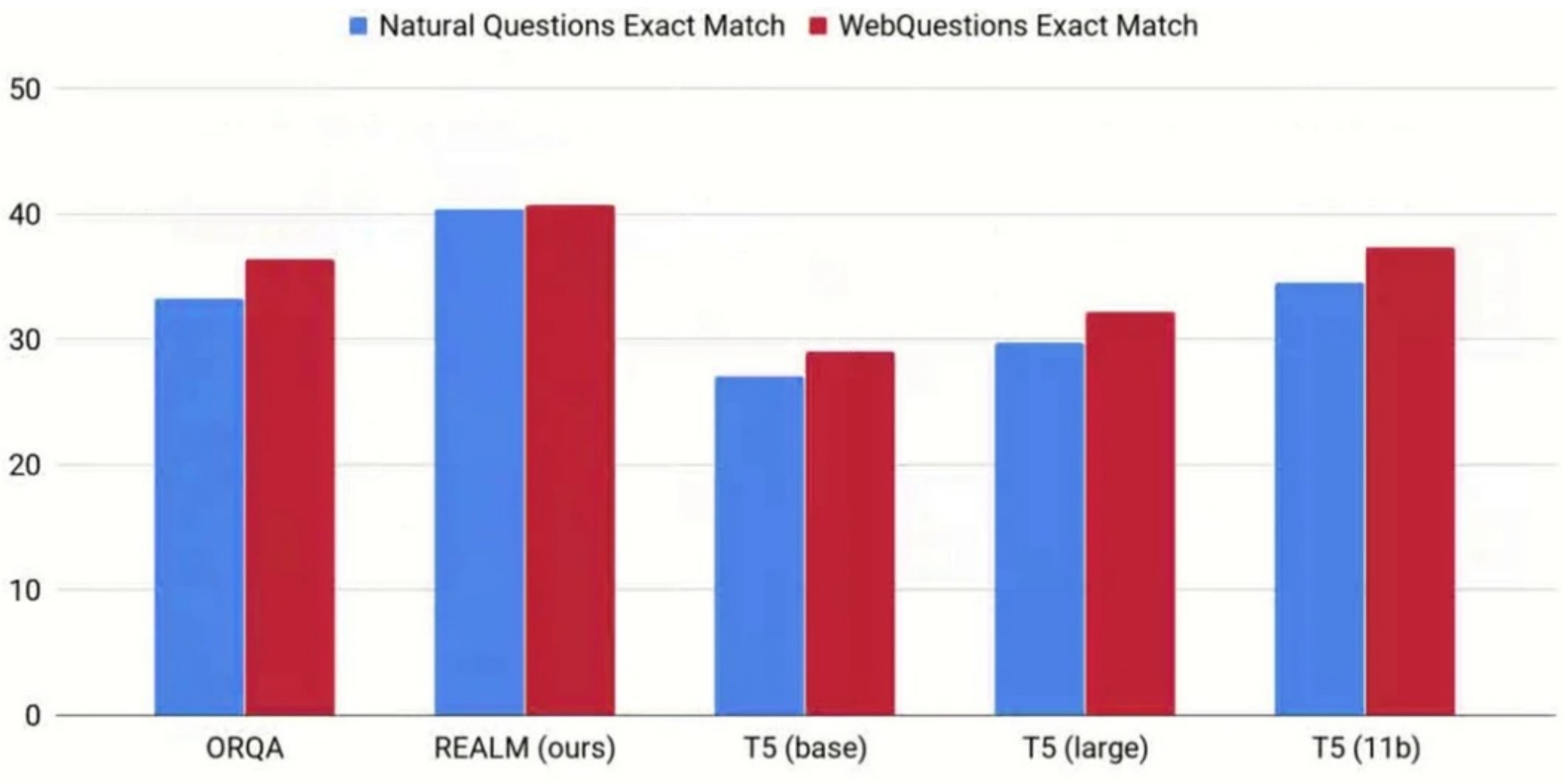
Key Results



[Guu+ 20]

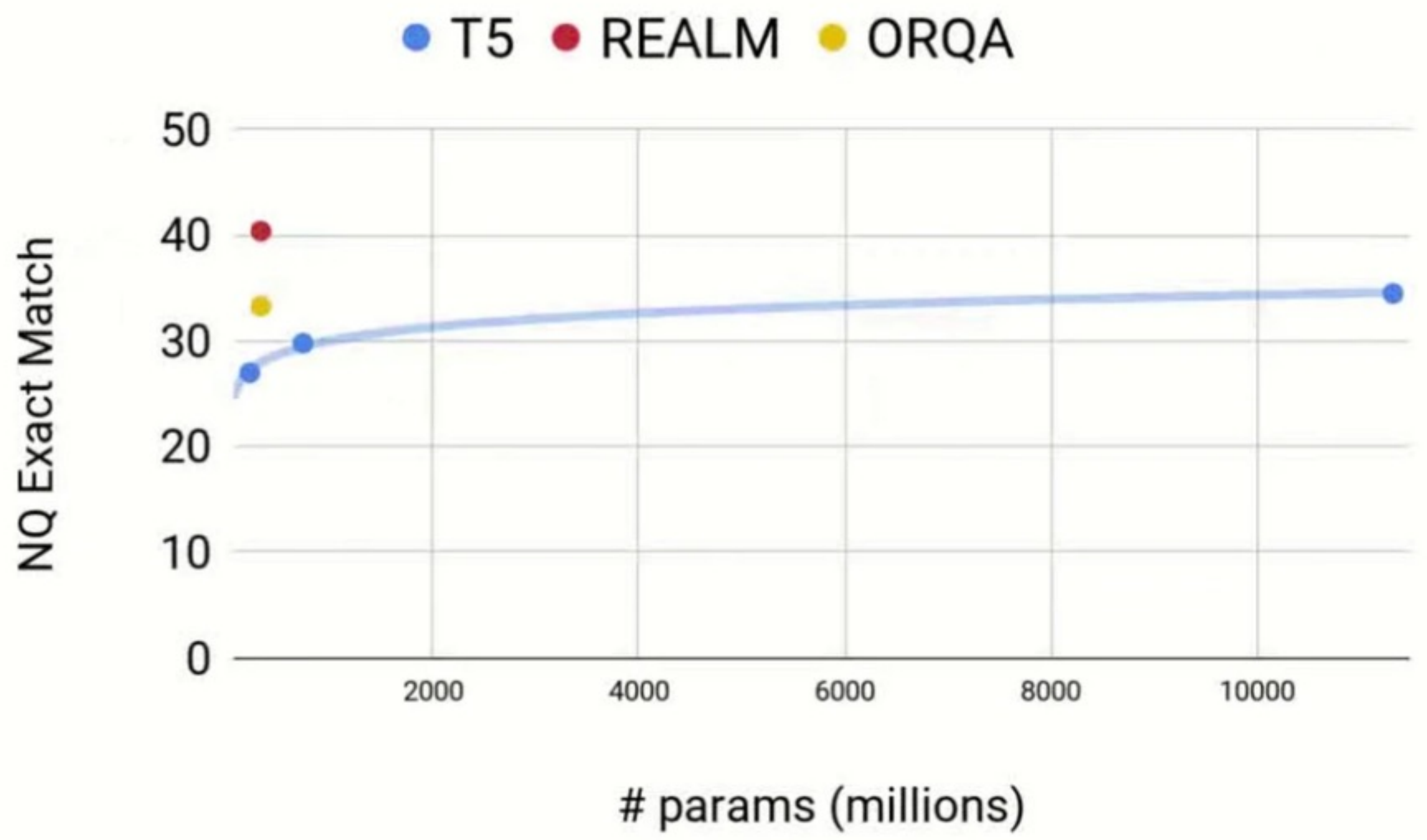
Key Results

Tencent AI Lab



[Guu+ 20]

Key Results



T5
REALM
ORQA
NQ Exact Match
50
40
30
20
10
0
2000
4000
6000
8000
10000
params (millions)

Comparison with KNN-LM

- Learnable Retriever and Joint Training Matters!

- Limitation:
  - Masked Language Model is unfriendly to Sequence Generation Tasks
  - Retrieval in very coarse-grained (document) level

Retrieval-Augmented Auto-Regressive LM

Tencent AI Lab

DeepMind

Improving language models by retrieving from trillions of tokens

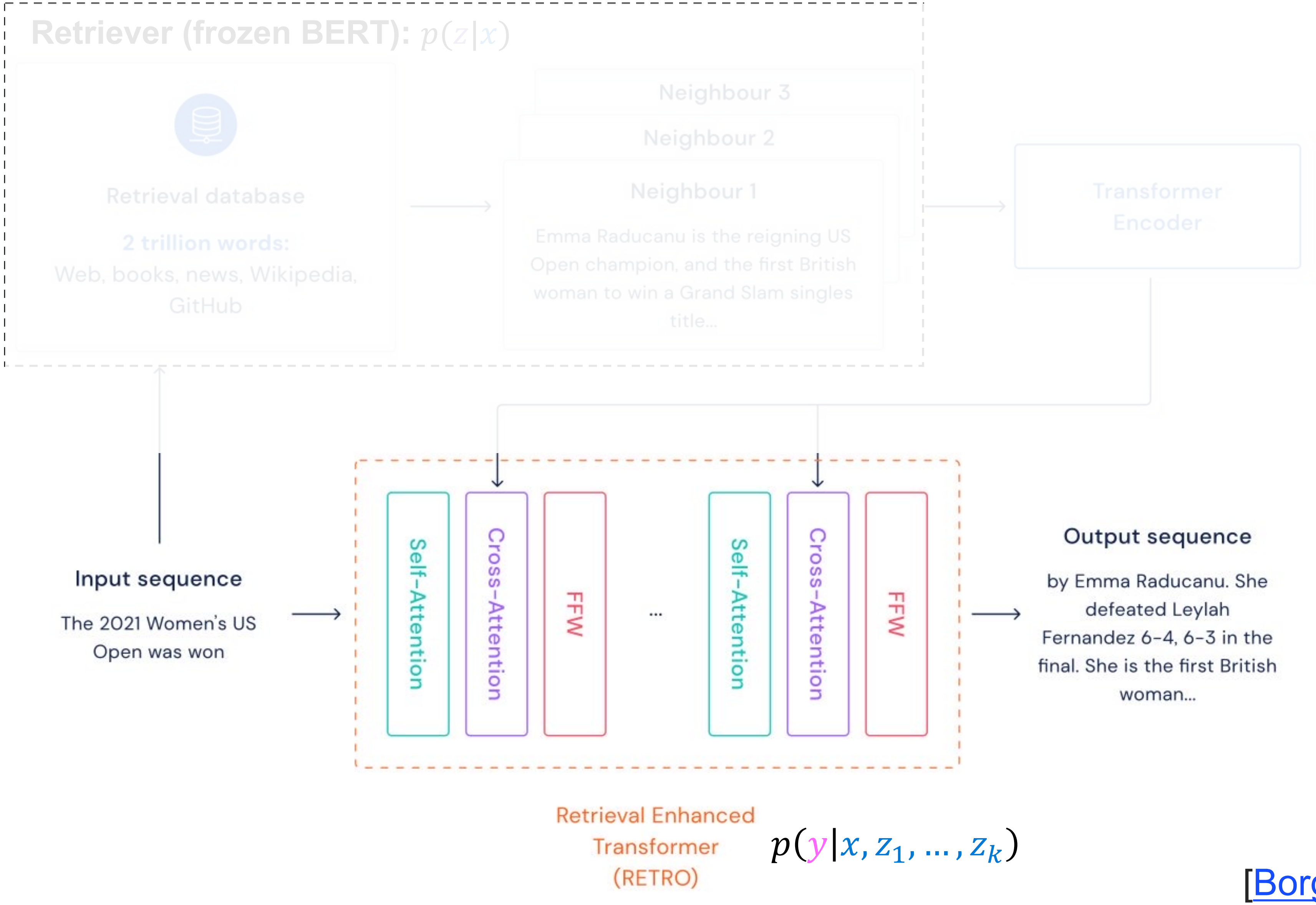
Sebastian Borgeaud[†], Arthur Mensch[†], Jordan Hoffmann[†], Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae[‡], Erich Elsen[‡] and Laurent Sifre[†,‡]

All authors from DeepMind, [†]Equal contributions, [‡]Equal senior authorship

Big Index + Small model

- RETRO: Retrieval-Enhanced transformer
  - Bigger and Bigger index:
    - from 200M~2B tokens (KNN-LM, REALM) to 2T tokens (RETRO)
  - Smaller and Smaller Model:
    - From 175B parameters (GPT3) to 172M ~ 7.5B parameters (RETRO)
  - Efficient training:
    - Works well without joint training

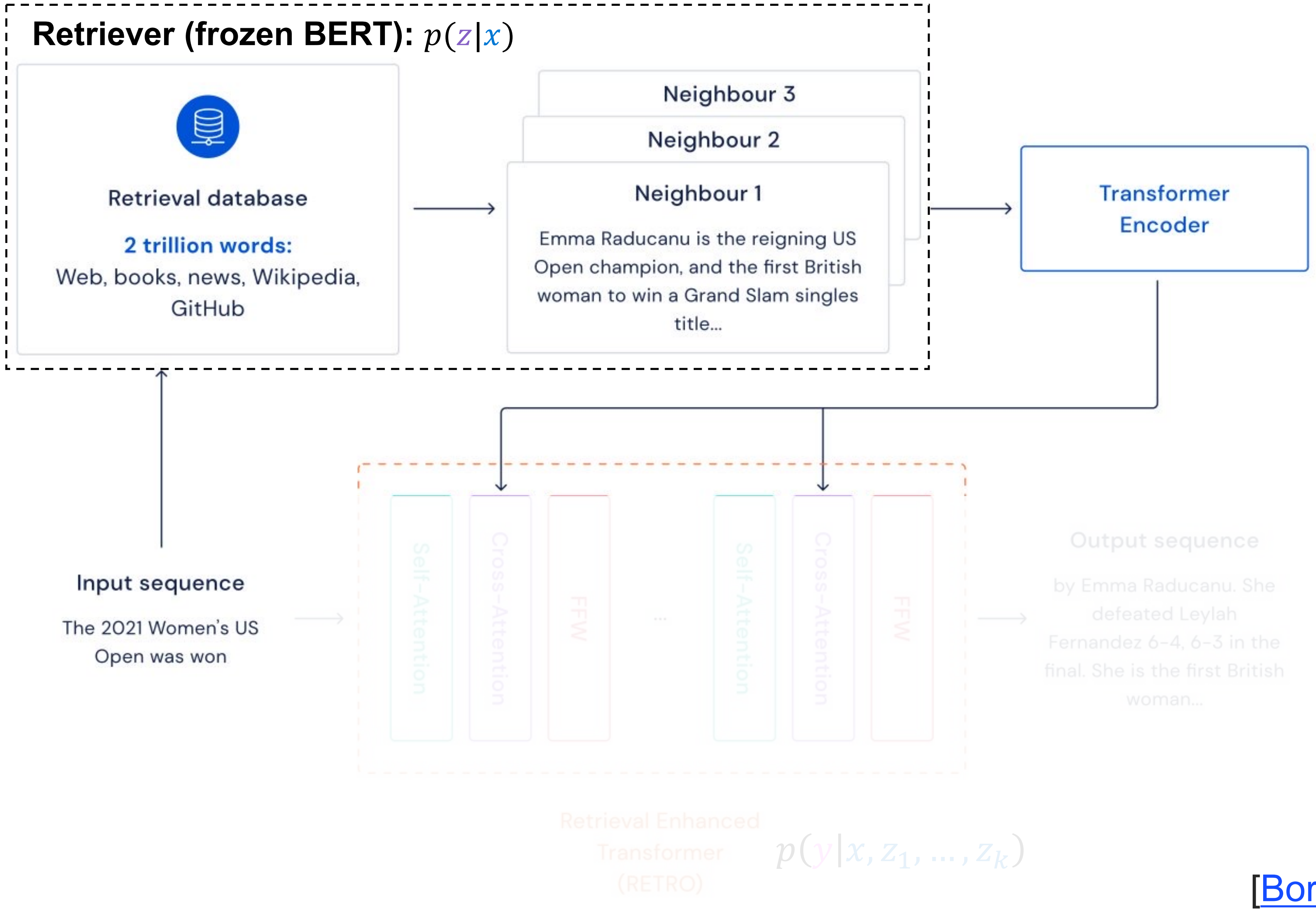
Main Framework: Decoder



$p(y|x, z_1, \ldots, z_k)$

[Borgeaud+ 22]

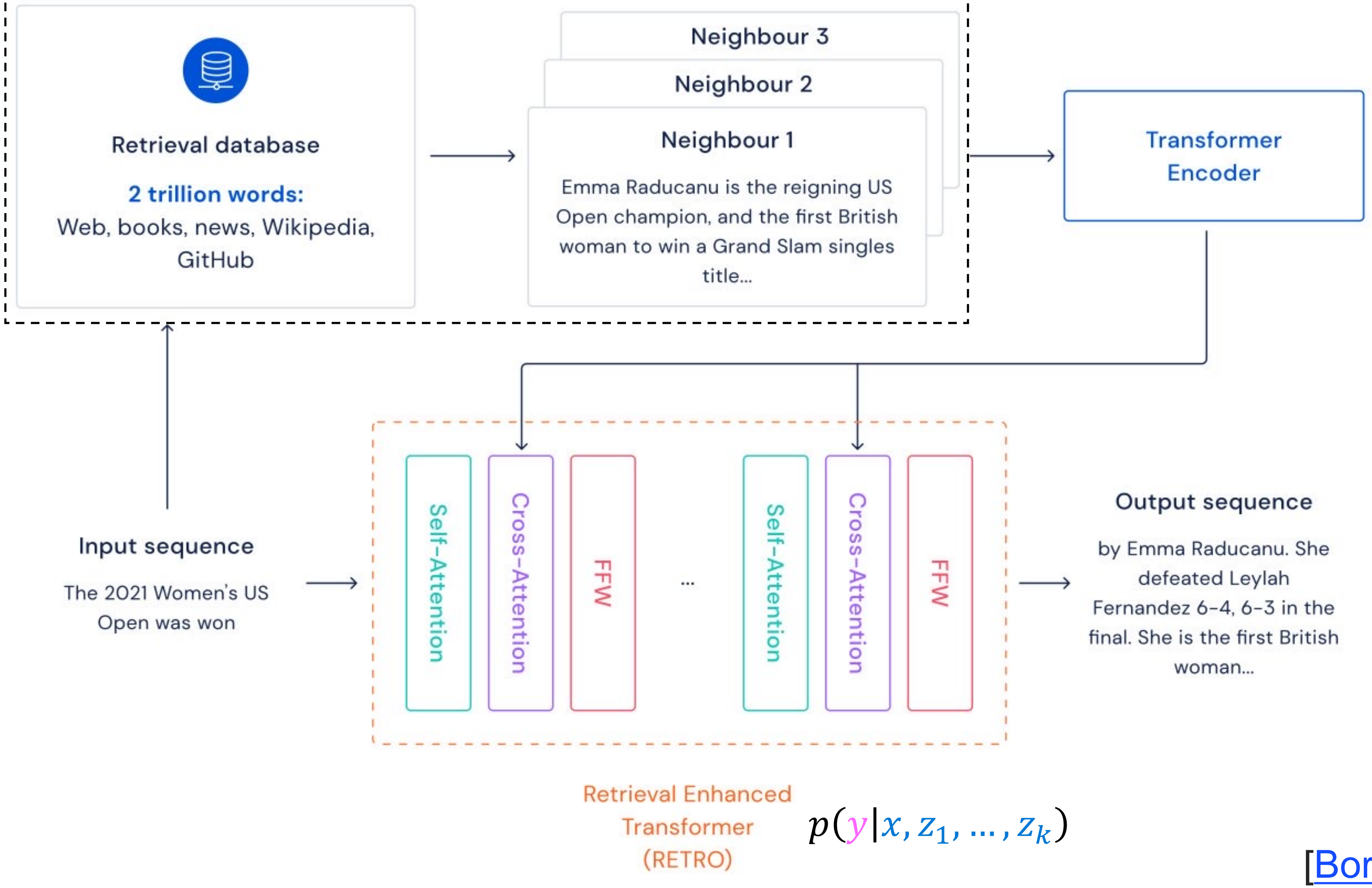
Main Framework: Memory-Encoder



[Borgeaud+ 22]

Main Framework: Encoder-Decoder

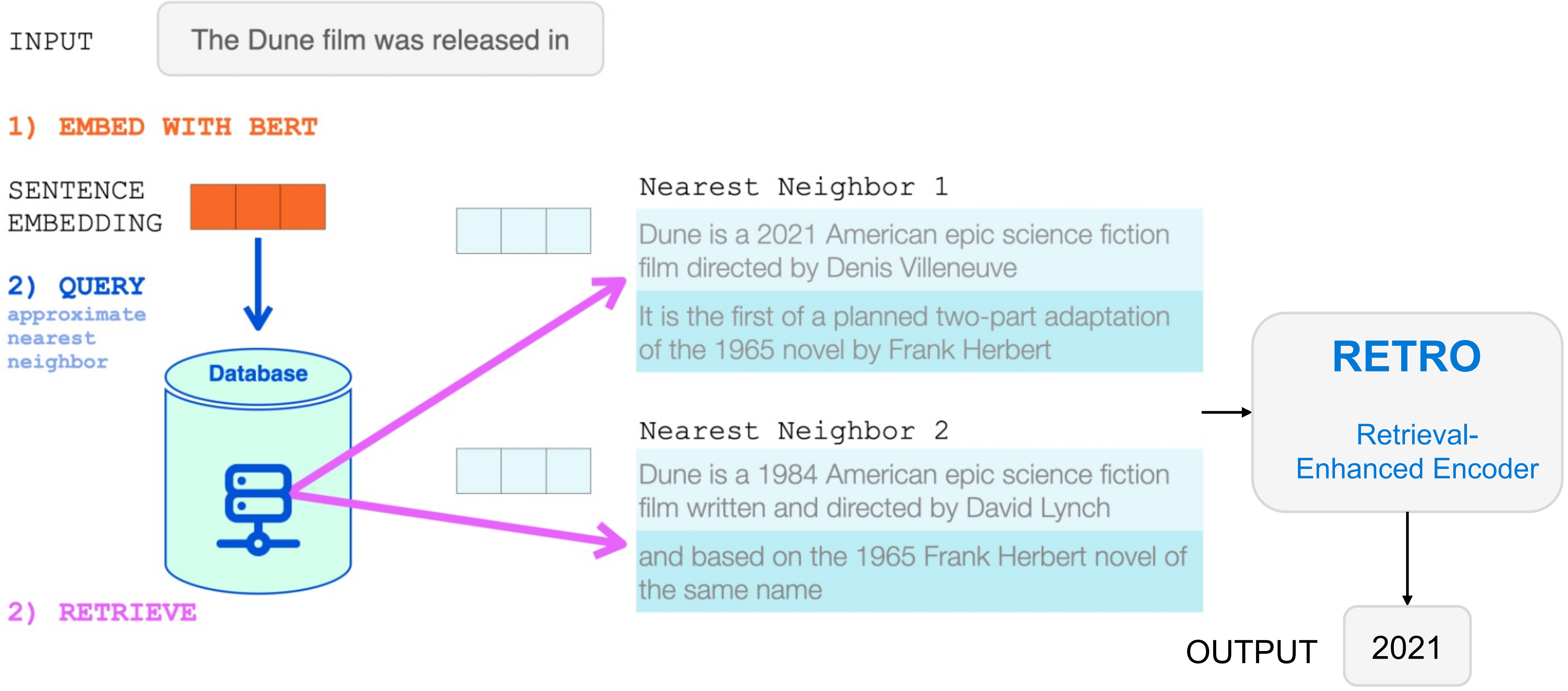
Retriever (frozen BERT): $p(z|x)$



$p(y|x, z_1, \dots, z_k)$

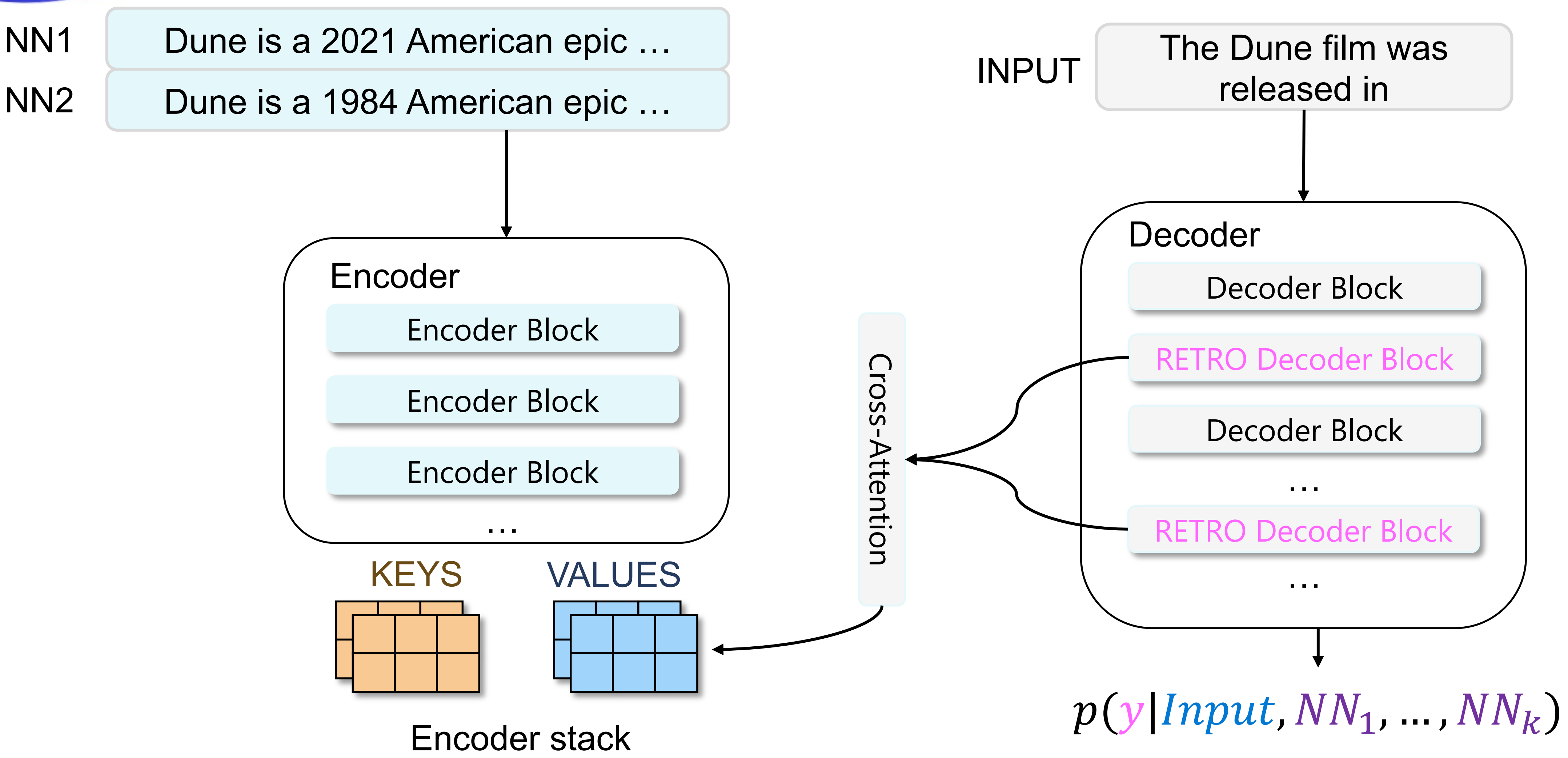
[Borgeaud+ 22]

Nearest Neighbor Search



https://jalammar.github.io/illustrated-retrieval-transformer/

Retrieval-Augmented Generation



Experimental Baselines

- Baselines:
  - Small models:

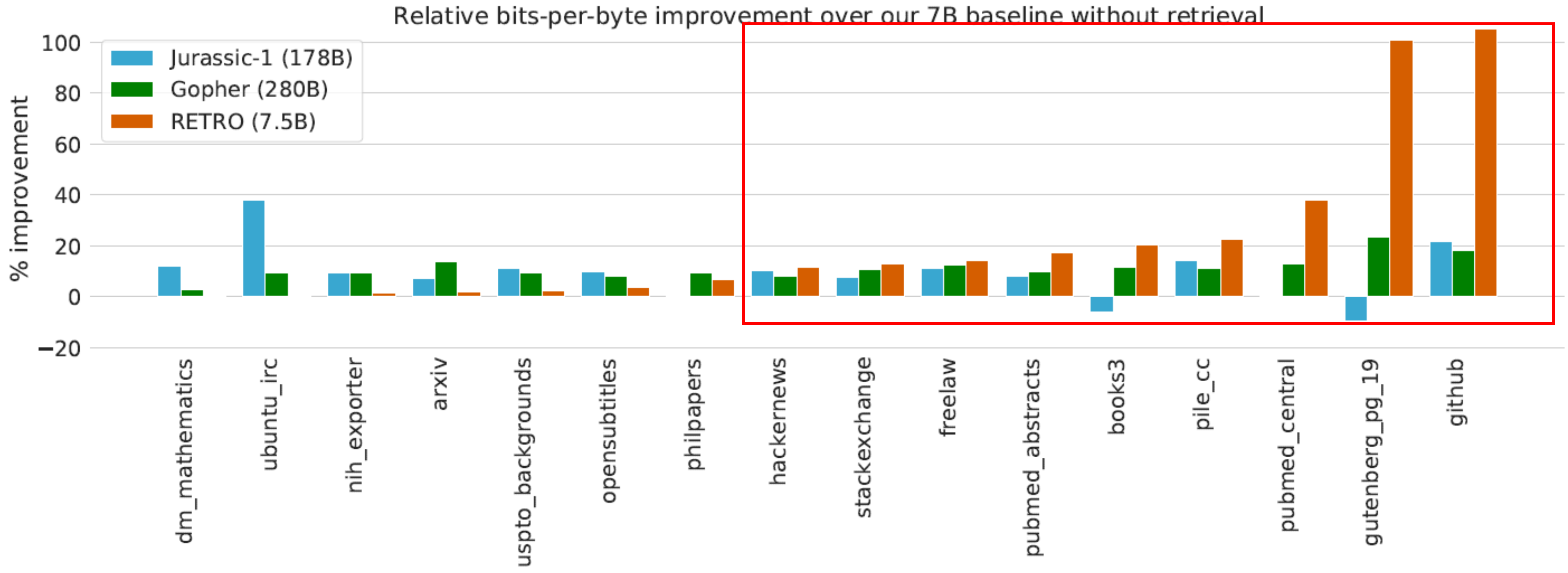
Baseline parameters	RETRO	d	d_{ffw}	# heads	Head size	# layers
132M	172M (+30%)	896	3,584	16	64	12
368M	425M (+15%)	1,536	6,144	12	128	12
1,309M	1,451M (+11%)	2,048	8,192	16	128	24
6,982M	7,532M (+8%)	4,096	16,384	32	128	32

  - Jurasic-1 (Lieber et al., 2021): 178B parameters
  - Gopher (Rae et al., 2021): 280B parameters

Gopher and Jurrasic-1 outperforms GPT-3 in most tasks!

Comparison with Large Models

- Outperforms 7B baselines on all test sets
- Comparable with two very large models (178B and 280B) in 16 domains



Comparison with Other Retrieval-Augmented Models

- Performance gain from big Database

Model	Retrieval Set	#Database tokens	#Database keys	Valid	Test
Adaptive Inputs (Baevski and Auli, 2019)	-	-	-	17.96	18.65
SPALM (Yogatama et al., 2021)	Wikipedia	3B	3B	17.20	17.60
*k*NN-LM (Khandelwal et al., 2020)	Wikipedia	3B	3B	16.06	16.12
Megatron (Shoeybi et al., 2019)	-	-	-	-	10.81
Baseline transformer (ours)	-	-	-	21.53	22.96
*k*NN-LM (ours)	Wikipedia	4B	4B	18.52	19.54
RETRO	Wikipedia	4B	0.06B	18.46	18.97
RETRO	C4	174B	2.9B	12.87	10.23
RETRO	MassiveText (1%)	18B	0.8B	18.92	20.33
RETRO	MassiveText (10%)	179B	4B	13.54	14.95
RETRO	MassiveText (100%)	1792B	28B	**3.21**	**3.92**

An Interesting Sample

The RETRO model stays more on-topic than the baseline sample

Input prompt

Beavers are interesting animals that live near rivers. They build

Baseline 7.1B sample

dams to create ponds. Frogs are amphibians so they can live in both land and water. They have great camouflage to hide from predators. The Golden Retriever, or Golden, is a very big dog...

RETRO 7.5B sample

their houses called beaver dams in the riverbeds. They also live on land. Beavers use their strong teeth and strong jaws to cut down trees and branches to build their homes. They also use their teeth and jaws to chew up food.

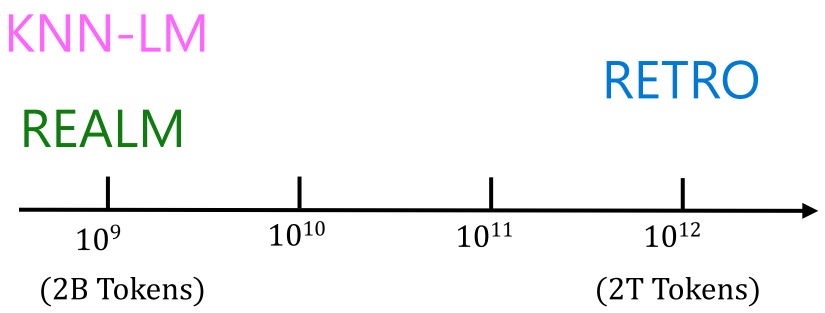
The Evolution of Retrieval-Augmented LM

- Three types:
  - KNN-LM——Token-level and Interpolation-based model
  - REALM——Document-level and Joint-Training model
  - RETRO——Chunk-level, Frozen-Retriever, huge index model

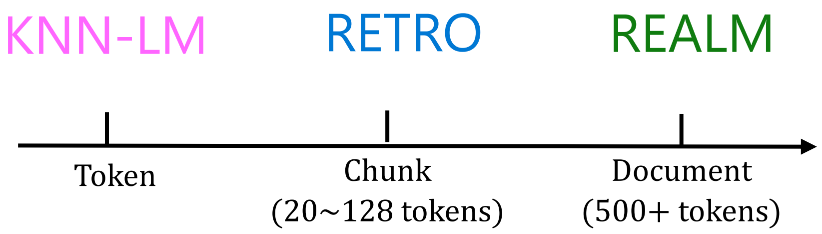
	# Retrieval tokens	Granularity	Retriever training	Retrieval integration
Continuous Cache	$\mathcal{O}(10^3)$	Token	Frozen (LSTM)	Add to probs
kNN-LM	$\mathcal{O}(10^9)$	Token	Frozen (Transformer)	Add to probs
SPALM	$\mathcal{O}(10^9)$	Token	Frozen (Transformer)	Gated logits
DPR	$\mathcal{O}(10^9)$	Prompt	Contrastive proxy	Extractive QA
REALM	$\mathcal{O}(10^9)$	Prompt	End-to-End	Prepend to prompt
RAG	$\mathcal{O}(10^9)$	Prompt	Fine-tuned DPR	Cross-attention
FiD	$\mathcal{O}(10^9)$	Prompt	Frozen DPR	Cross-attention
EMDR^2	$\mathcal{O}(10^9)$	Prompt	End-to-End (EM)	Cross-attention
RETRO (ours)	$\mathcal{O}(10^{12})$	**Chunk**	**Frozen (BERT)**	**Chunked cross-attention**

The Difference

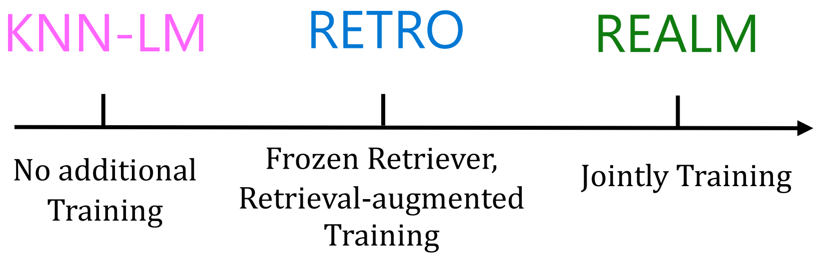
- Datastore Size:



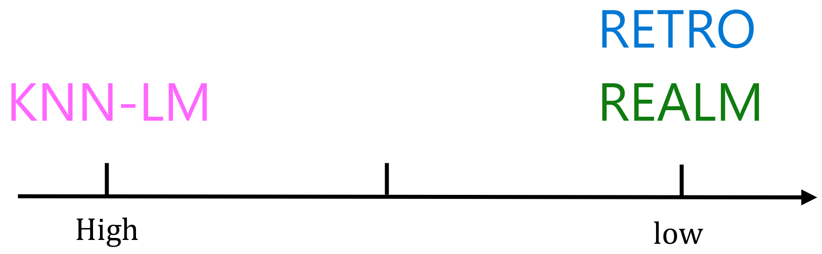
- Datastore granularity:



- Training Complexity:



- Inference Latency:



Outline

- Background and Introduction
- Language Modeling
- **Open-Domain Dialogue Systems**
  - **Background and Motivation**
  - **Shallow Integration**
  - **Deep Integration**
- Neural Machine Translation
- Conclusion and Outlook

Dialogue Systems

- Dialogue Systems aim to bridge humans and machines with a natural language interface.

JARVIS – Iron Man's Personal Assistant

Baymax – Personal Healthcare Companion

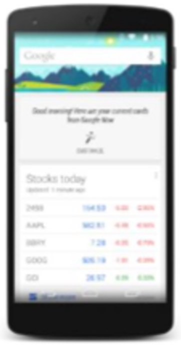
- Humans have long dreamed a machine that understands our languages and responds accordingly.

*Figure [Chen & Gao 17]

Real-world Dialogue Systems

- Dialogue Systems aim to bridge humans and machines with a natural language interface.

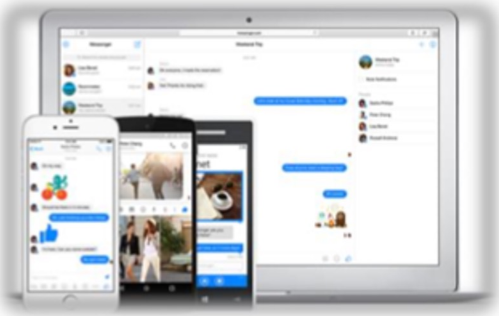
Apple Siri (2011)



Google Now (2012)
Google Assistant (2016)

Microsoft Cortana (2014)

Amazon Alexa/Echo (2014)



Facebook M & Bot (2015)

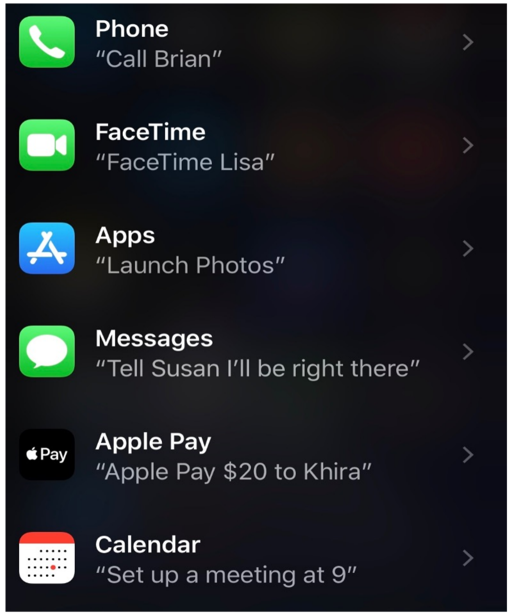
Google Home (2016)

Apple HomePod (2017)

*Figure [Chen & Gao 17]

Categorization of Dialogue Systems

- Dialogue Systems can be categorized into three classes.
  - Task-oriented bot "I need to get this done"
  - Question answering bot "I have a question"
  - Open-domain chit-chat bot "Let's chat for fun"



Apple Siri

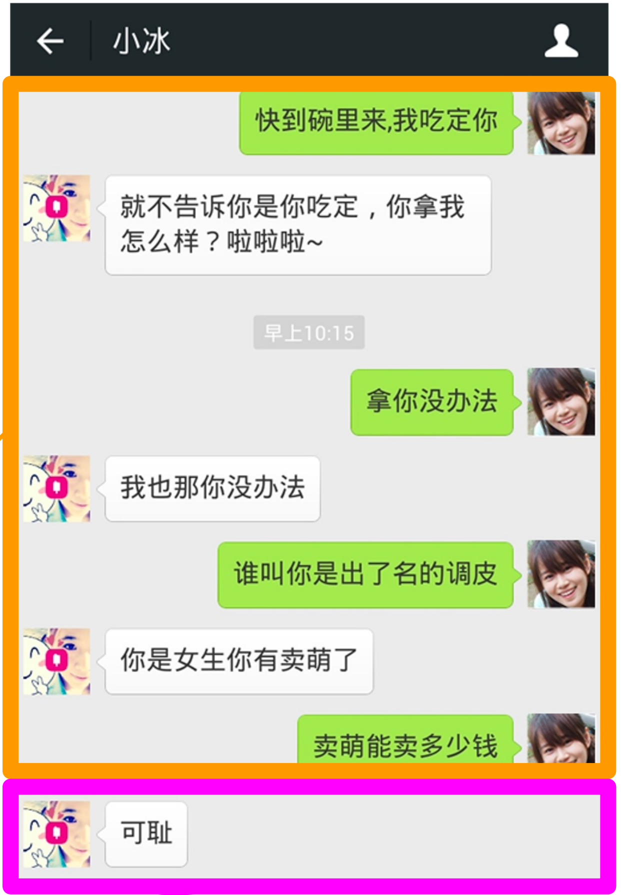
IBM Watson won Jeopardy Q&A

XiaoIce

- It is also possible to put them in one chat bot

Open-domain Chit-chat Systems

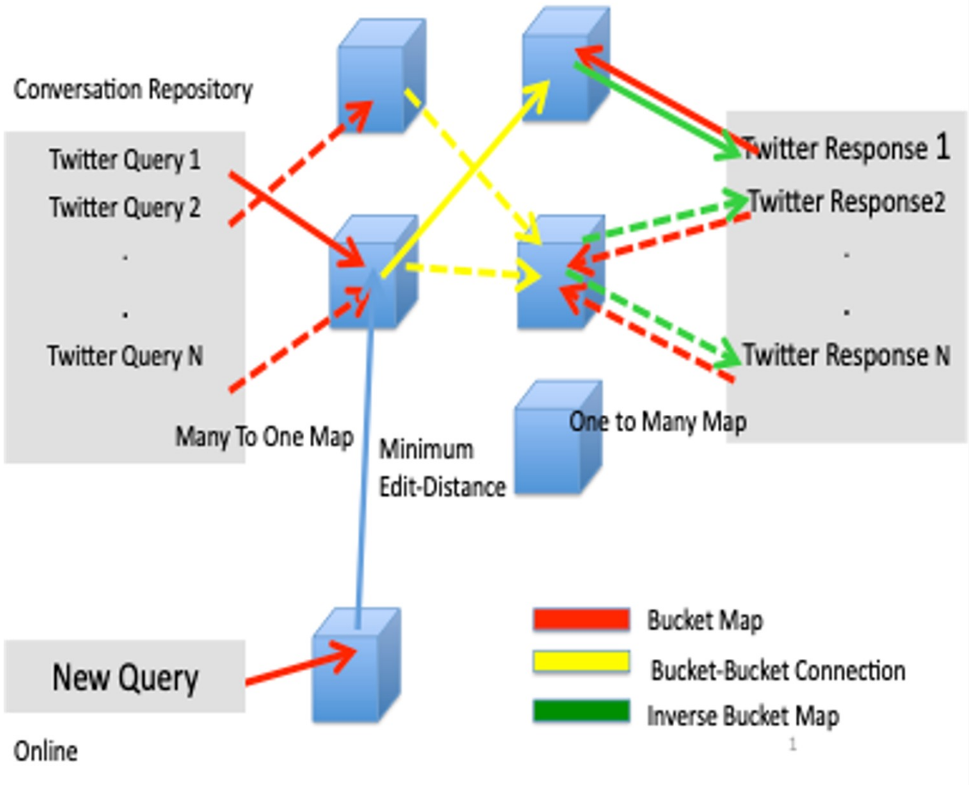
- Dialogue Systems can be categorized into three classes.
  - Task-oriented bot "I need to get this done"
  - Question answering bot "I have a question"
  - Open-domain chit-chat bot "Let's chat for fun"
- Compared to other types, open-domain chit-chat is
  - More open-ended (one-to-many)
  - focused on creating human-like conversations
  - Not restricted in specific domains or tasks

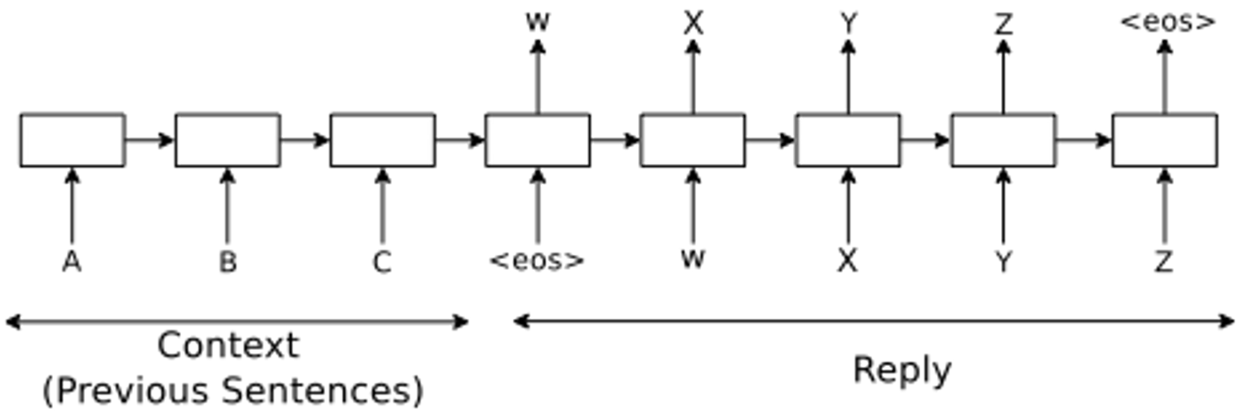


- input: context/query/history
- output: response

Approaches to Open-domain Chit-chat Systems

- Early work in data-driven dialogue response systems
  - retrieval-based [Jafarpour+ 10;Ji+ 14;Hu+ 15]
  - Generation-based [Sordoni+ 15; Vinyals & Le 15; Shang+ 15]

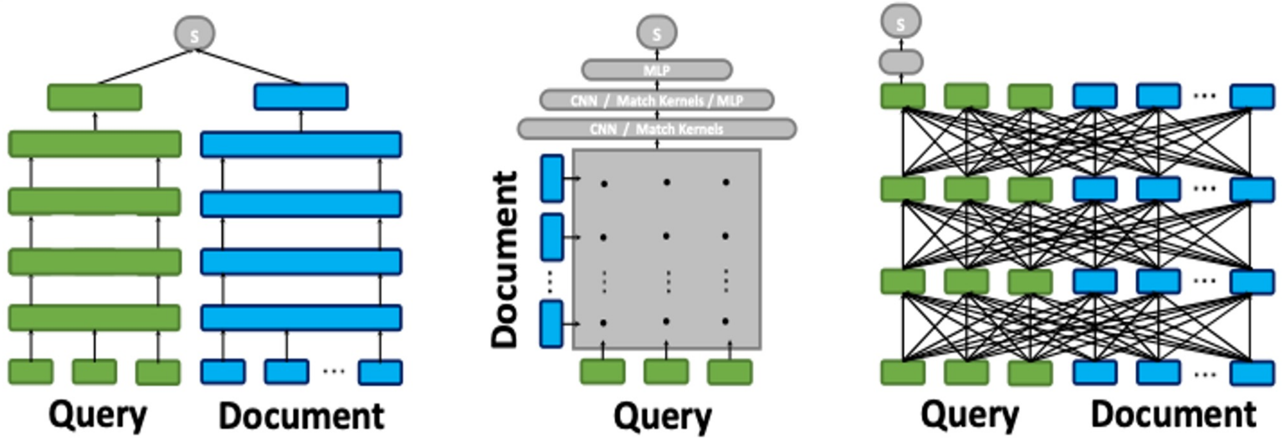




Retrieval-based Dialogue Response Systems

- The **ingredients** of retrieval-based dialogue response systems
  - A (large) database of context-response pairs (or single utterances)
  - A similarity function measuring context-context similarity (e.g, BM25, TFIDF)
  - A relevance function measuring context-response relevance
- Most recent work has been focused on context-response relevance

query-document



(a) Representation-based Similarity *(e.g., DSSM, SNRM)*
(b) Query-Document Interaction *(e.g., DRMM, KNRM, Conv-KNRM)*
(c) All-to-all Interaction *(e.g., BERT)*

classic problem in information retrieval

[Khattab & Zaharia 20]

Pros & Cons of Retrieval-based Systems

- Advantages:
  - fluent
  - informative
  - controllable

written & filtered by humans!

- Disadvantage:
  - This is likely that there is **no** appropriate response in the database

not tailored for input context!

User: How do you like the movie Iron Man?

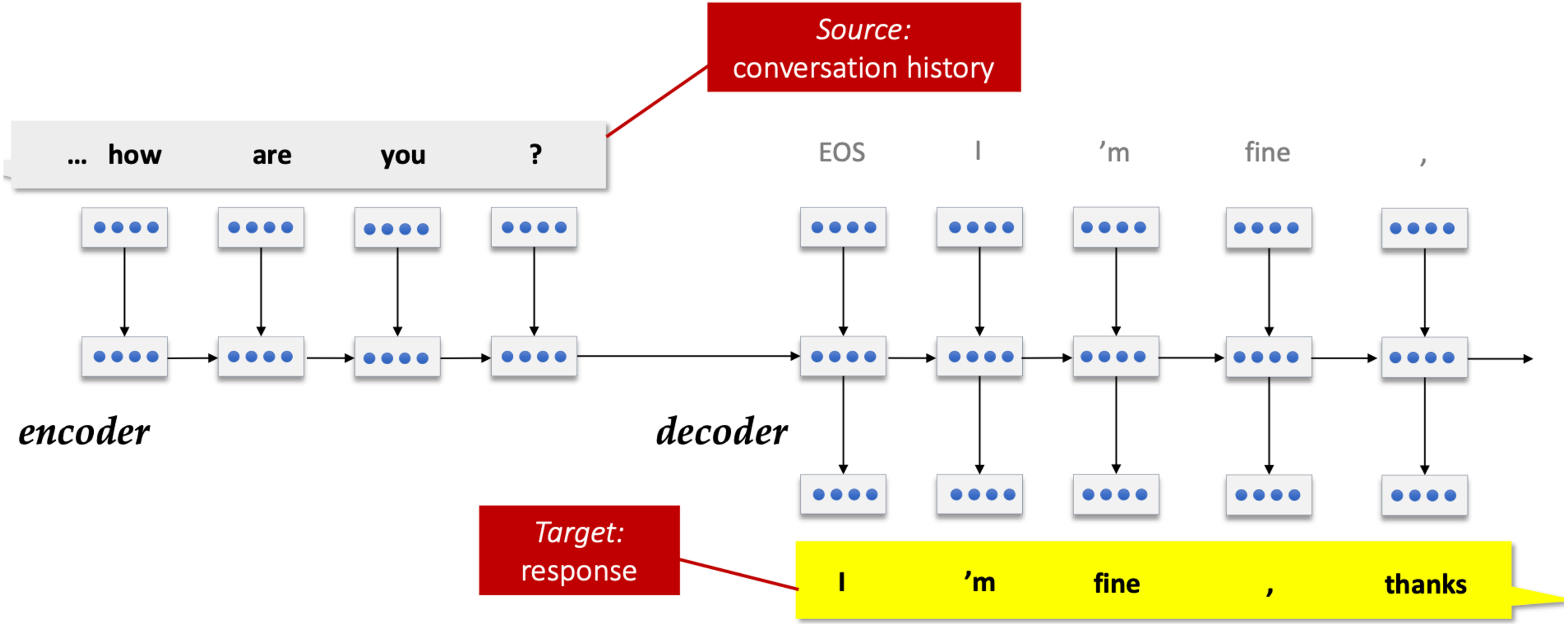
System: Oh, I almost cried when the Batman races to save Rachel.

User: What are you talking about?

* suppose Iron Man is not included the database

Generation-based Dialogue Response Systems

- Generation-based dialogue response systems
  - Seq2Seq (encoder-decoder), similar to neural machine translation
  - RNN/CNN/Transformer etc

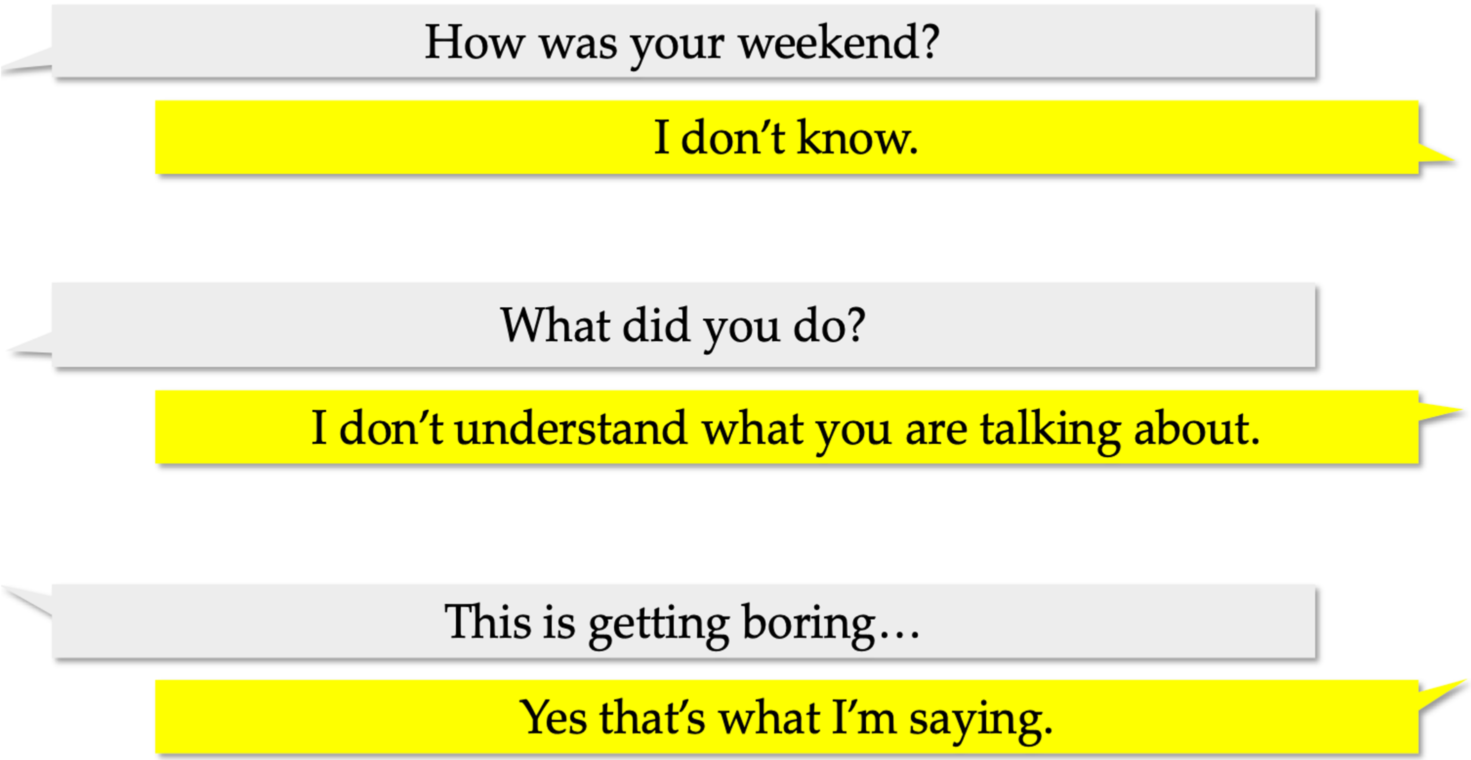


[Sordoni+ 15; Vinyals & Le 15; Shang+ 15]

*Figure [Gao+ 18]

Pros & Cons of Generation-based Systems

- Advantages:
  - universal
  - coherent
- Disadvantages:
  - Boring
  - Uninformative
  - Less controllable



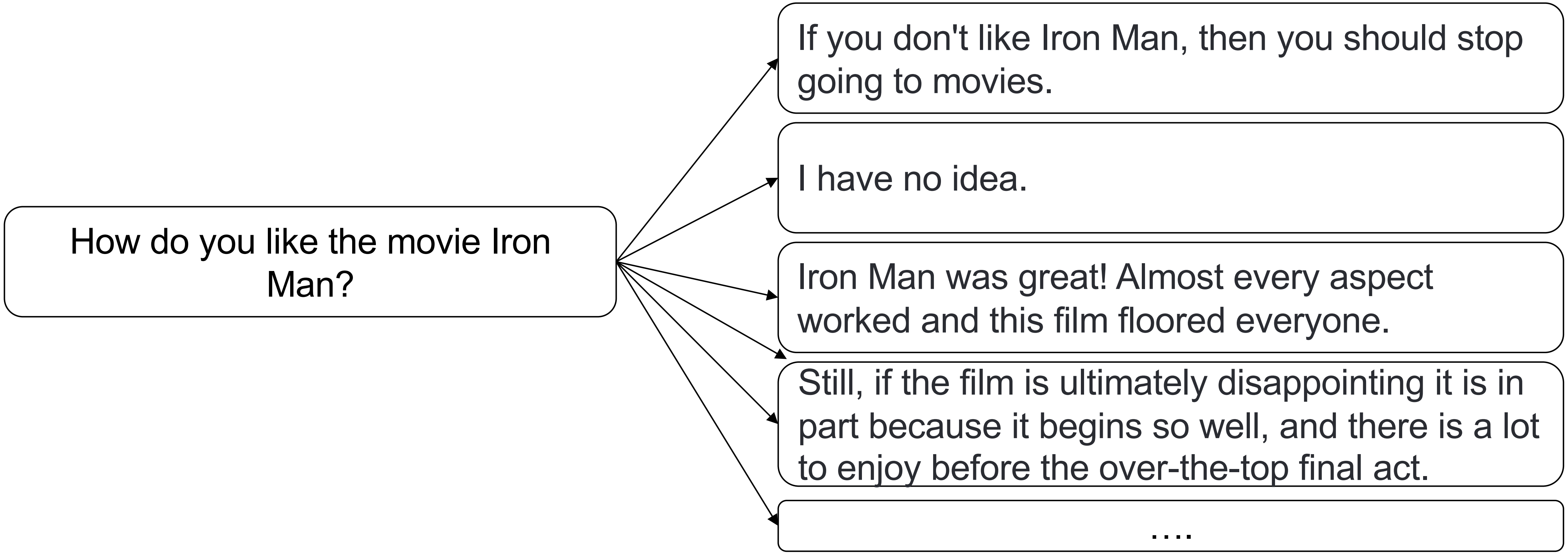
it could say anything

Or…just say "**I don't know!**"

*Figure [Gao+ 18]

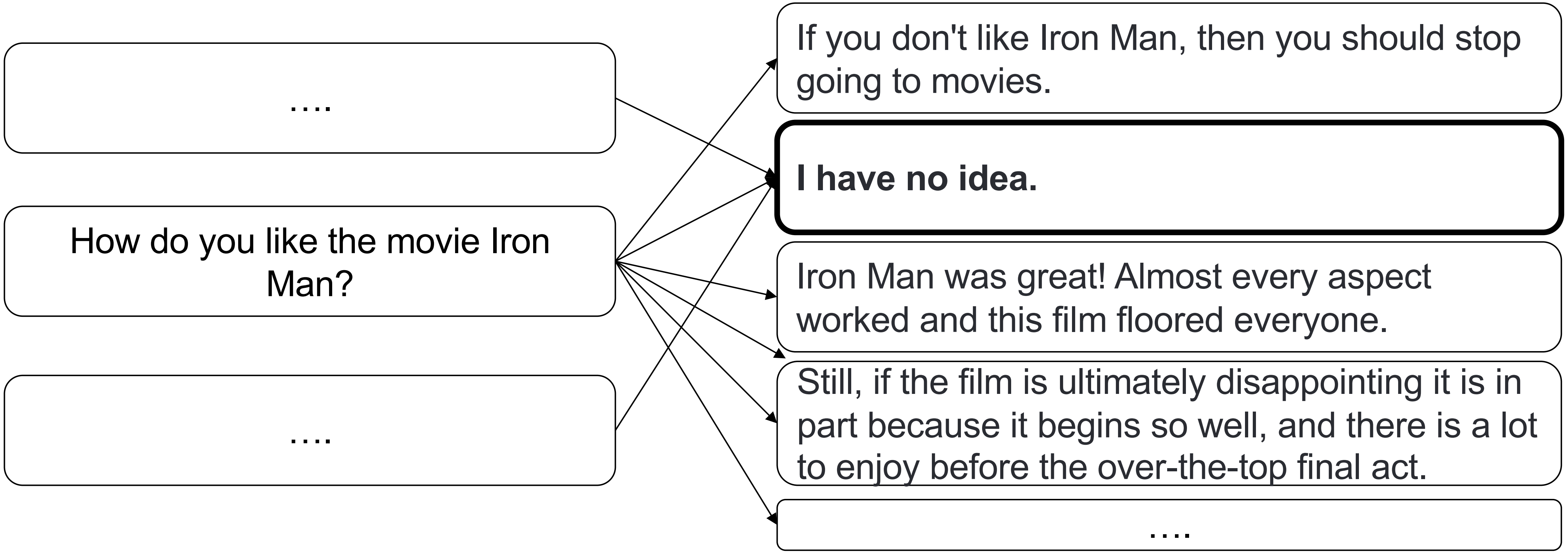
Safe Response Problem

- Safe response problem is one most critical issue in generation-based systems
- Recall the goal of open-domain chit-chat
  - maximize user engagement with informative and enjoyable human-like responses
- Cause: trained models prefer the most common response among others



Safe Response Problem

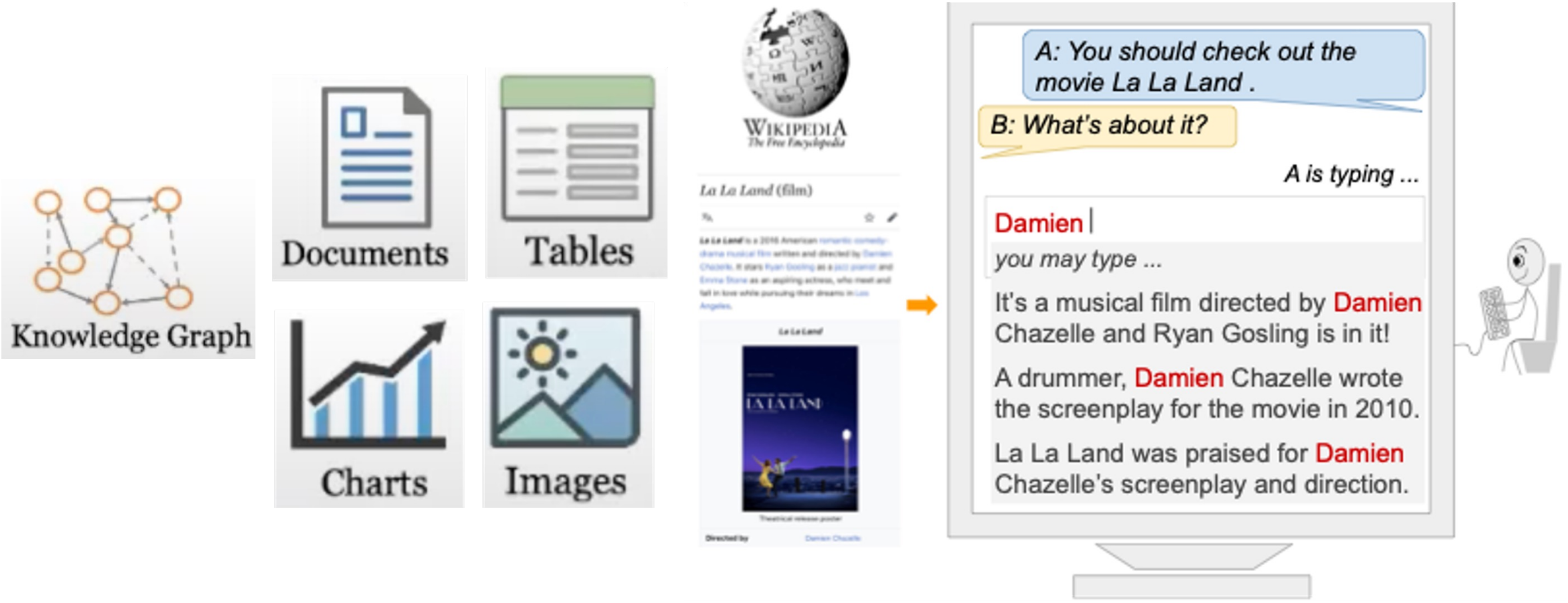
- Safe response problem is one most critical issue in generation-based systems
- Recall the goal of open-domain chit-chat
  - maximize user engagement with informative and enjoyable human-like responses
- Cause: trained models prefer the most common response among others



Remedies for the Safe Response Problem

- One-to-many modeling [Li+ 16; Zhao+ 17; Zhou+ 17; Zhang+ 18; etc]
  - Conditional variational autoencoder, reinforcement Learning, persona, emotion, etc.

- Grounded response generation [Dinan+ 18; Zhou+ 18; Wu+ 21; Komeili+ 22; etc]
  - Grounded on documents, knowledge graphs, images, etc



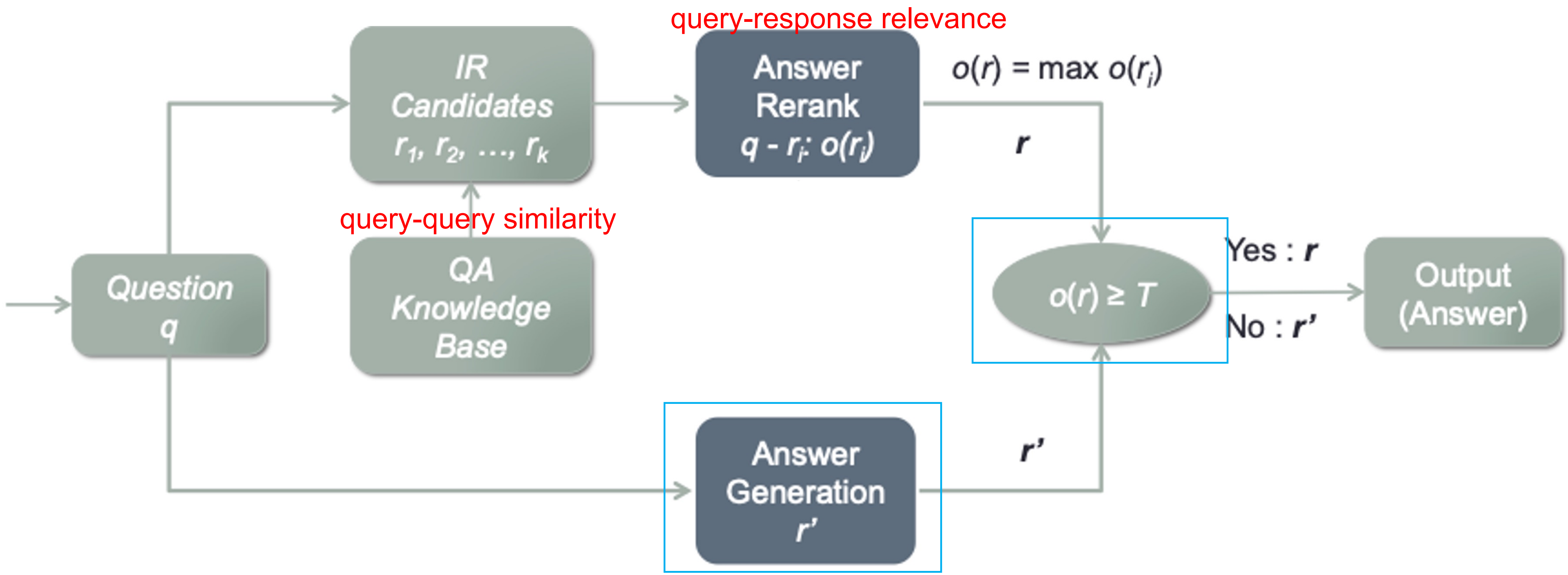
Retrieval vs. Generation

	Retrieval-based Systems	Generation-based Systems
Informativeness	**informative, long**	bland, short
Relevance	good only if similar contexts are in the database	**can generate new responses to unseen contexts**
Controllability	easy to control the database	Blackbox neural models

Retrieval + Generation?

Shallow Integration of Retrieval and Generation

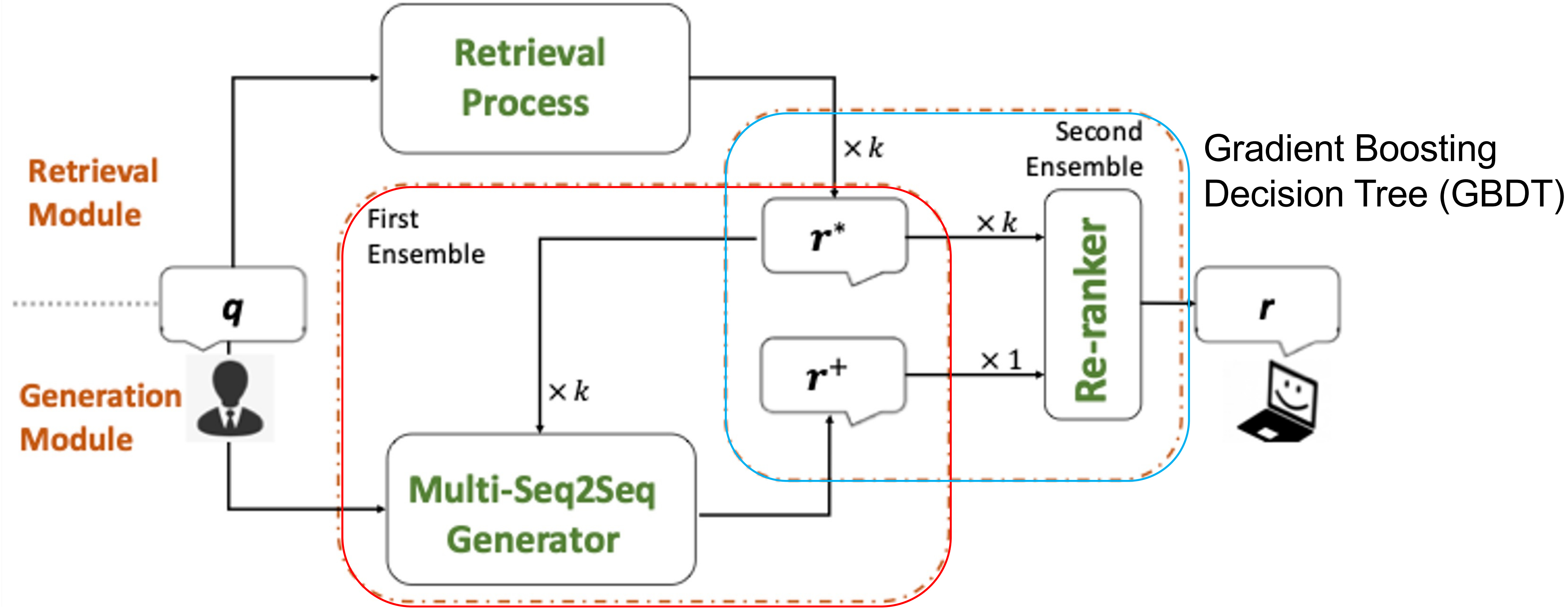
- Switch to generation-based systems when retrieval is "not good"



[Qiu + 17]

Shallow Integration of Retrieval and Generation

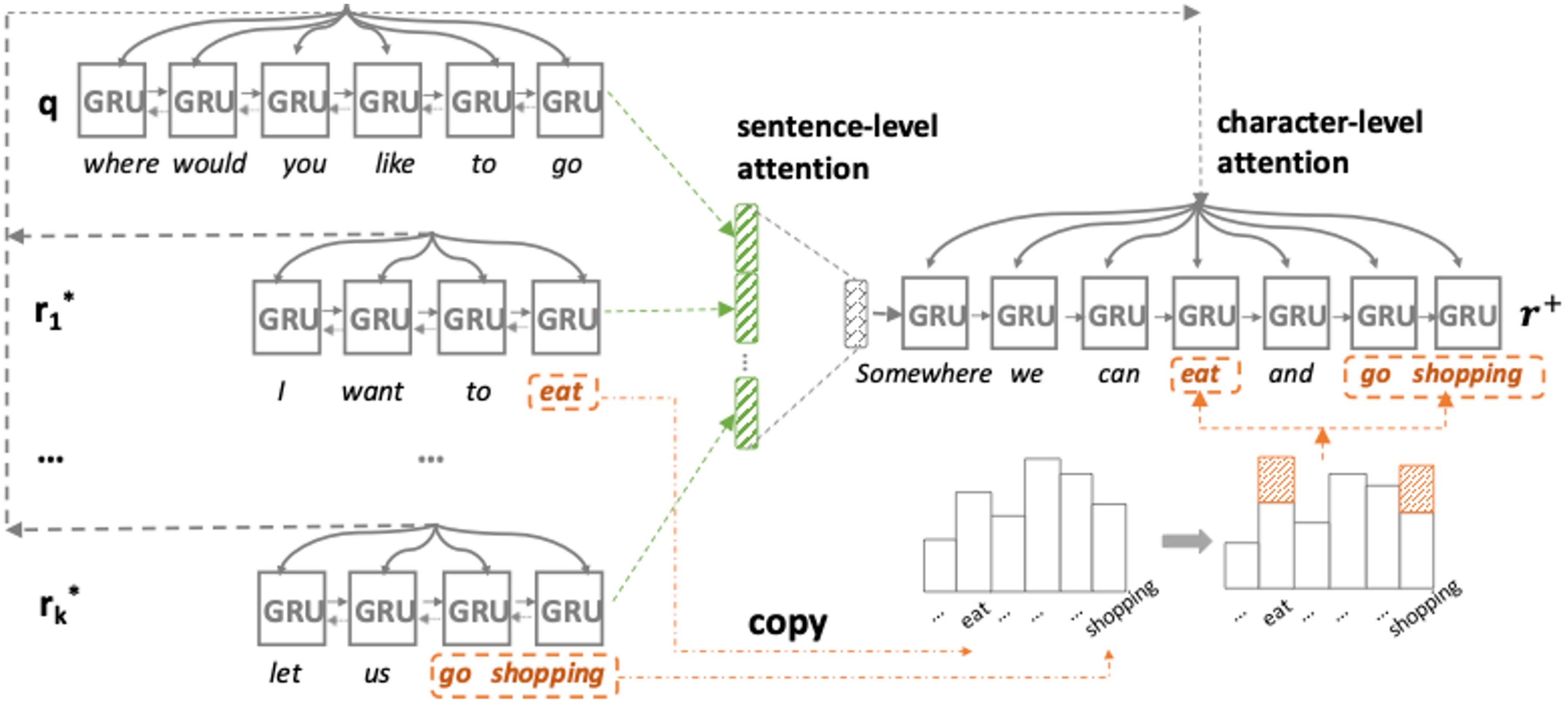
- First Ensemble: Retrieval results are fed into generation-based systems
- Second Ensemble: Rerank all produced responses (generation & retrieval)



[Song + 18]

Shallow Integration of Retrieval and Generation

- First Ensemble: Retrieval results are fed into generation-based systems
  - multi-seq2seq model



[Song + 18]

Shallow Integration of Retrieval and Generation

- Second Ensemble: Rerank all produced responses (generation & retrieval)

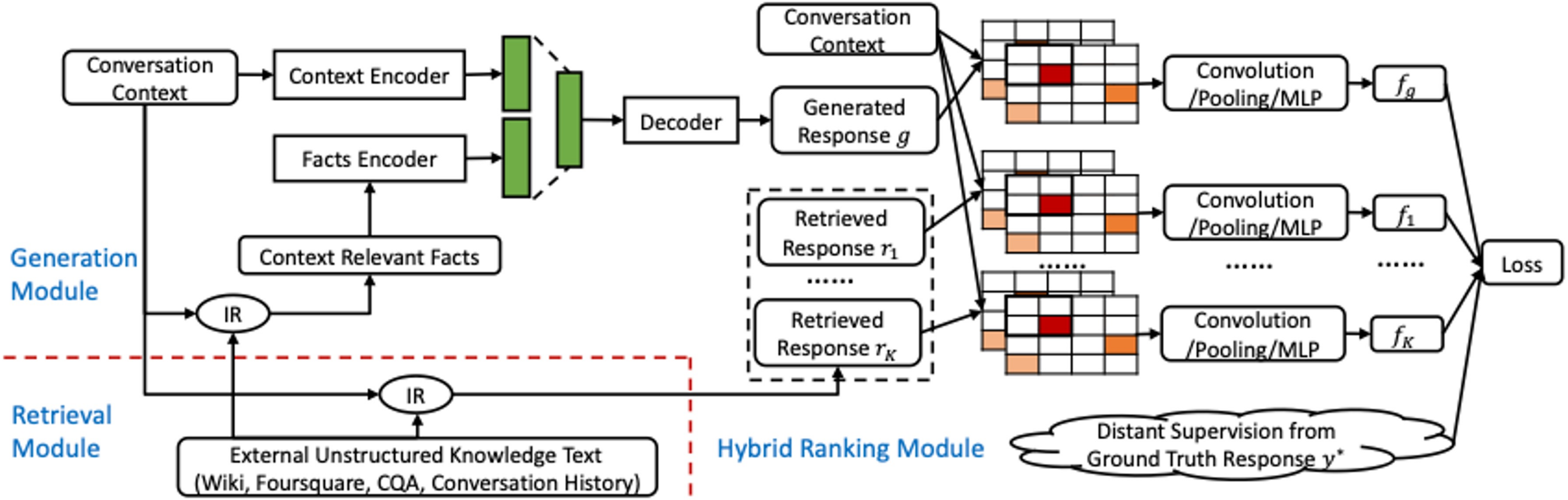
Gradient Boosting Decision Tree (GBDT)

- term similarity
- entity similarity
- topic similarity
- “translation” score
- length
- fluency

[Song + 18]

Shallow Integration of Retrieval and Generation

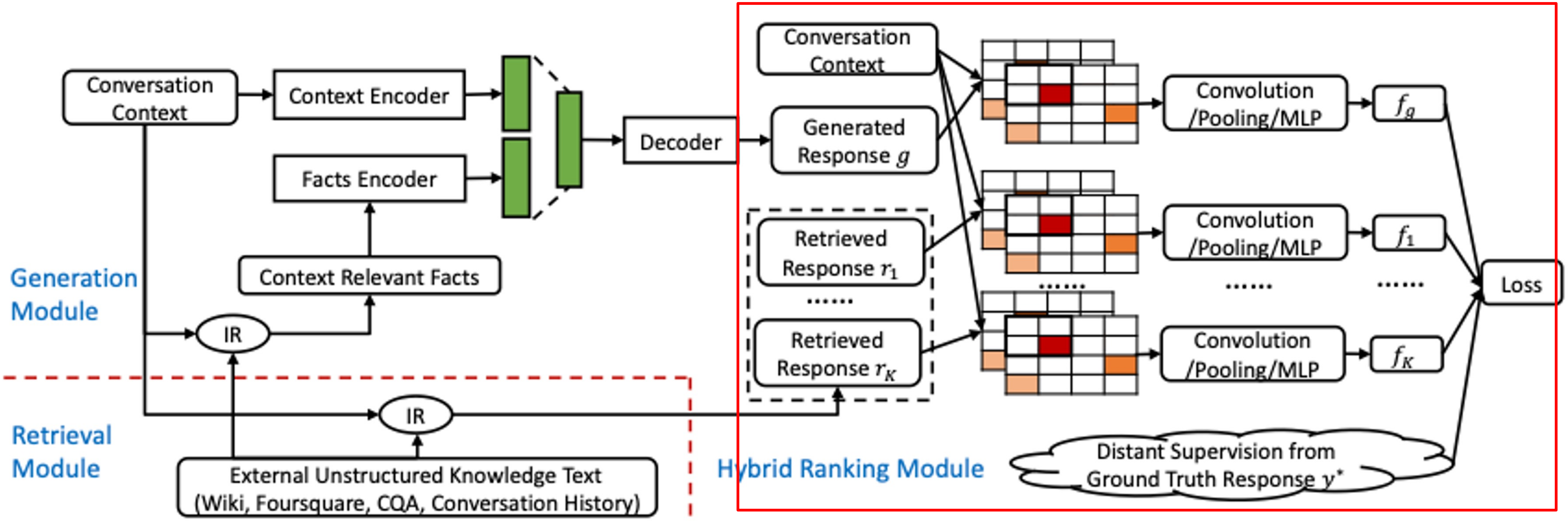
- Improving the Second Ensemble: Rerank all produced responses
  - Model: GBDT => deep neural models
  - Training Data: ground-truth/random negatives => labeled system outputs



[Yang + 19]

Shallow Integration of Retrieval and Generation

- Improving the Second Ensemble: Rerank all produced responses
  - Model: GBDT => deep neural models
  - Training Data: ground-truth/random negatives => labeled system outputs

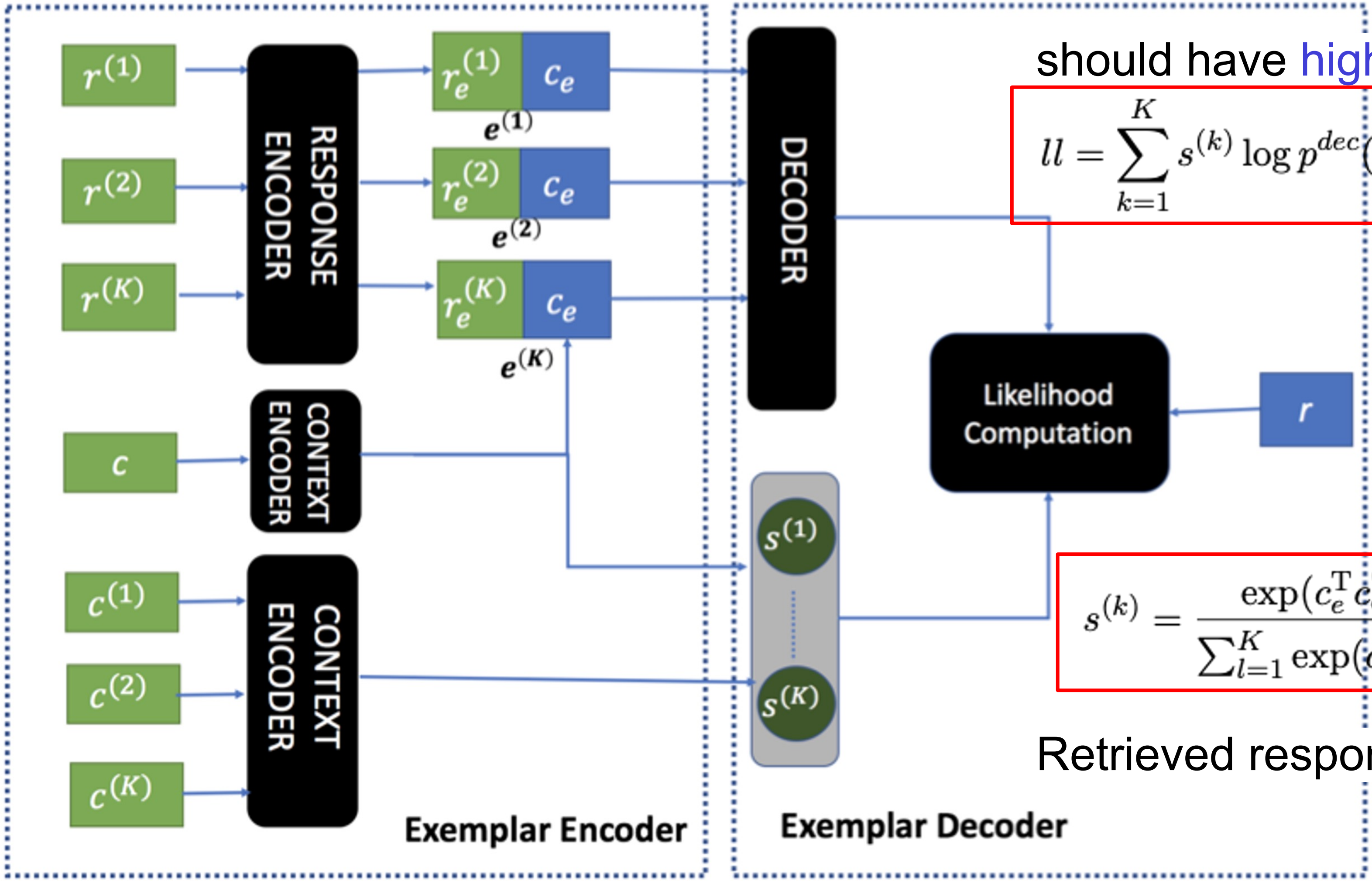


(q,r+,r−)

[Yang + 19]

Shallow Integration of Retrieval and Generation

- Improving the First Ensemble: retrieval-augmented generation



should have higher effect during generation

$$ll = \sum_{k=1}^{K} s^{(k)} \log p^{dec}(r|e^{(k)})$$

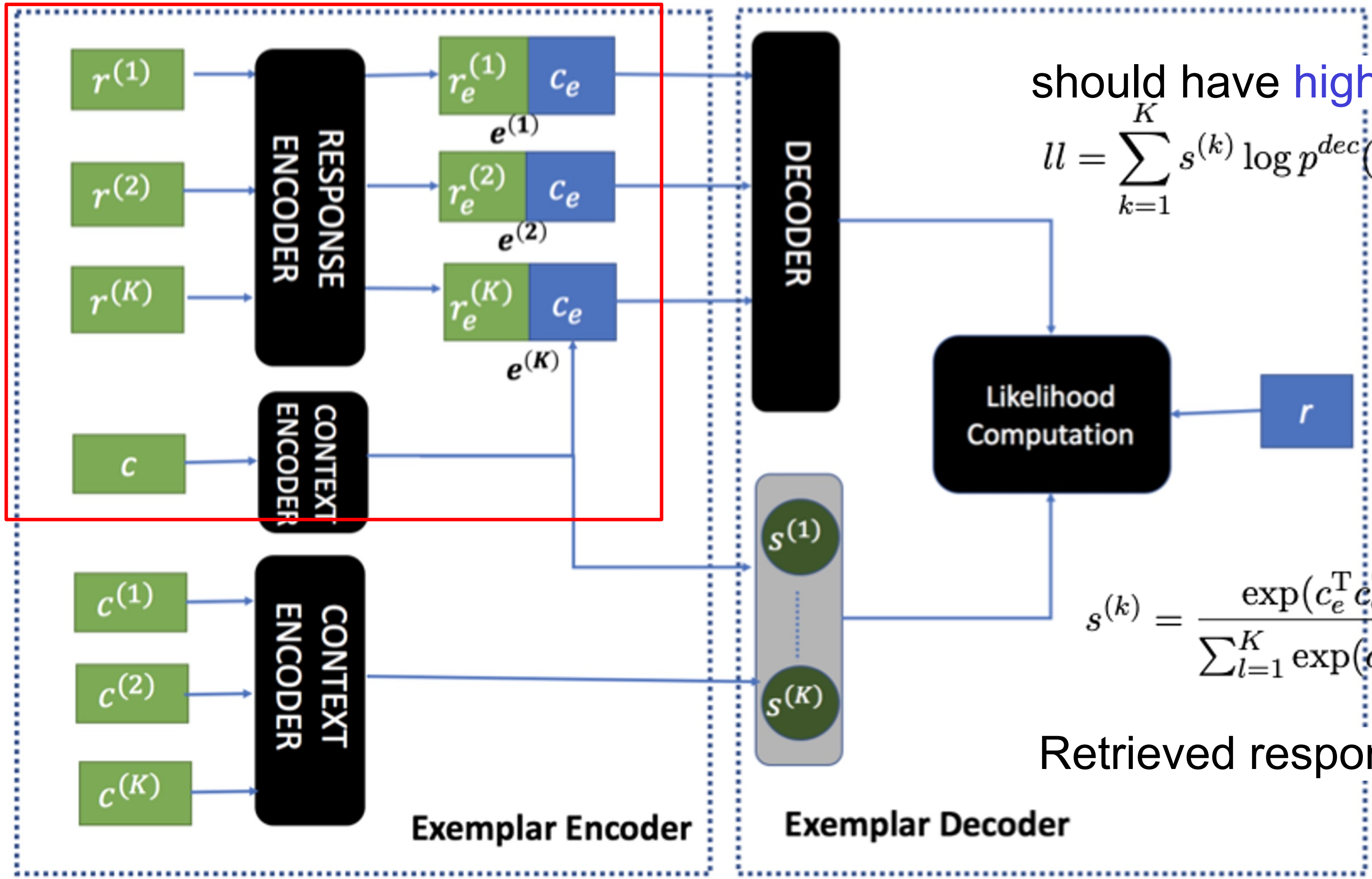
$$s^{(k)} = \frac{\exp(c_e^{\mathrm{T}} c_e^{(k)})}{\sum_{l=1}^{K} \exp(c_e^{\mathrm{T}} c_e^{(l)})}$$

Retrieved responses with more similar contexts

[Pandey+ 18]

Shallow Integration of Retrieval and Generation

- Improving the First Ensemble: retrieval-augmented generation



should have higher effect during generation

$$ll = \sum_{k=1}^{K} s^{(k)} \log p^{dec}(r|e^{(k)})$$

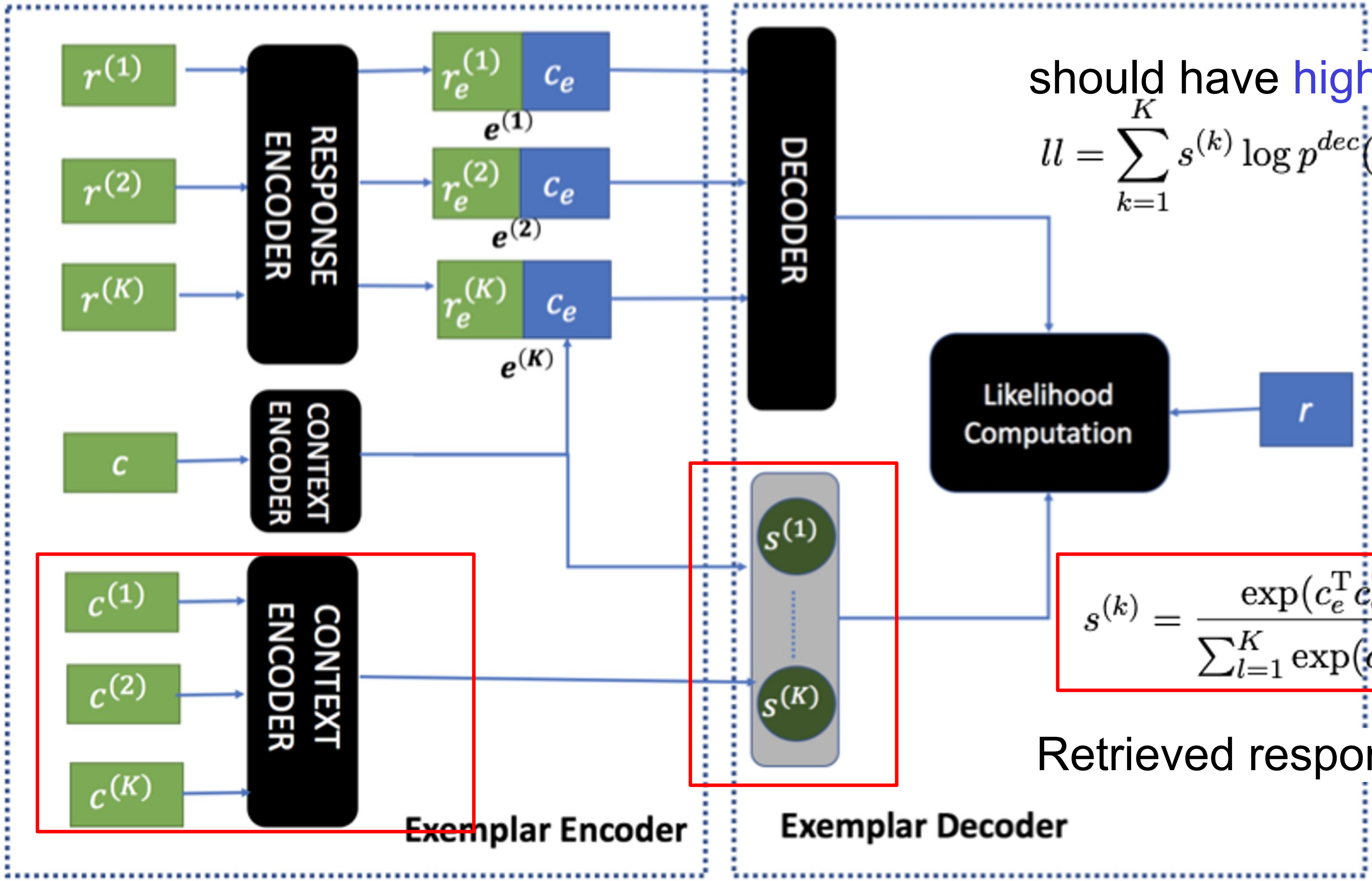
$$s^{(k)} = \frac{\exp(c_e^{\mathrm{T}} c_e^{(k)})}{\sum_{l=1}^{K} \exp(c_e^{\mathrm{T}} c_e^{(l)})}$$

Retrieved responses with more similar contexts

[Pandey+ 18]

Shallow Integration of Retrieval and Generation

- Improving the First Ensemble: retrieval-augmented generation



should have higher effect during generation

$$ll = \sum_{k=1}^{K} s^{(k)} \log p^{dec}(r|e^{(k)})$$

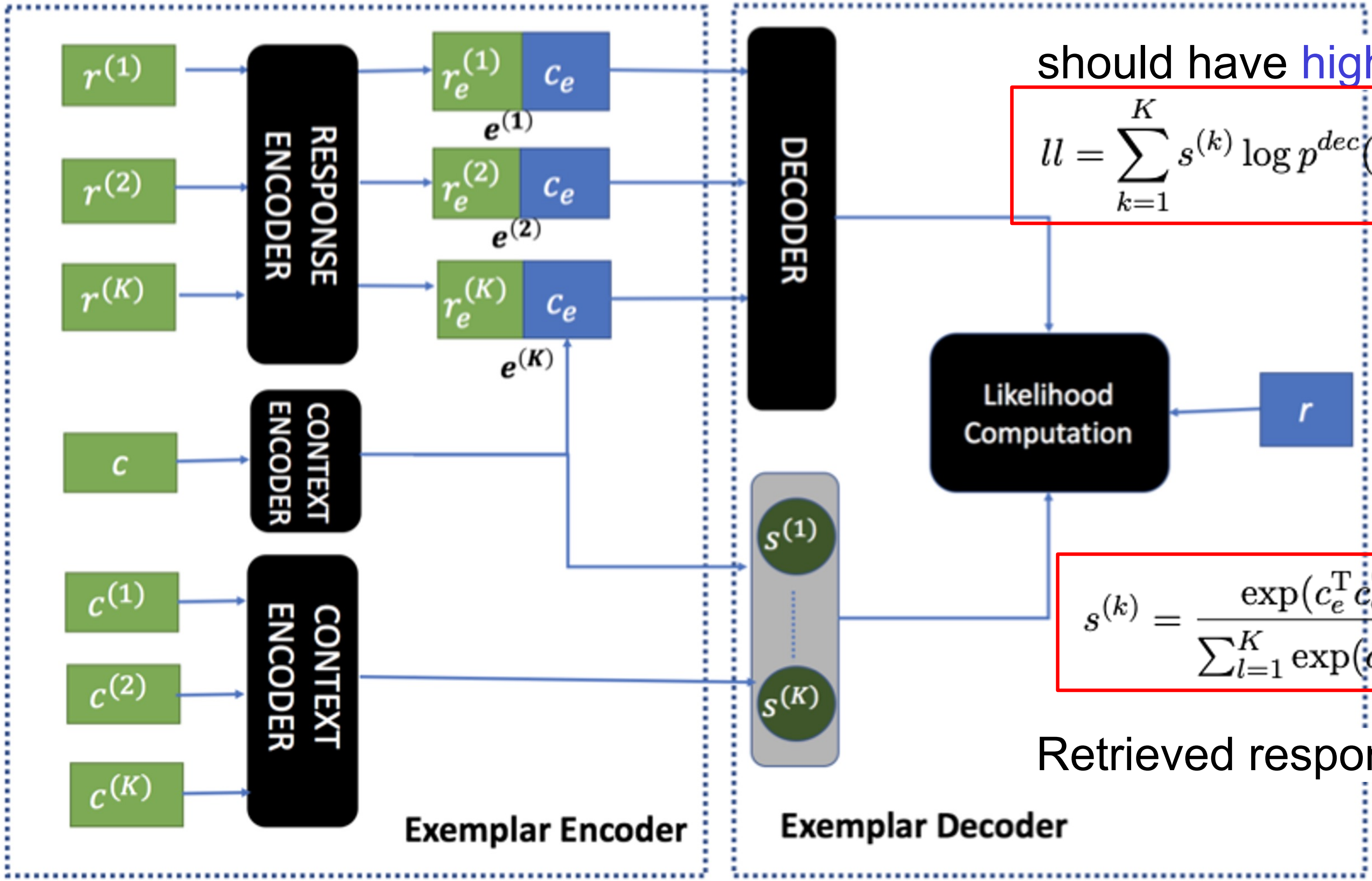
$$s^{(k)} = \frac{\exp(c_e^{\mathrm{T}} c_e^{(k)})}{\sum_{l=1}^{K} \exp(c_e^{\mathrm{T}} c_e^{(l)})}$$

Retrieved responses with more similar contexts

[Pandey+ 18]

Shallow Integration of Retrieval and Generation

- Improving the First Ensemble: retrieval-augmented generation



should have higher effect during generation

$$ll = \sum_{k=1}^{K} s^{(k)} \log p^{dec}(r|e^{(k)})$$

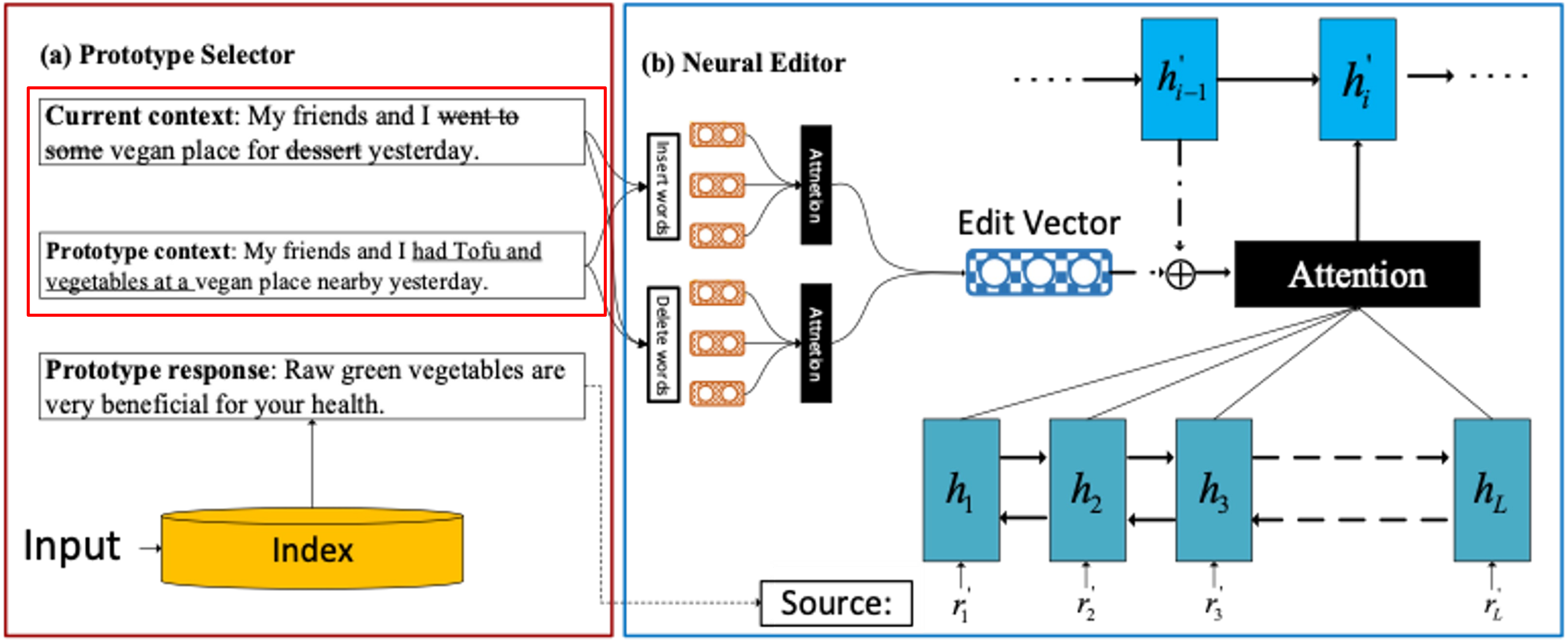
$$s^{(k)} = \frac{\exp(c_e^{\mathrm{T}} c_e^{(k)})}{\sum_{l=1}^{K} \exp(c_e^{\mathrm{T}} c_e^{(l)})}$$

Retrieved responses with more similar contexts

[Pandey+ 18]

Shallow Integration of Retrieval and Generation

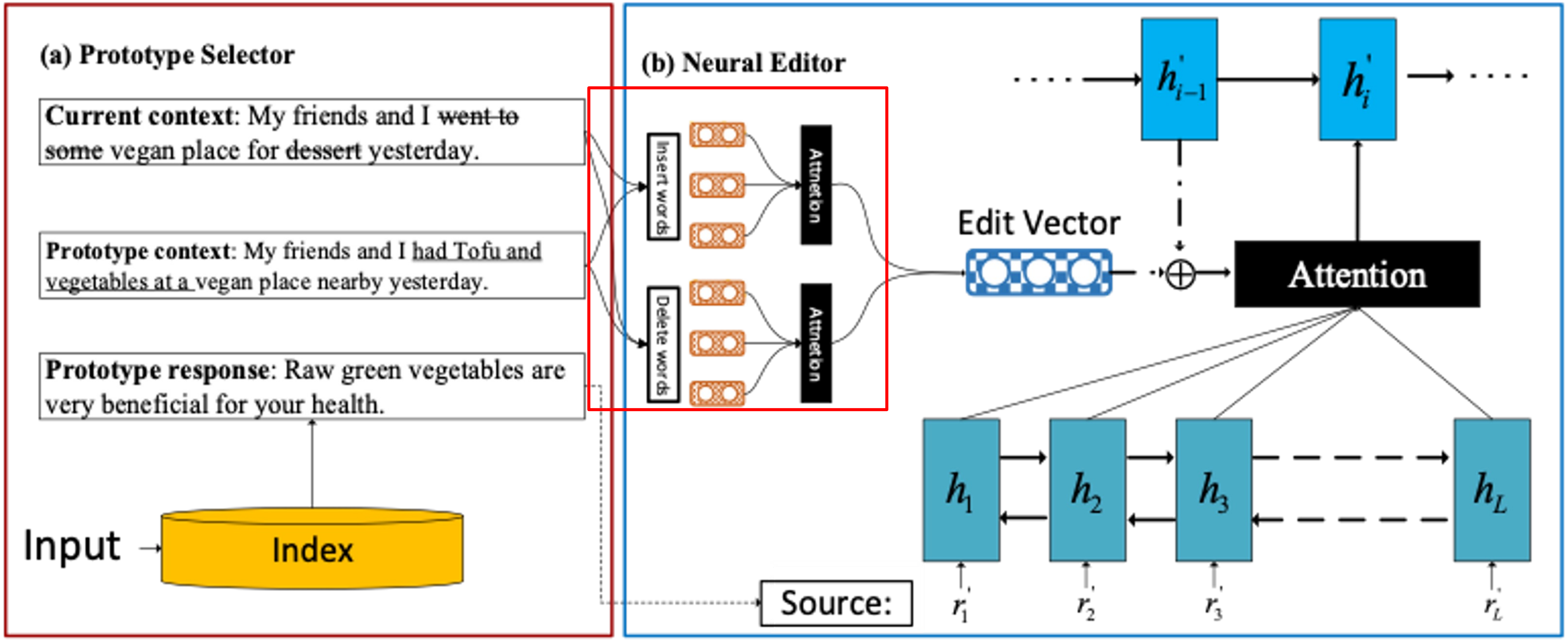
- Improving the First Ensemble: retrieval-augmented generation
  - Differences in **contexts** provide an important signal for differences in **responses**.



[Wu + 19]

Shallow Integration of Retrieval and Generation

- Improving the First Ensemble: retrieval-augmented generation
  - Differences in **contexts** provide an important signal for differences in **responses**.



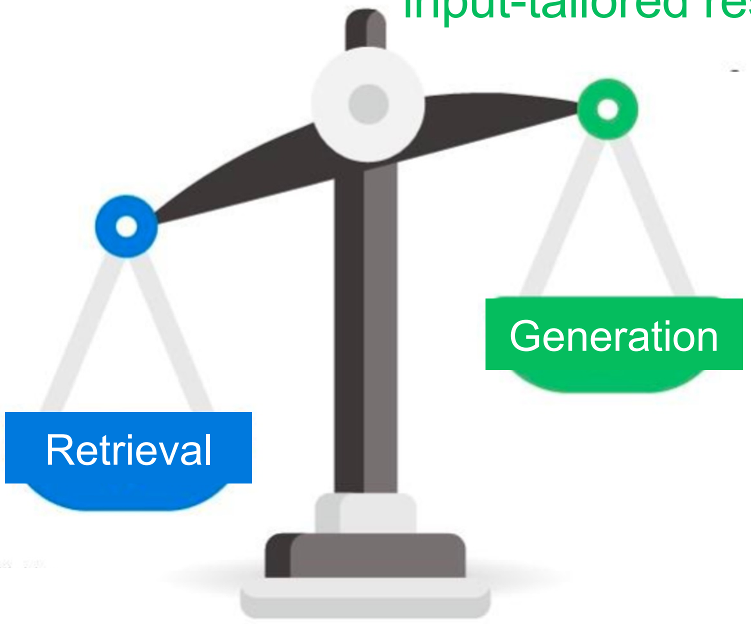
[Wu + 19]

Problems when Integrating Retrieval and Generation

- Collapsing to the ordinary retrieval system

when the retrieval is generally good

lose the ability to make input-tailored responses



overly rely on retrieval
even copy irrelevant content

Filter out irrelevant content from retrieval

The retrieved responses typically contain excessive information, including inappropriate words or entities. It is necessary to filtered out irrelevant content.

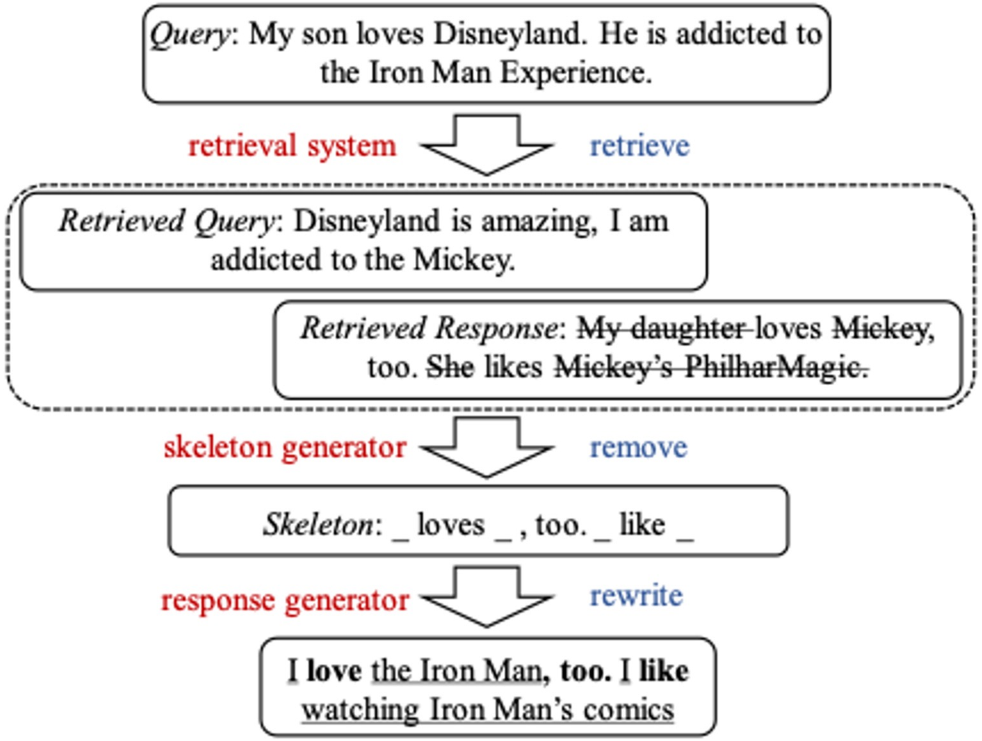
Maintain the generalizability of generation

The guidance from retrieval should only specify a response pattern or provide some information, but leave the details to be elaborated by the generation model.

Deep Integration of Retrieval and Generation

- Retrieve-Remove-Rewrite
  - extracting **response skeleton**

explicitly control the information inflow

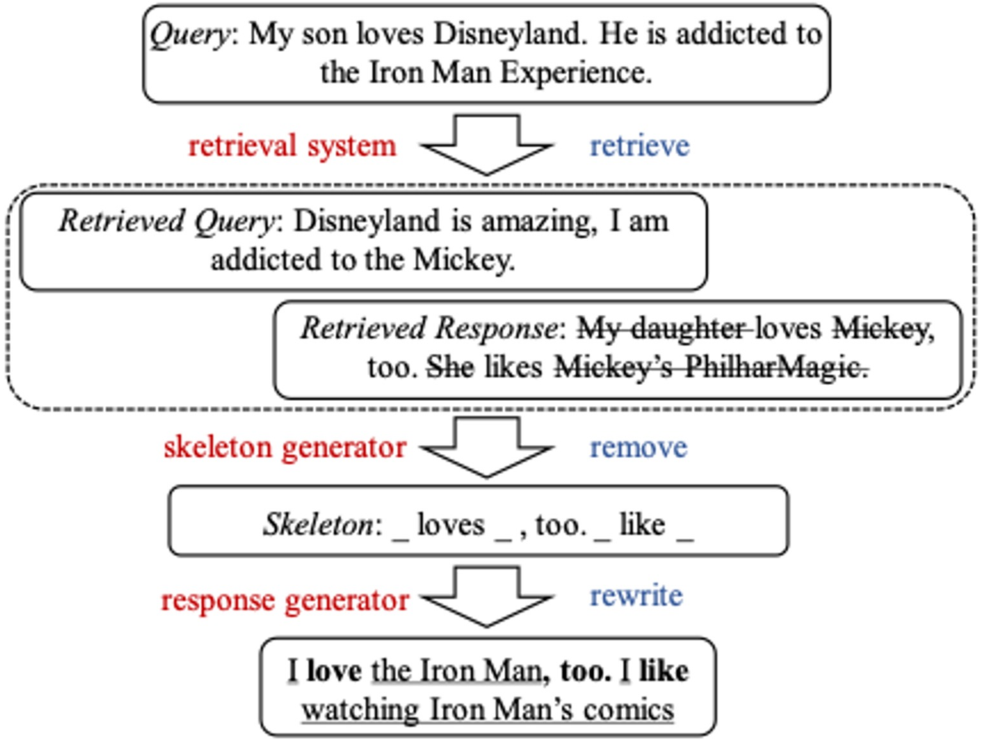


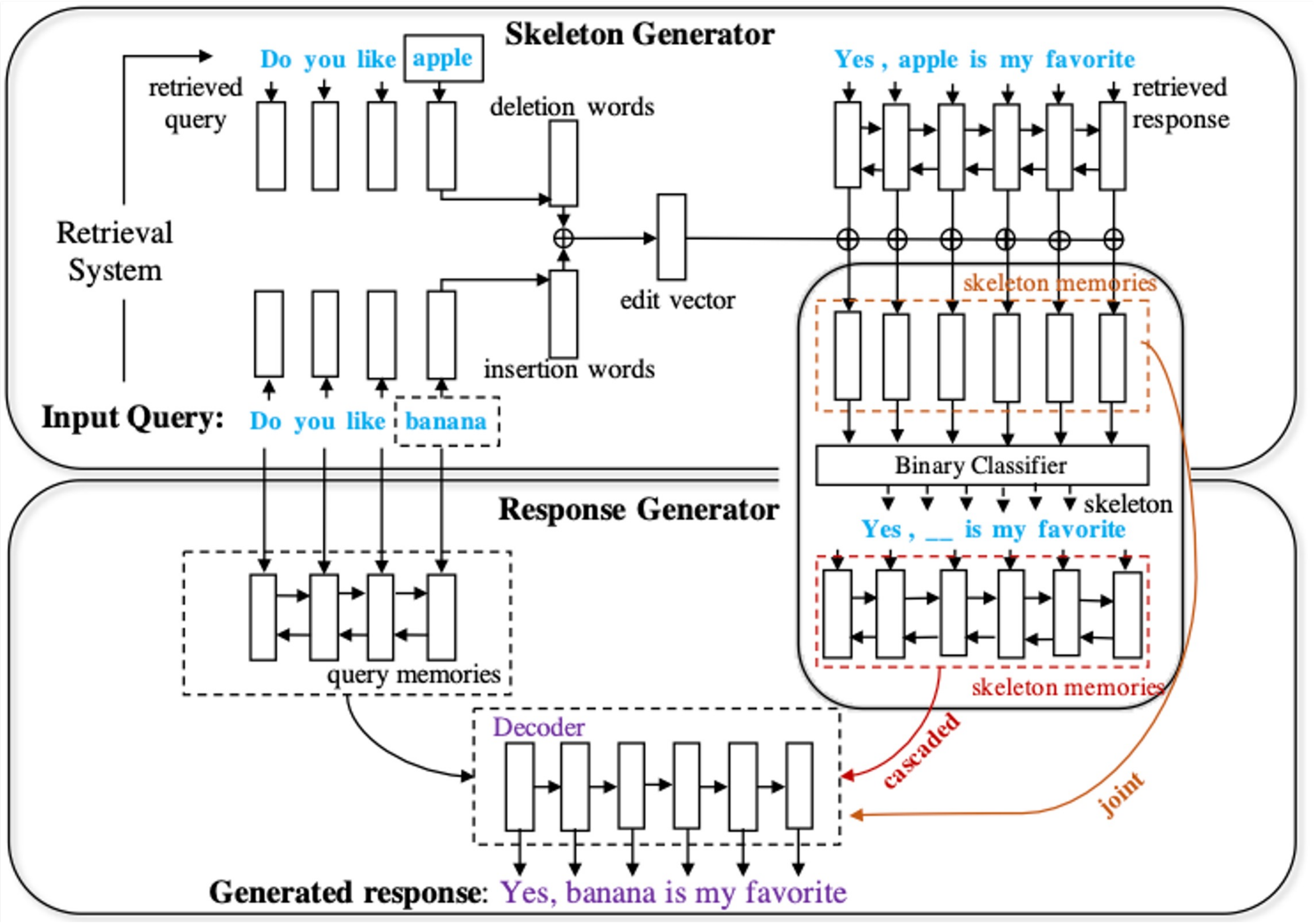
[Cai + 19]

Deep Integration of Retrieval and Generation

- Retrieve-Remove-Rewrite
  - extracting **response skeleton**

explicitly control the information inflow





[Cai + 19]

Deep Integration of Retrieval and Generation

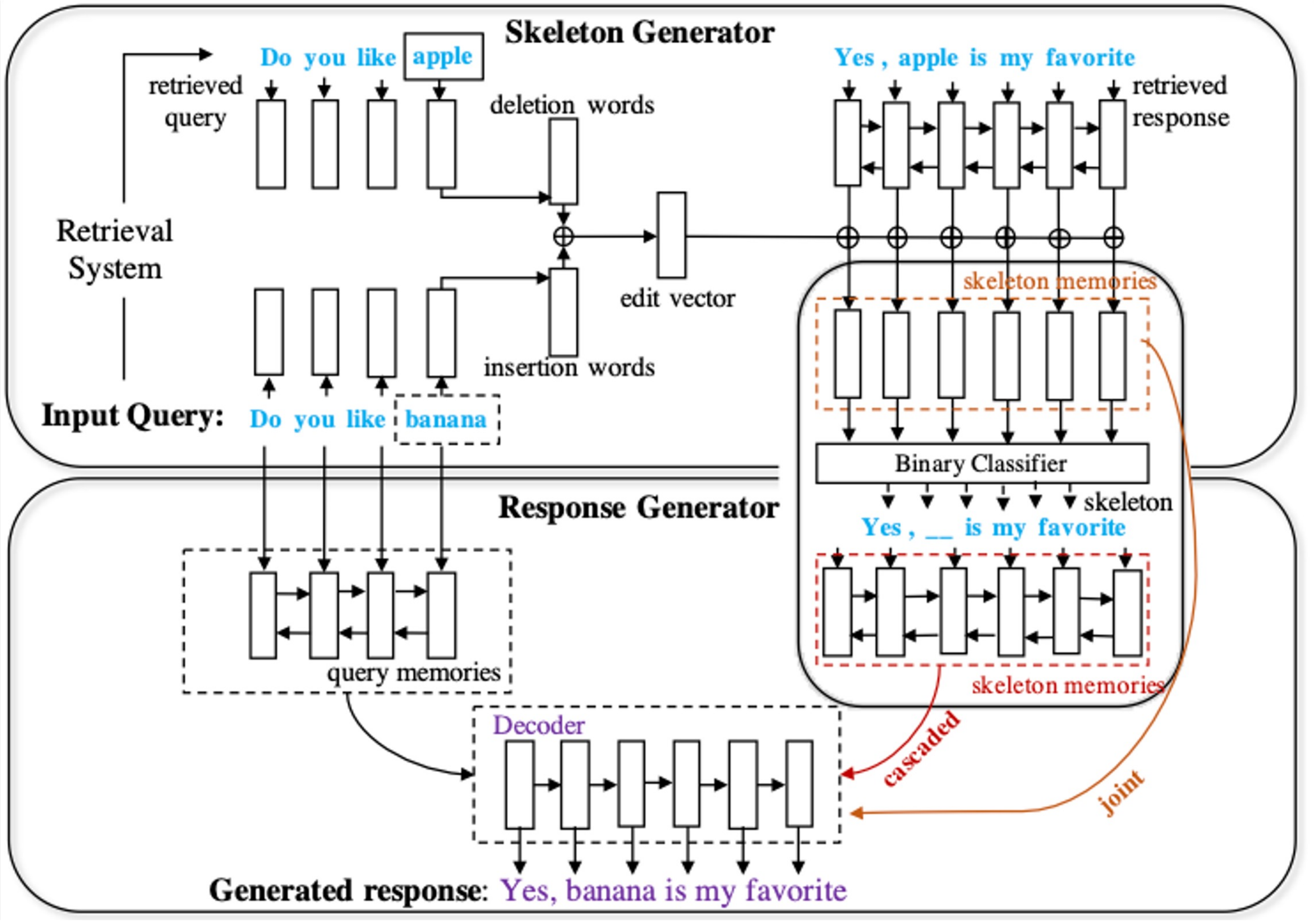
- Retrieve-Remove-Rewrite
  - extracting **response skeleton**

explicitly control the information inflow

Definition 1 Proxy Skeleton: *Given a training quadruplet* (q, q', r, r') *and a stop word list* S*, the proxy skeleton for* r *is generated by replacing some tokens in* r' *with a placeholder "<blank>". A token* r'_i *is kept if and only if it meets the following conditions*

1. $r'_i \notin S$

2. r'_i *is a part of the longest common subsequence (LCS) (Wagner and Fischer, 1974) of* r *and* r'.



[Cai + 19]

Deep Integration of Retrieval and Generation

- Retrieve-Remove-Rewrite
  - extracting **response skeleton**

explicitly control the information inflow

First RL Agent: Skeleton Generator

Second RL Agent: Response Generator

Reward Function: a pre-trained critic *D*

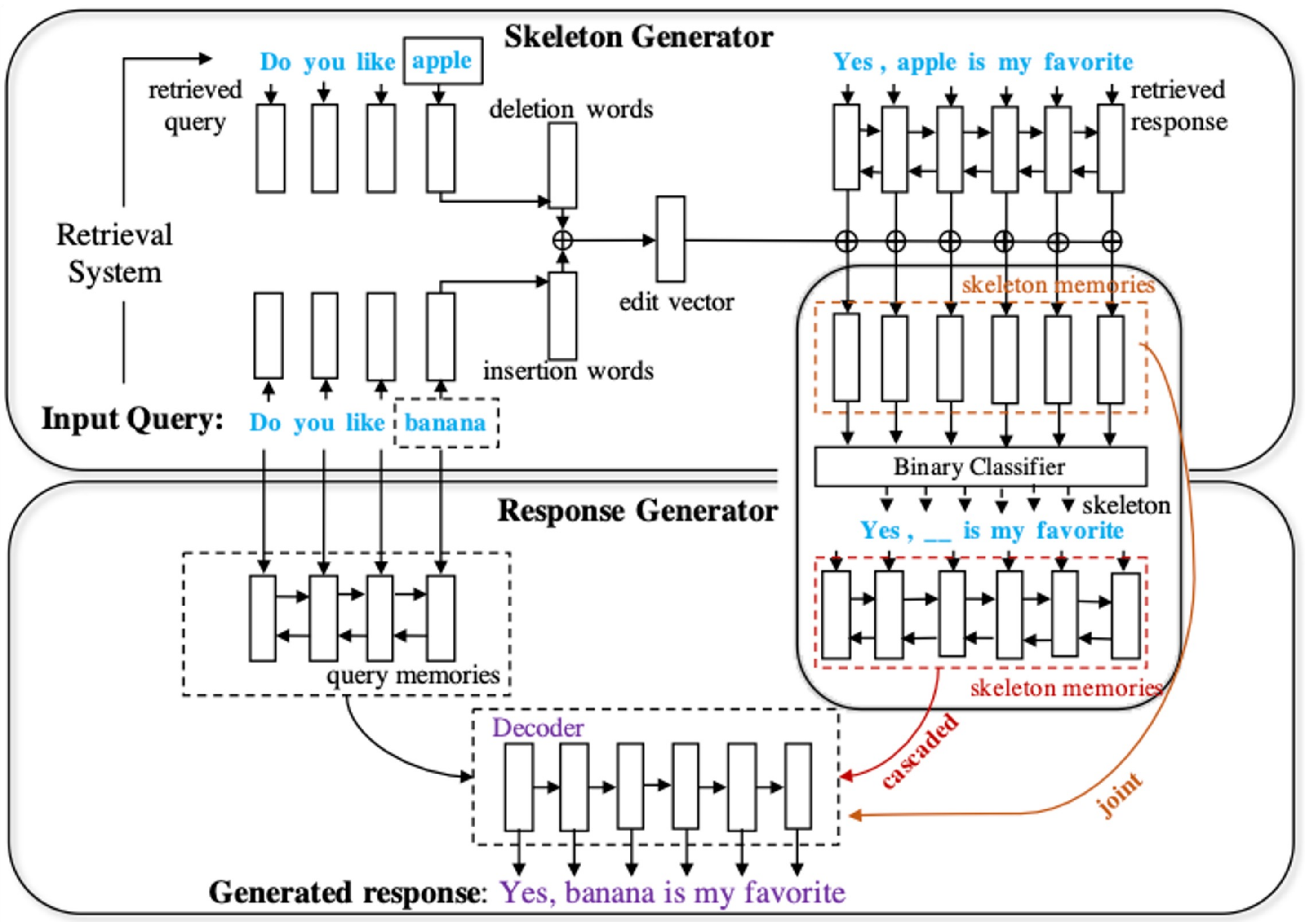
$$\log D(r|q,\hat{r},\bar{r},r) = \log \frac{\exp({h_r}^{\mathrm{T}} M_D h_q)}{\sum_{x\in\{\hat{r},\bar{r},r\}} \exp({h_x}^{\mathrm{T}} M_D h_q)}$$

query

ground-truth

machine-generated

random



[Cai + 19]

Deep Integration of Retrieval and Generation

- Retrieve-Abstract-Follow
  - extracting **semantic structure**

preserve the semantic structure

avoid over-reliant on copying (inappropriate) words

Context	My friends and I have started eating vegan food since yesterday.
Exemplar **Frames** **Responses**	Eggs are very beneficial for your body. FOOD USEFULNESS BODY-PARTS Vegan food can be good for your health. Vegetables can do wonders for your body Vegan food is very healthy.
Exemplar **Frames** **Responses**	I want to drink milk as well. DESIRING INGESTION FOOD You want to eat some vegan food? We eat a lot of vegetables. It's delicious. We like to eat organic food.

[Gupta + 21]

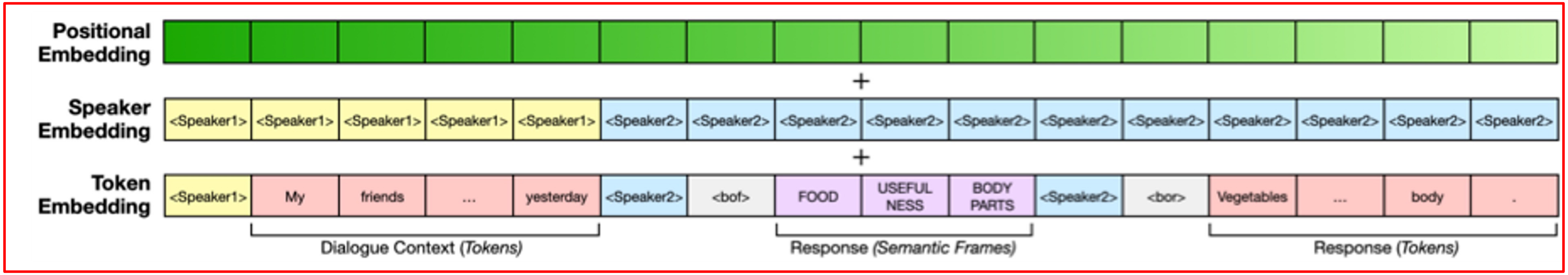
Deep Integration of Retrieval and Generation

- Retrieve-Abstract-Follow
  - extracting **semantic structure**

preserve the semantic structure

avoid over-reliant on copying (inappropriate) words

Context	My friends and I have started eating vegan food since yesterday.
Exemplar	Eggs are very beneficial for your body.
Frames	FOOD USEFULNESS BODY-PARTS
Responses	Vegan food can be good for your health. Vegetables can do wonders for your body Vegan food is very healthy.
Exemplar	I want to drink milk as well.
Frames	DESIRING INGESTION FOOD
Responses	You want to eat some vegan food? We eat a lot of vegetables. It's delicious. We like to eat organic food.



[Gupta + 21]

Deep Integration of Retrieval and Generation

Model	Dist-2	Dist-3	MaUdE	Coherent	Fluent	Consistent	Interesting
Retrieval	0.294	0.526	0.921	2.41	2.61	2.48	2.32
GPT2-Gen	0.249	0.494	0.905	2.42	2.55	2.41*	2.18*
LSTM-Tokens	0.182	0.380	0.890	2.04*	2.10*	2.11*	1.89*
LSTM-Frames	0.185	0.392	0.901	2.36*	2.30*	2.33*	1.97*
GPT2-Tokens	0.254	0.513	**0.927**	2.19*	2.47*	2.29*	2.11*
EDGE (Ours)	**0.278**	**0.571**	0.922	**2.52**	**2.63**	**2.56**	**2.39**
Human	0.385	0.720	0.911	2.76	2.69	2.78	2.44

Context	*Human1*: they sell everything. *Human2*: well, i want chinese food.	*Human1*: actually i have a passion for chinese literature. *Human2*: you do?
Retrieved	well, what do you want to eat?	yes, reading is my hobby.
Frames	WHAT DESIRING INGESTION ?	YES LINGUISTIC-MEANING
GPT2-Gen	it's a good idea.	yes. i'm passionate.
LSTM-Tokens	well, what's the you do?	yes, i do.
LSTM-Frames	i hope so.	yes, i did.
GPT2-Tokens	i'm not sure what to get.	what are you interested in?
EDGE (Ours)	you want to eat something chinese?	yes. i studied chinese literature at university.

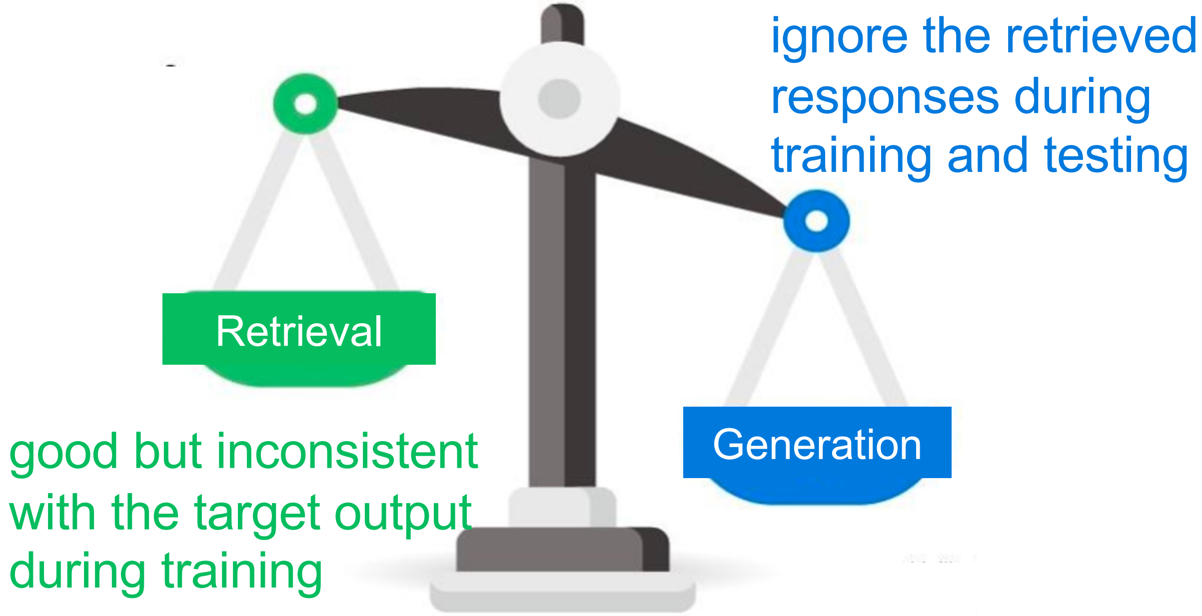
[Gupta + 21]

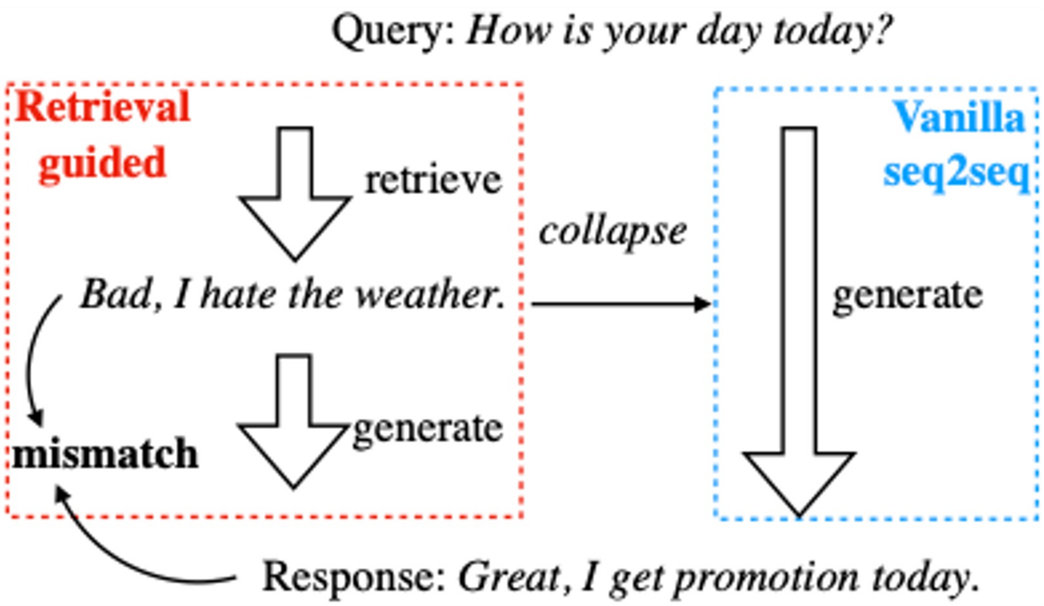
Problems when Integrating Retrieval and Generation

- **Collapsing** to the ordinary generation system

inconsistent context-retrieval-response triples for training

context-relevant ≠ response-relevant





Deep Integration of Retrieval and Generation

- Response-consistent skeletons generated automatically from the target response
- Accurate skeleton extraction with distant supervision from semantic matching

context

How is your day today?

+

Training Triples

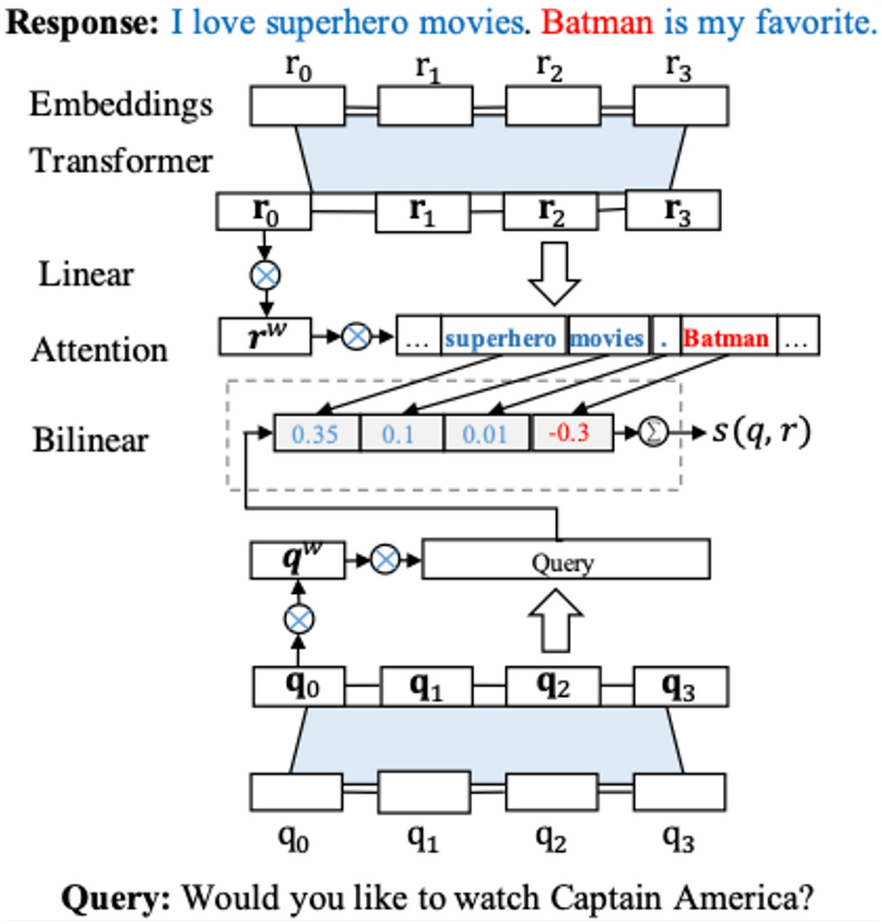
response

Great! I get promotion today.

Great! I get _ today.
_! _ _ promotion _.
I promotion _ Great!
Great! I _ _ tomorrow.
...

skeleton

mask, shuffle, replace

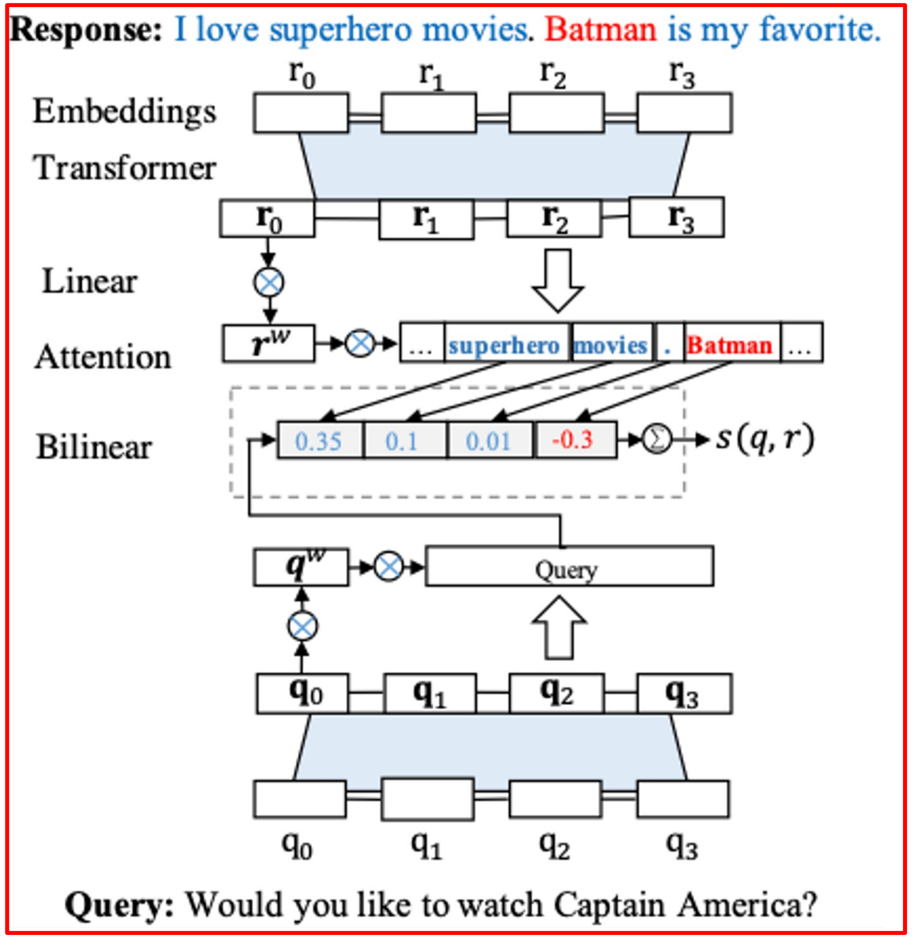


[Cai + 19]

Deep Integration of Retrieval and Generation

- Response-consistent skeletons generated automatically from the target response
- Accurate skeleton extraction with distant supervision from semantic matching

$$s(q,r) = \mathbf{x}_q{}^T W^s \boxed{\mathbf{x}_r}$$
$$= \mathbf{x}_q{}^T W^s \boxed{\sum_{k=1}^{m} \omega_k (\mathbf{r}_k + \mathbf{e}_{r_k})}$$



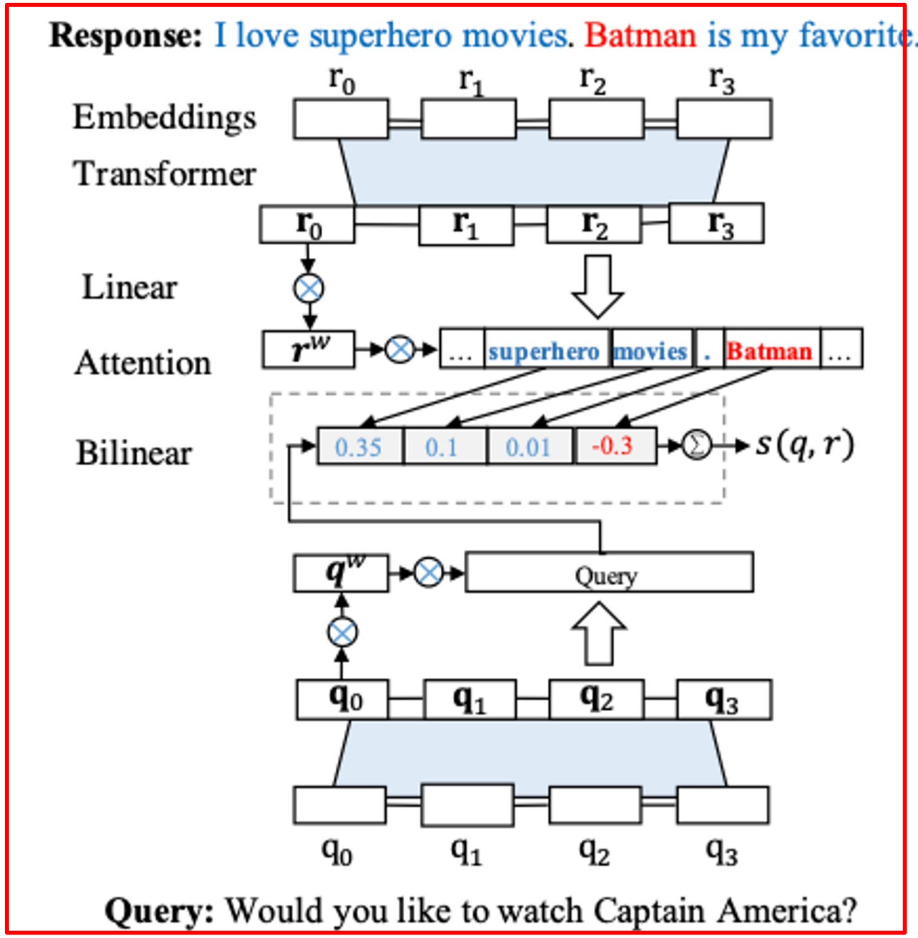
[Cai + 19]

Deep Integration of Retrieval and Generation

- Response-consistent skeletons generated automatically from the target response
- Accurate skeleton extraction with distant supervision from semantic matching

$$
\begin{aligned}
s(q, r) &= \mathbf{x}_q{}^T W^s \boxed{\mathbf{x}_r} \\
&= \mathbf{x}_q{}^T W^s \boxed{\sum_{k=1}^{m} \boxed{\omega_k} (\mathbf{r}_k + \mathbf{e}_{r_k})}
\end{aligned}
$$

weights   token embddings



[Cai + 19]

Deep Integration of Retrieval and Generation

- Response-consistent skeletons generated automatically from the target response
- Accurate skeleton extraction with distant supervision from semantic matching

$$s(q,r) = \mathbf{x}_q{}^T W^s \boxed{\mathbf{x}_r}$$

$$= \mathbf{x}_q{}^T W^s \boxed{\sum_{k=1}^{m} \boxed{\omega_k}(\mathbf{r}_k + \mathbf{e}_{r_k})} = \sum_{k=1}^{m} \omega_k \mathbf{x}_q{}^T W^s (\mathbf{r}_k + \mathbf{e}_{r_k})$$

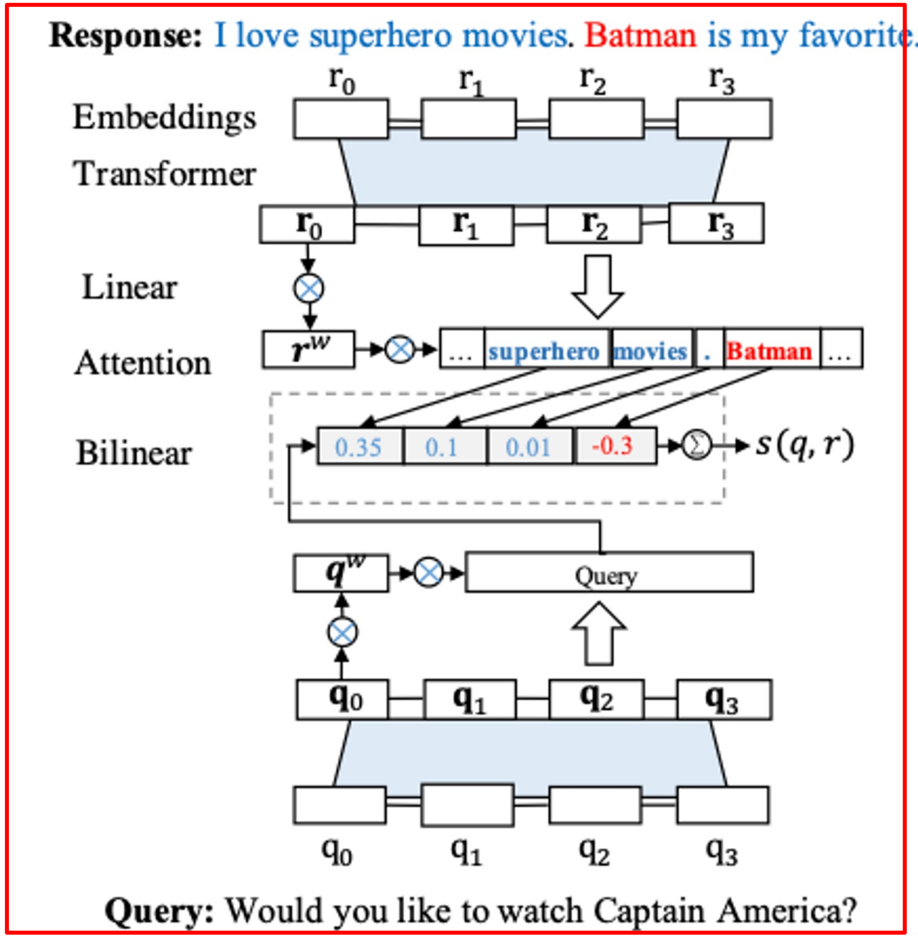
weights

token embddings

Let $s_k = \mathbf{x}_q{}^T W^s (\mathbf{r}_k + \mathbf{e}_{r_k})$, we arrive at:

$$s(q,r) = \sum_{k=1}^{m} \omega_k \boxed{s_k}$$

local matching scores



[Cai + 19]

Deep Integration of Retrieval and Generation

- Improve the best of two worlds:
  - Higher informativeness than vanilla retrieval
  - Higher relevance than vanilla generation

Models	Informativeness	Relevance	Fluency
Retrieval	2.65 (0.90)†	2.58 (0.86)	2.96 (0.72)
Seq2Seq	2.01 (0.65)	2.58 (0.53)	2.71 (0.43)
Seq2Seq-MMI	2.47 (0.70)	2.79 (0.67)	2.99 (0.61)
RetrieveNRefine^{++}	2.30 (0.79)	2.62 (0.63)	2.82 (0.51)
EditVec	2.29 (0.61)	2.62 (0.60)	2.83 (0.47)
Skeleton-Lex	2.45 (0.61)	2.80 (0.56)	2.99 (0.46)
Ours	**2.69** (0.87)	**3.11** (0.55)	**3.20** (0.55)

[Cai + 19]

Deep Integration of Retrieval and Generation

- Model response-posterior distribution

$$P(y|x) = \sum_{z \in \text{top-k}(P_\eta(.|x))} \underbrace{P_\eta(z|x)}_{\text{retriever}} \underbrace{P_\theta(y|x, z)}_{\text{generator}}$$

context-relevant ≠ response-relevant

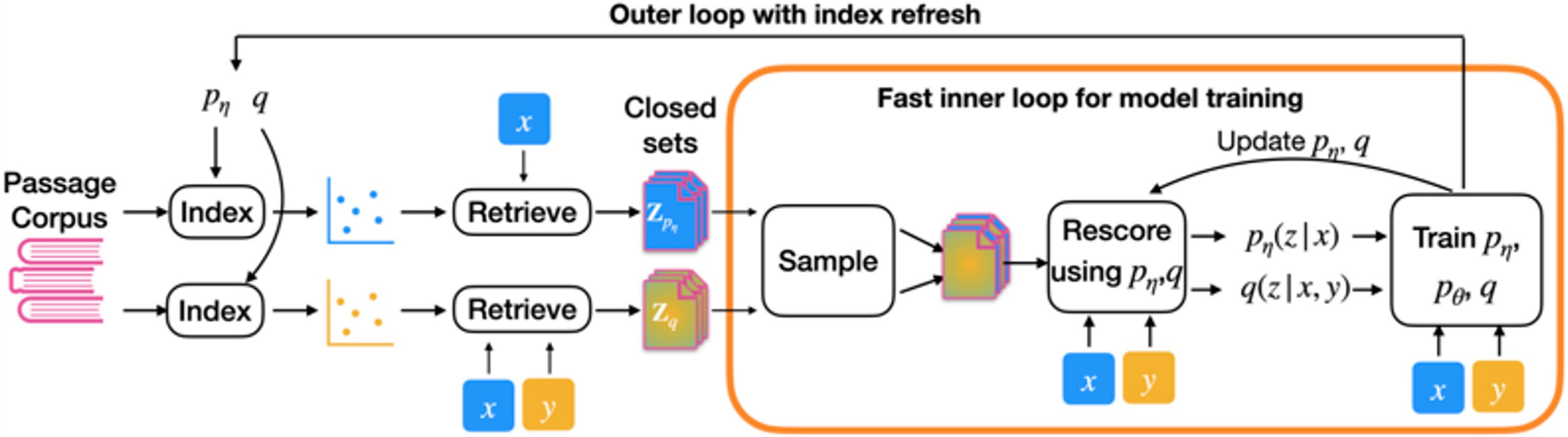
[Paranjape + 21]

Deep Integration of Retrieval and Generation

- Model response-posterior distribution

$$P(y|x) = \sum_{z \in \text{top-k}(P_\eta(.|x))} \underbrace{P_\eta(z|x)}_{\text{retriever}} \underbrace{P_\theta(y|x,z)}_{\text{generator}} \Rightarrow \log P(y|x) \geq \underbrace{\boxed{\mathbb{E}_{z \sim Q(.|x,y)}[\log P_\theta(y|x,z)]}}_{\text{response-posterior}} - \boxed{D_{\mathrm{KL}}(Q|P_\eta)}$$

- differentiate response-relevant from other context-relevant retrieval
- encourage the retriever to trust response-relevant



[Paranjape + 21]

Takeaways

- Retrieval helps generation in open-domain dialogues
  - promote **informativeness** and **relevance**
  - provide **explainability** and **controllability**
- but... should be used with caution for the following problems
  - Information overflow (**overly rely on retrieval**)
  - Inconsistent context-retrieval-response training triples (**ignore retrieval**)

Outline

- Background and Introduction
- Language Modeling
- Open-Domain Dialogue Systems
- **Neural Machine Translation**
  - Motivation
  - TM-augmented NMT Framework
  - TM-augmented Models
    - Standard model
    - Dual model
    - Unified model
- Conclusion and Outlook

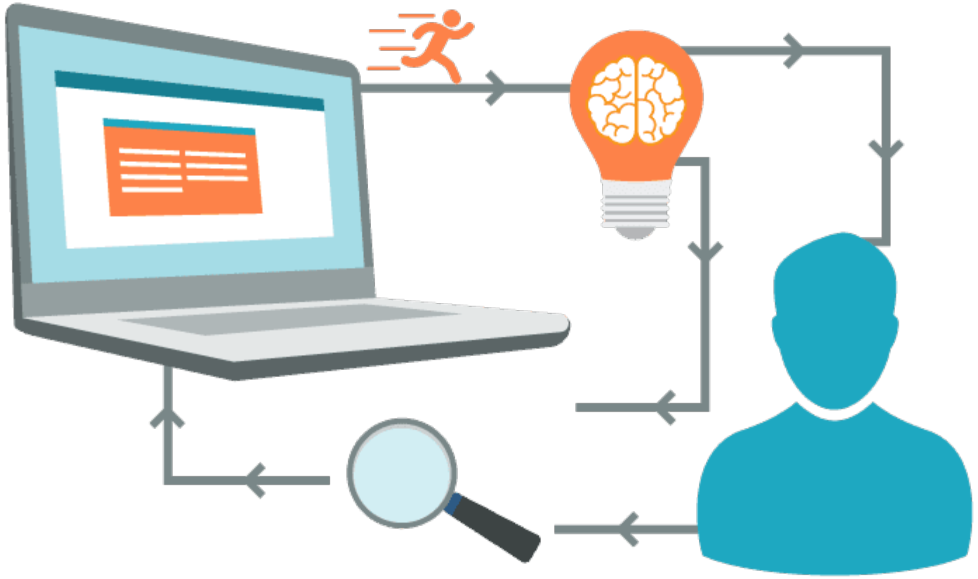
Why retrieval is beneficial to translation?

x	huoqu huo shezhi yu pizhu guanlian de duixiang 获取 或 设置 与 批注 关联 的 对象

↓

↓

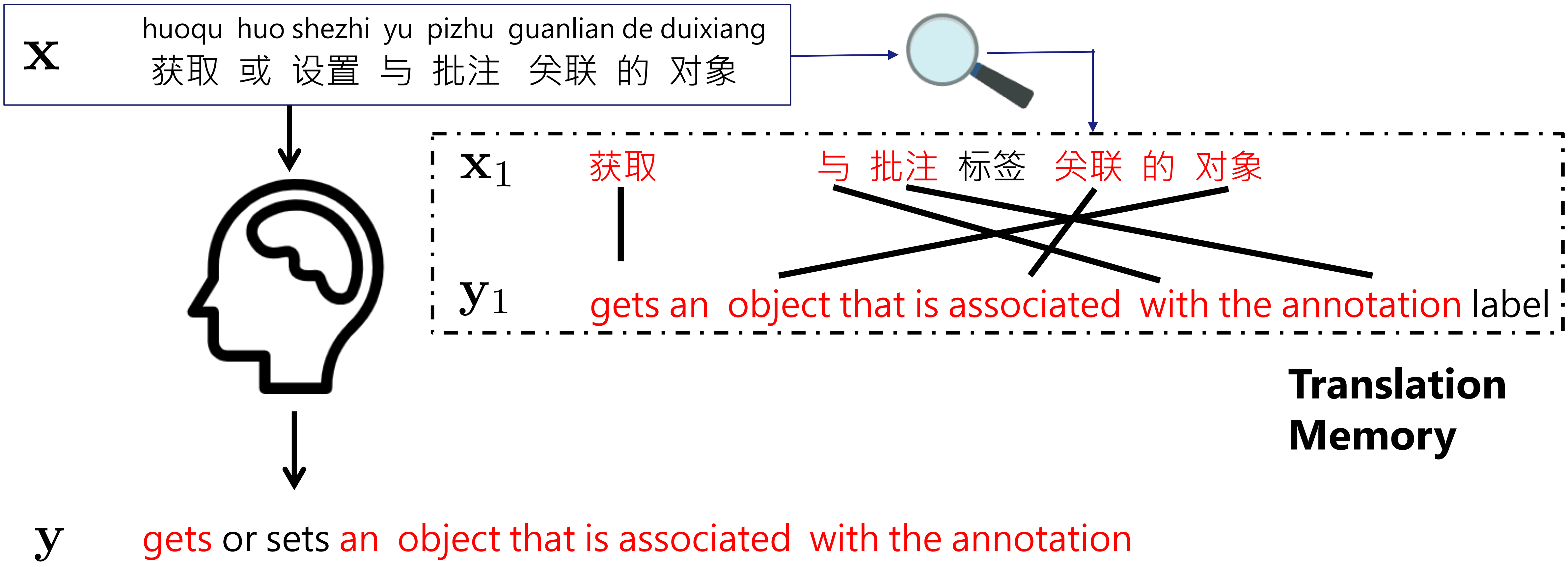
y ?



Retrieval for translation is called translation memory (TM)
TM originated from human translation scenario in 1970s

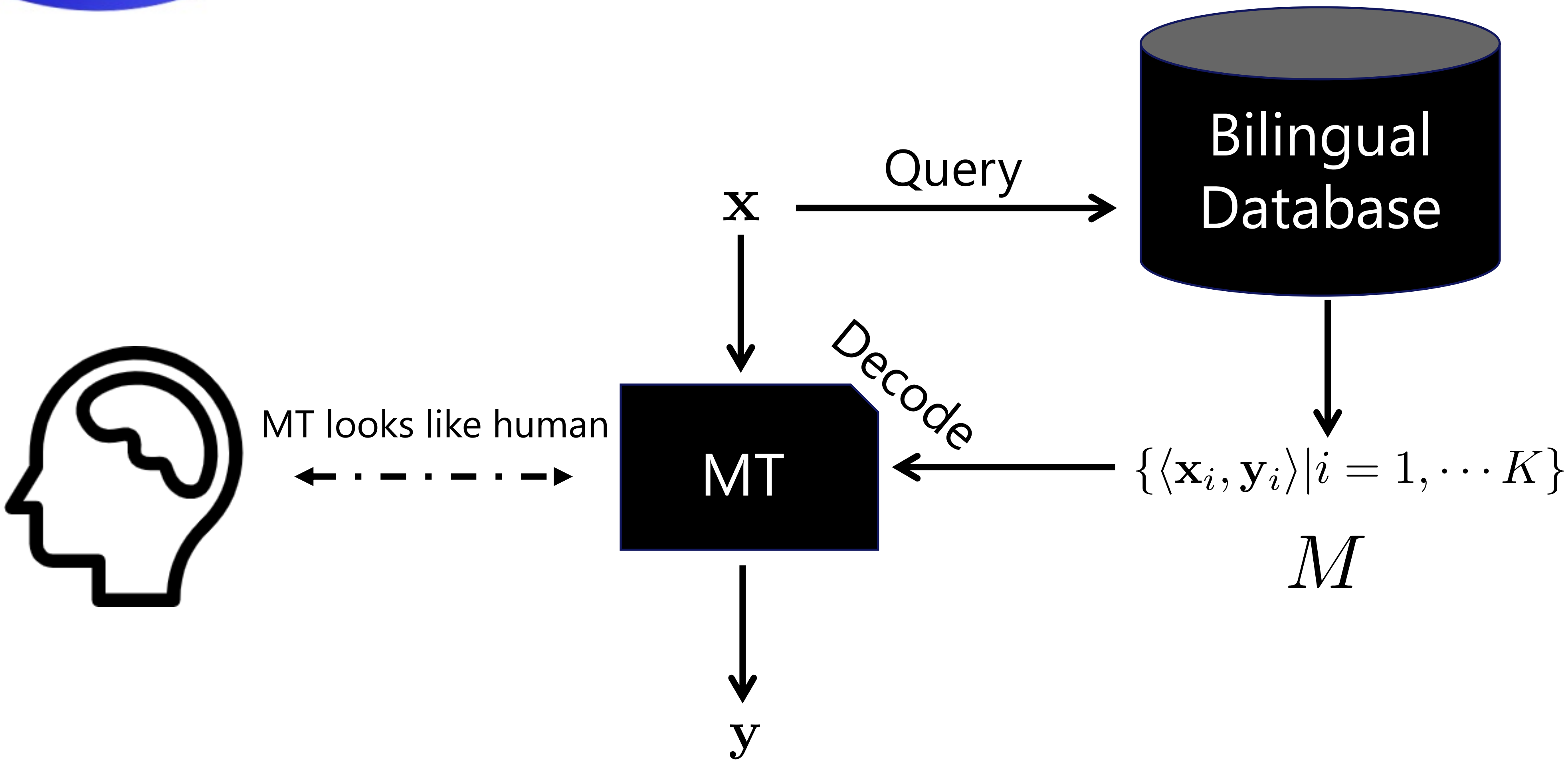
- Translating from scratch is not easy

Why retrieval is beneficial to translation?

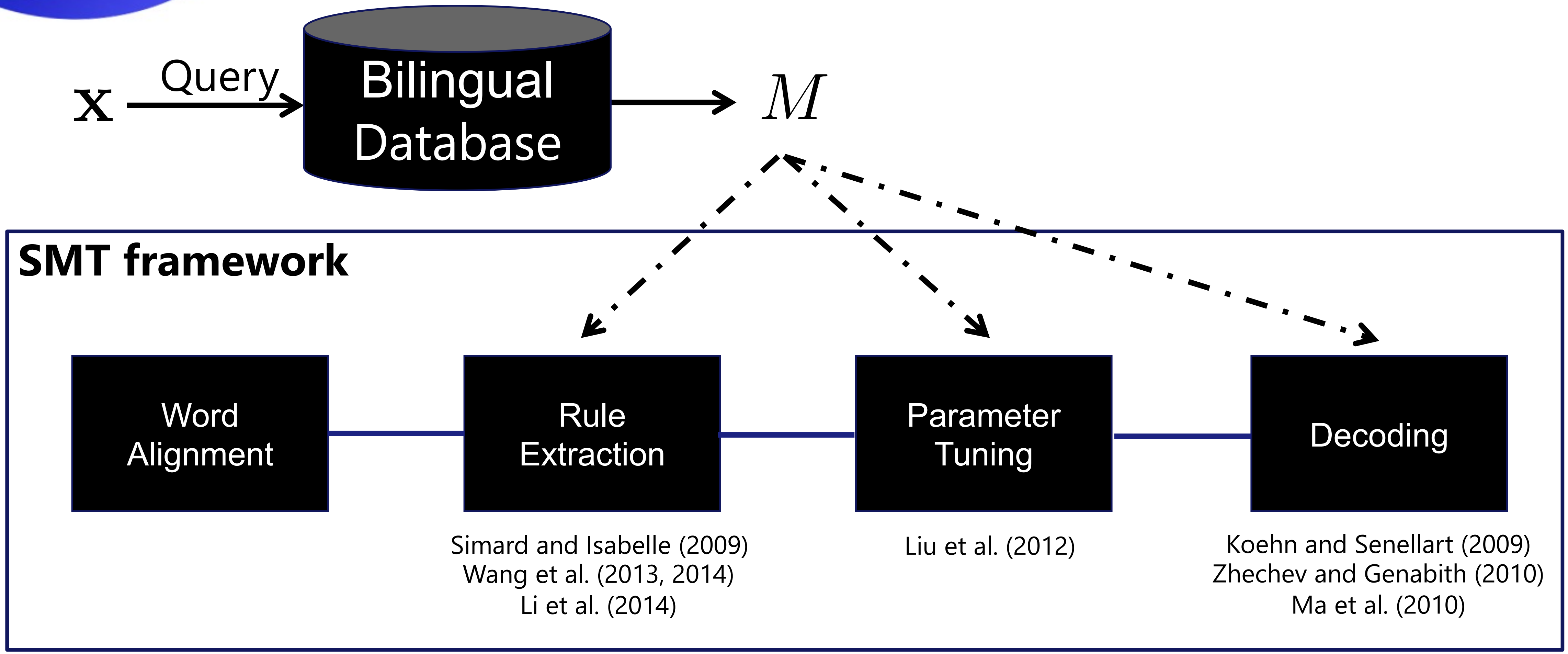


- Translation memory includes **useful translation knowledge**
- Translating from memory is easier

TM augmented MT: Paradigm



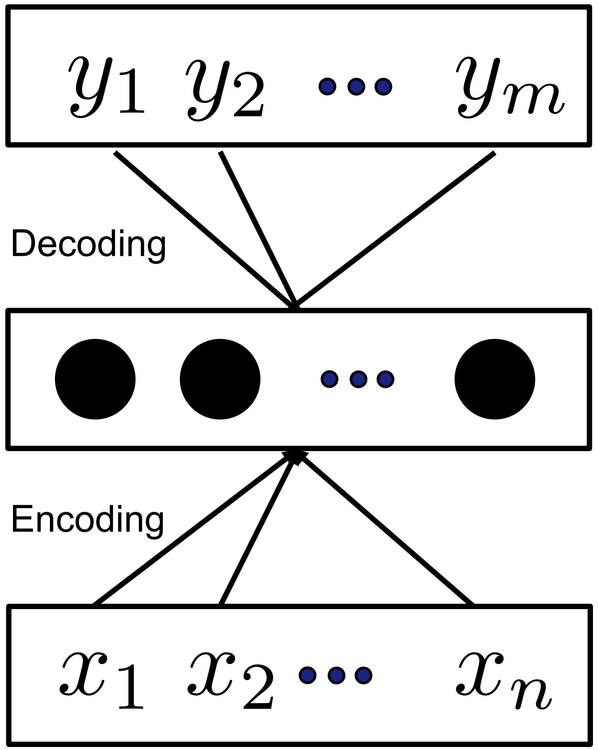
TM augmented SMT



Challenge: error propagation due to the pipeline framework

NMT: End-to-End Framework

End-to-end modeling



End-to-end training

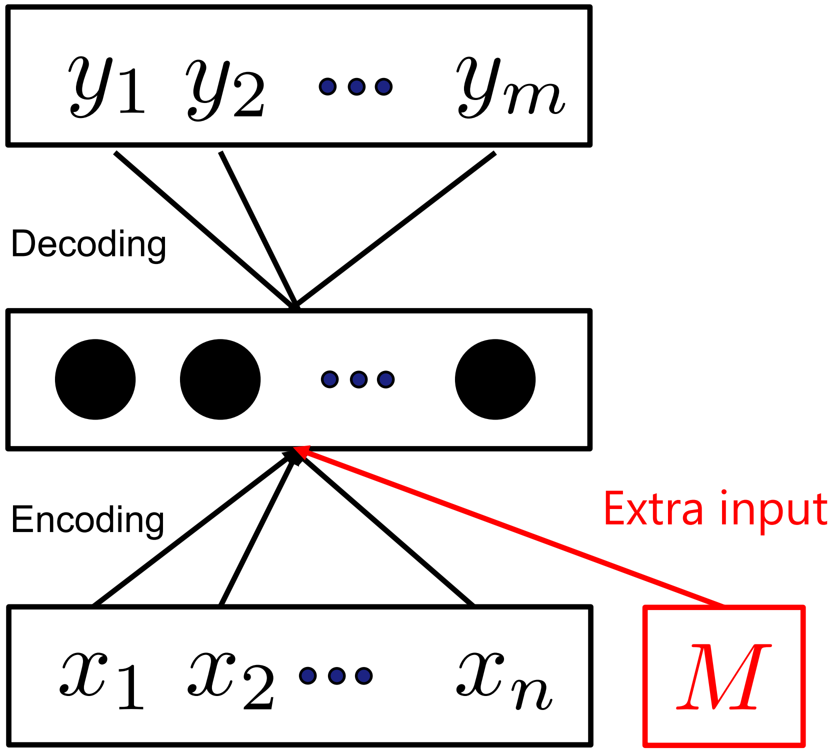
$$\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y} \rangle} \log p(\mathbf{y} | \mathbf{x}; \theta)$$

NMT achieves SOTA performance on many benchmarks

NMT: End-to-End Framework

Tencent AI Lab

End-to-end modeling



End-to-end training

$$\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y} \rangle} \log p(\mathbf{y} | \mathbf{x}; \theta)$$

$$\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y}, M \rangle} \log p(\mathbf{y} | \mathbf{x}, M; \theta)$$

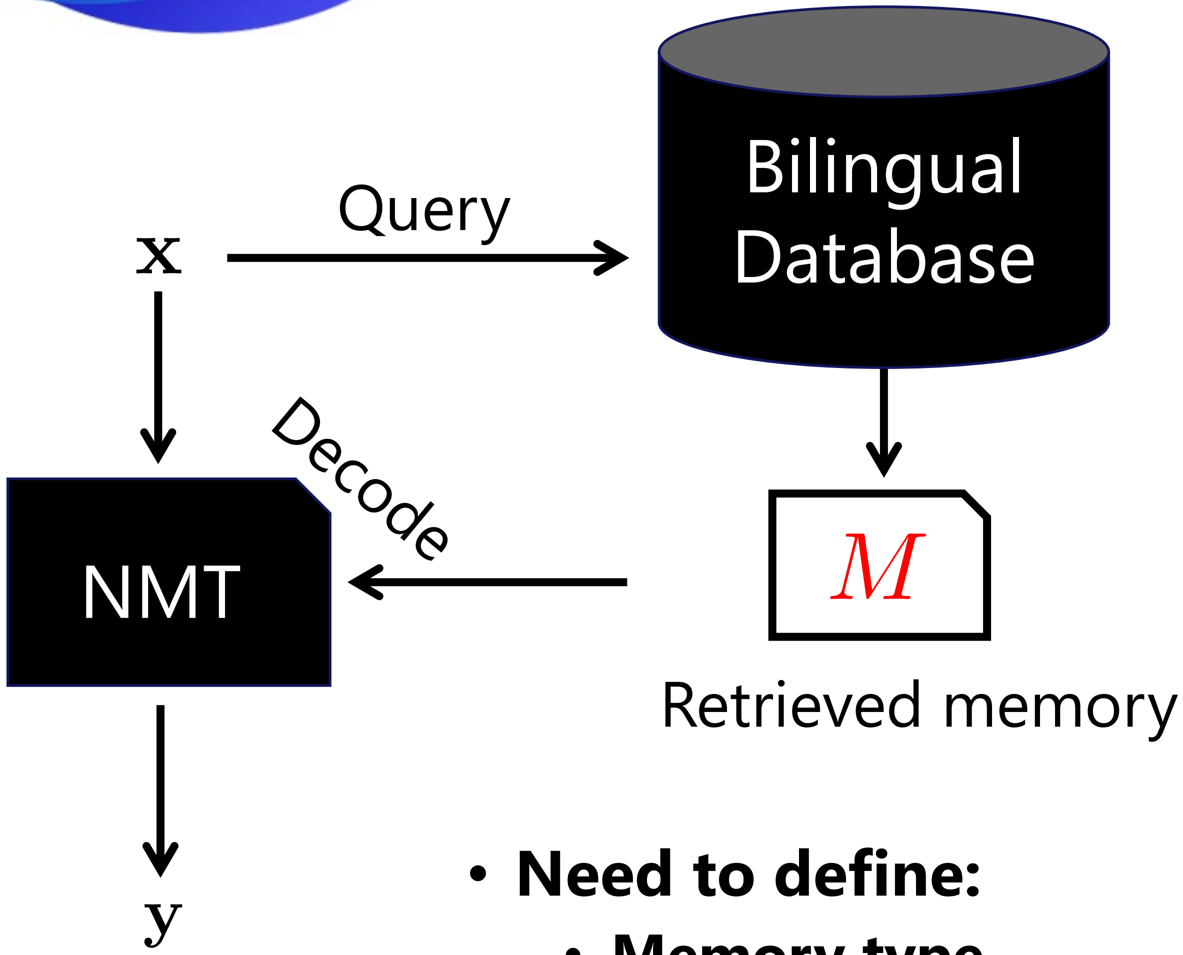
Easily scaling to leverage any extra information

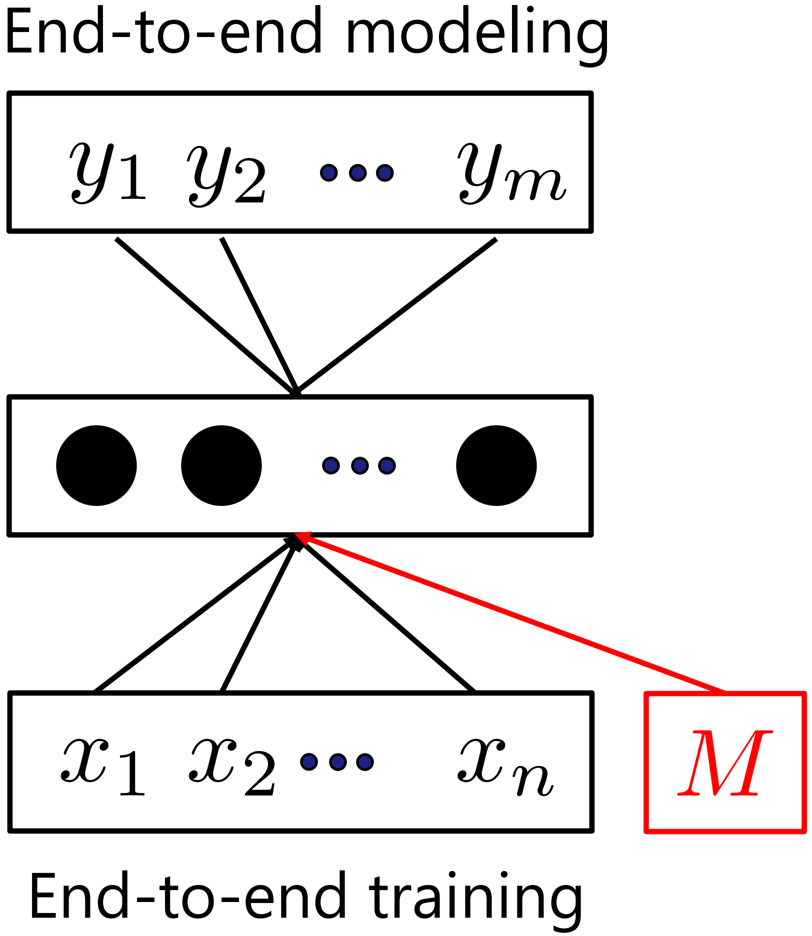
Making TM-augmented NMT promising

Outline

- Background and Introduction
- Language Modeling
- Open-Domain Dialogue Systems
- **Neural Machine Translation**
  - Motivation
  - **TM-augmented NMT Framework**
  - TM-augmented Models
    - Standard model
    - Dual model
    - Unified model
- Conclusion and Outlook

TM-augmented NMT Framework: Overview





- **Need to define:**
  - **Memory type**
  - **Retrieval metric**
  - **Model architecture**

$$\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y}, M \rangle} \log p(\mathbf{y} | \mathbf{x}, M; \theta)$$

TM-augmented NMT Framework: Memory Type

$\mathbf{x}$	huoqu huo shezhi yu pizhu guanlian de duixiang 获取 或 设置 与 批注 关联 的 对象
$\hat{\mathbf{y}}_{1:7}$	gets or sets an object that is ?

Test sentence

$\mathbf{x}^1$	huoqu yu pizhu biaoqian guanlian de duixiang 获取 与 批注 标签 关联 的 对象
$\mathbf{y}^1$	gets an object that is **associated** with the annotation label

A sentence in database

- Type 1: <sentence, sentence>

Query $\mathbf{x}$

$$\langle \mathbf{x}^1, \mathbf{y}^1 \rangle$$

Key-value pairs

Sentence-level memory

- Type 2: <sentence, word>

Query $\mathbf{x} || \hat{\mathbf{y}}_{1:7}$

$$\langle \mathbf{x}^1 || \mathbf{y}^1_{1:5}, \text{associated} \rangle$$
$$\cdots$$

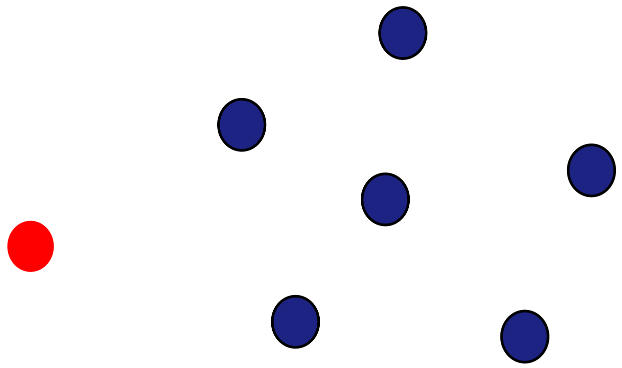
Key-value pairs

word-level memory

TM-augmented NMT Framework: Memory Type

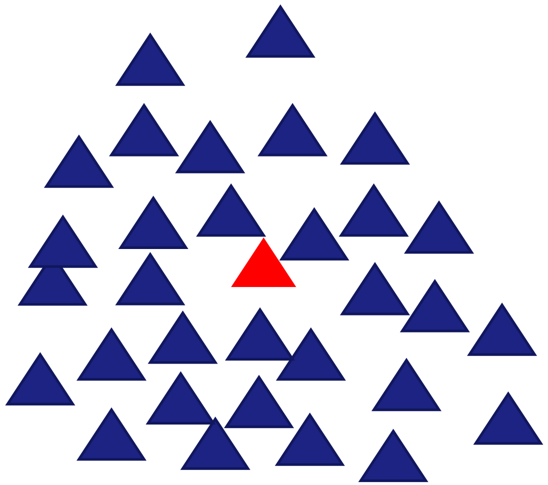
- Sentence-level memory type VS word-level memory type

Query $\langle \mathbf{x}^1, \mathbf{y}^1 \rangle$



Database is sparse
- may not have similar neighbors
- High retrieval efficiency

Query $\langle \mathbf{x}^1 || \mathbf{y}^1_{1:5}, \mathrm{associated} \rangle$



Database is dense
- may have similar neighbors
- Low retrieval efficiency

TM-augmented NMT Framework: Retrieval Metrics

	huoqu huo shezhi yu pizhu guanlian de duixiang
$\mathbf{x}$	获取 或 设置 与 批注 关联 的 对象
$\hat{\mathbf{y}}_{1:7}$	gets or sets an object that is ?

Test sentence

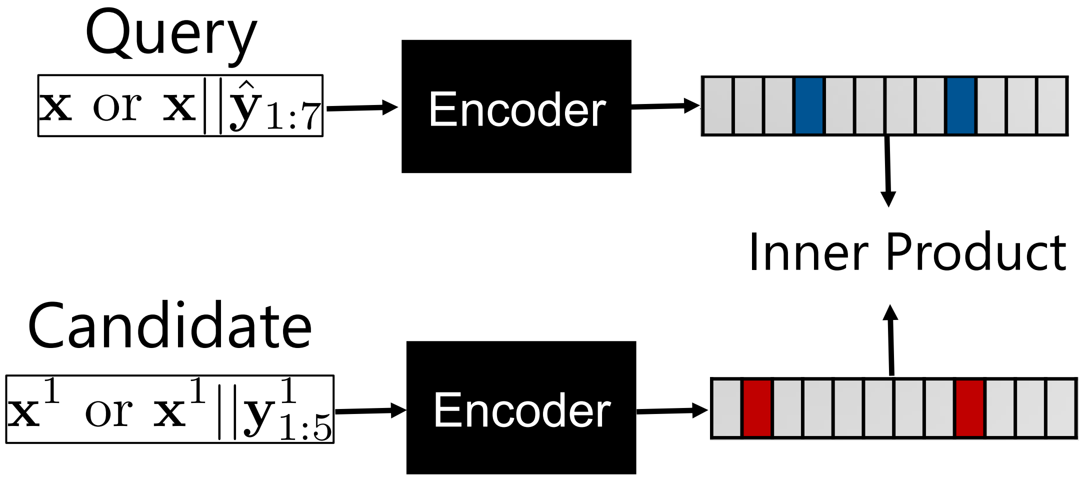
$\mathbf{x}^1$	huoqu yu pizhu biaoqian guanlian de duixiang 获取 与 批注 标签 关联 的 对象
$\mathbf{y}^1$	gets an object that is **associated** with the annotation label

A sentence in database

- Word Matching
  - TF-IDF
  - Normalized edit distance

$$1 - \frac{\text{edit-dist}(\mathbf{x}, \mathbf{x}^1)}{\max(|\mathbf{x}|, |\mathbf{x}^1|)}$$

- Dense Retrieval



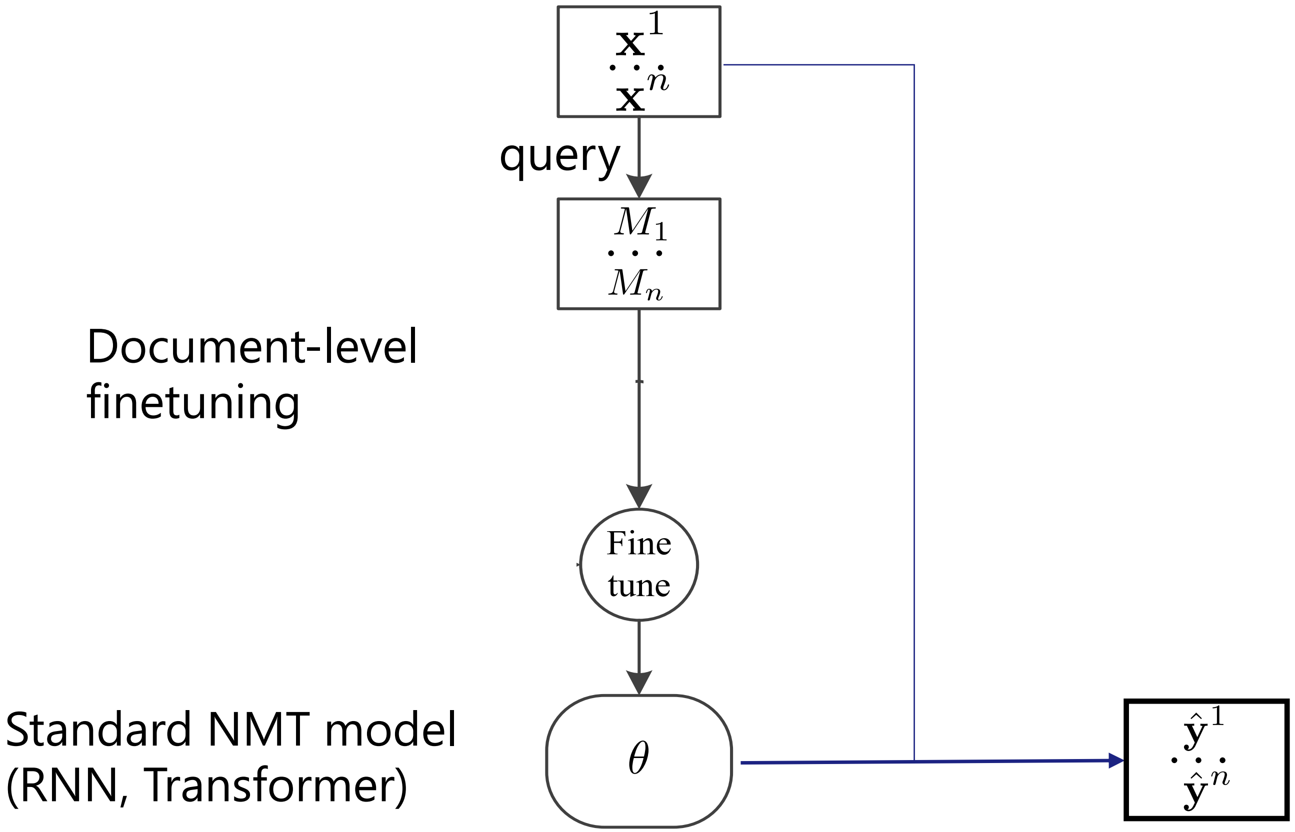
TM-augmented NMT: Categories

Ref.	Memory Type	Retrieval Metric	*Model Architecture*
Li et al. (2016) Farajian et al. (2017) **Bulte et al. (2019)**	<sentence, sentence>	Word Matching	***Standard model (fixed NMT architecture)***
Xu et al. (2020)	<sentence, sentence>	Word Matching Dense retrieval	
Zhang et al. (2018)	<sentence, sentence>	Word Matching	***Dual model (partially changed architecture)***
Khandelwal et al. (2021) Zheng et al. (2021) Wang et al. (2022) Meng et al. (2022)	<sentence, word>	Dense retrieval	
Gu et al. (2018) *Xia et al. (2019)* *He et al. (2021)*	<sentence, sentence>	Word Matching	***Unified model (changed architecture)***
Cai et al. (2021)	<sentence, sentence>	Dense retrieval	

Outline

- Background and Introduction
- Language Modeling
- Open-Domain Dialogue Systems
- **Neural Machine Translation**
  - Motivation
  - TM-augmented NMT Framework
  - **TM-augmented Models**
    - **Standard model**
    - Dual model
    - Unified model
- Conclusion and Outlook

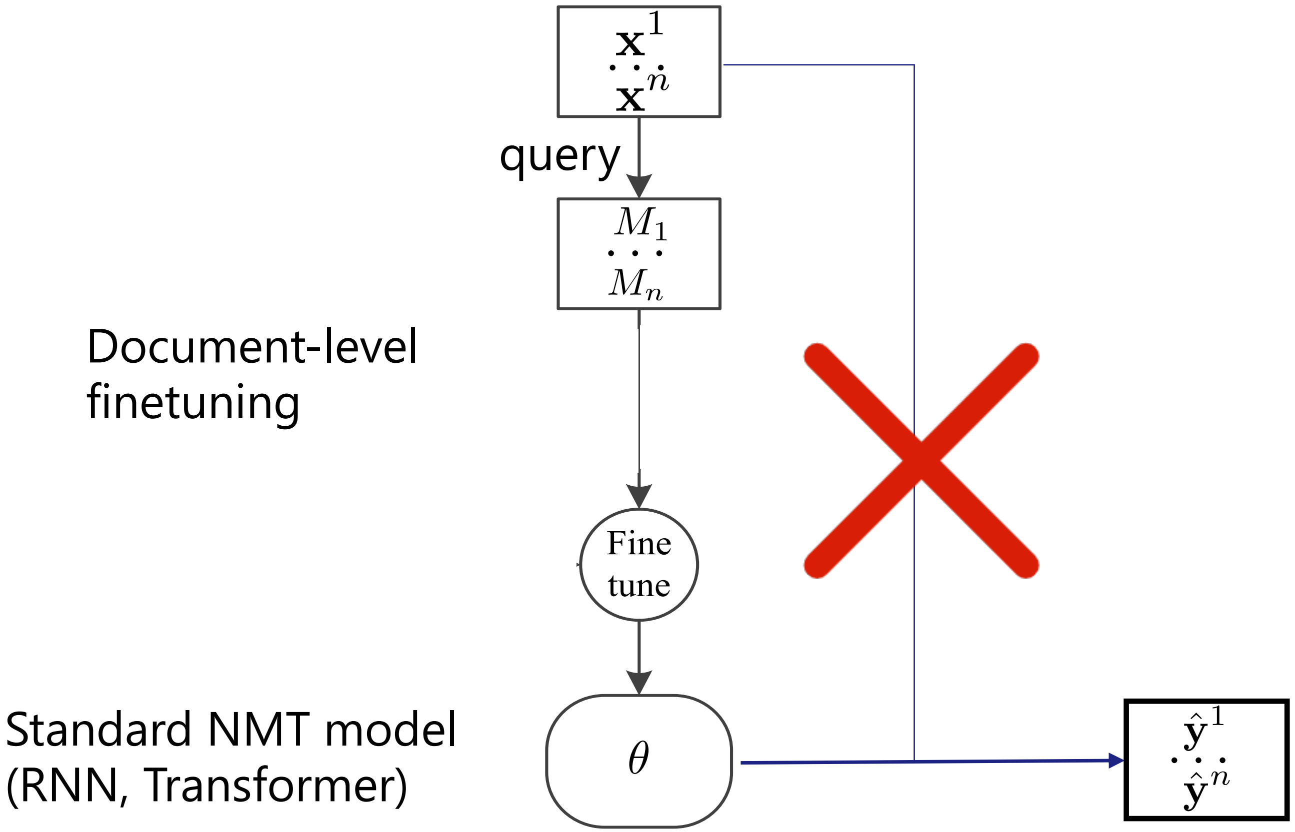
Standard Model: Finetuning



$\mathrm{TM}^i (i \neq 1)$ may not be similar to $\mathbf{x}^1$

Fig credit: Xiaoqing Li, Jiajun Zhang, Chengqing Zong. One sentence one model for neural machine translation. arxiv16.

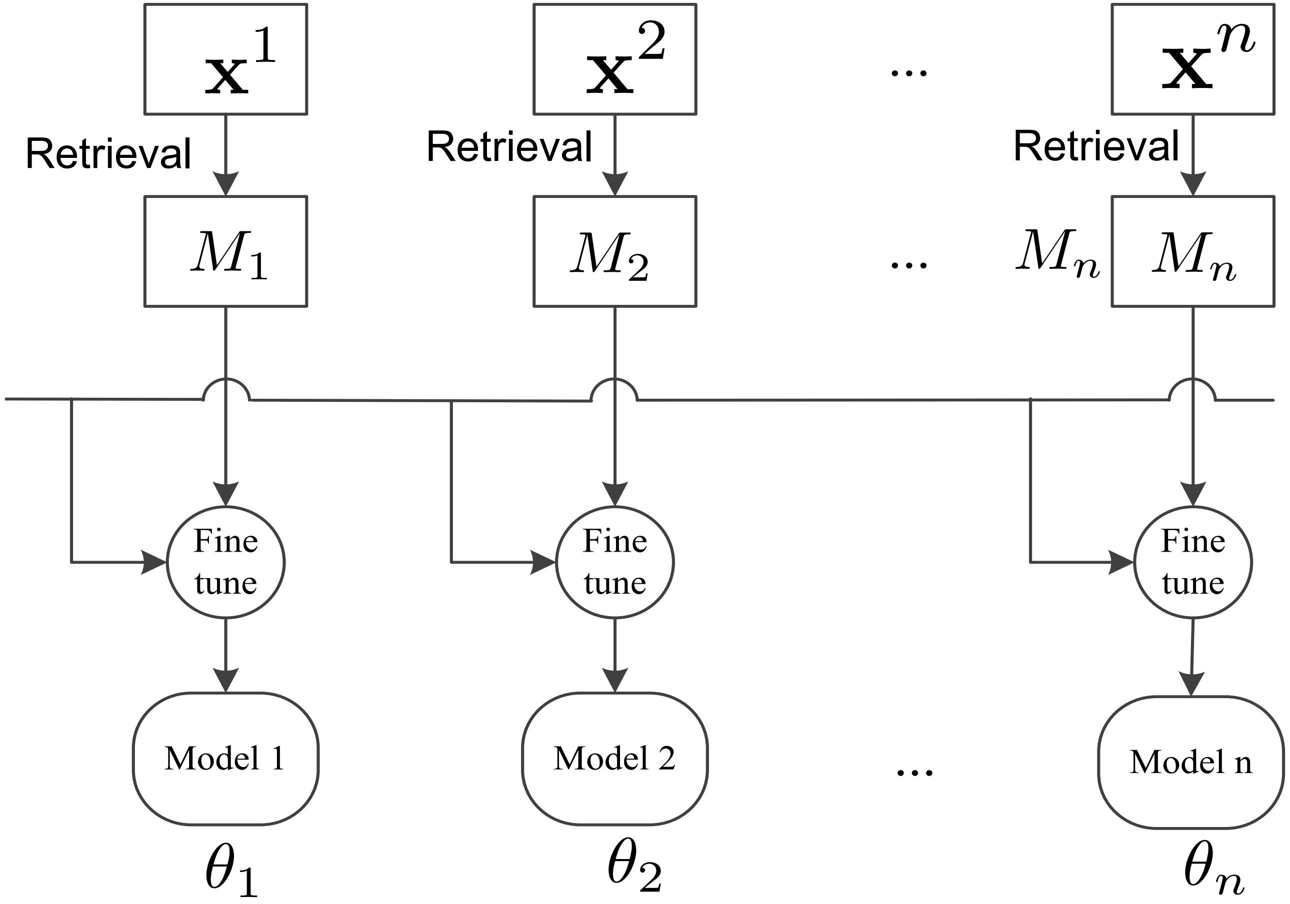
Standard Model: Finetuning



$\mathrm{TM}^i (i \neq 1)$ may not be similar to $\mathbf{x}^1$

Fig credit: Xiaoqing Li, Jiajun Zhang, Chengqing Zong. One sentence one model for neural machine translation. arxiv16.

Standard Model: Sentence-level Finetuning

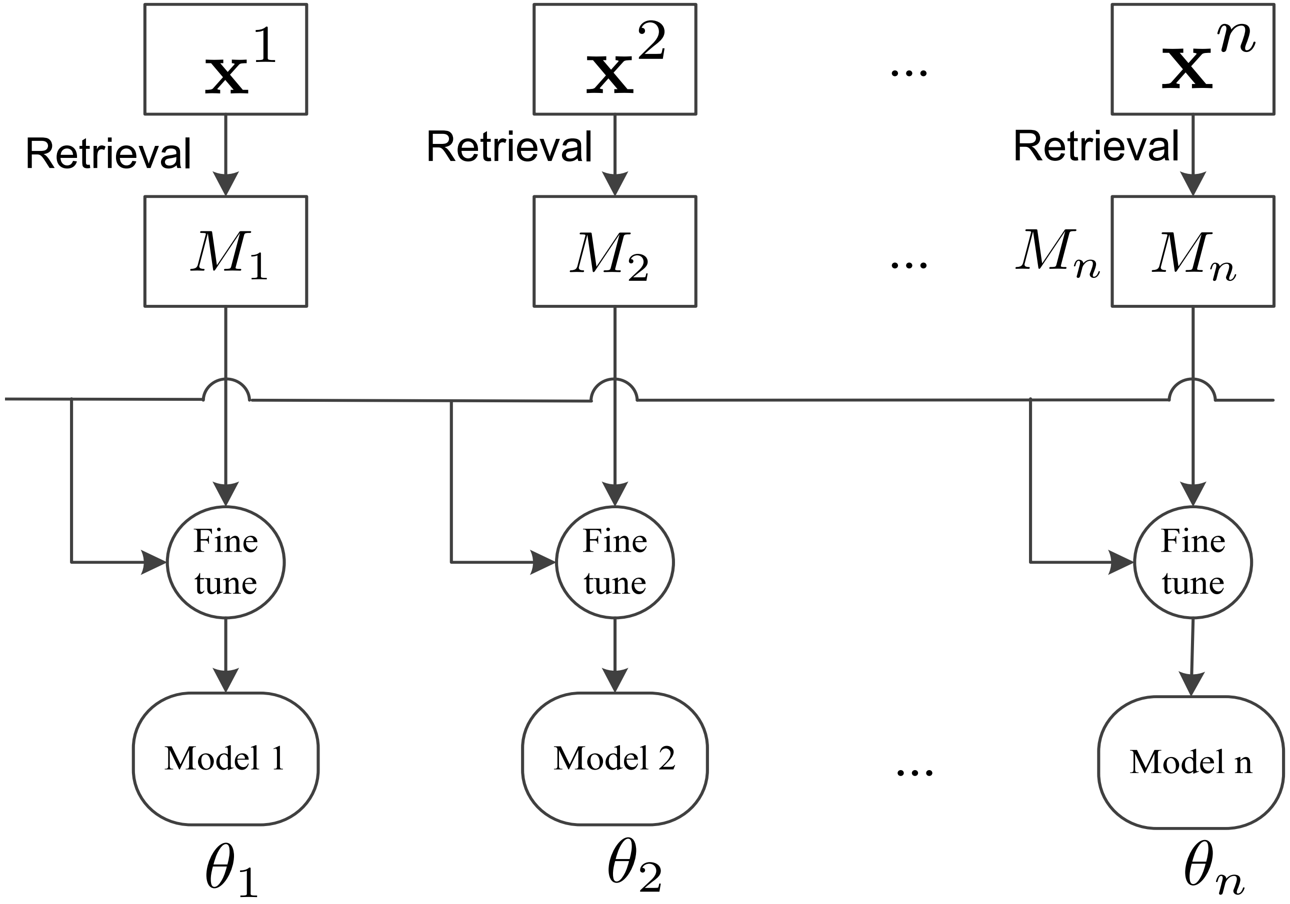


Standard NMT model
(RNN, Transformer)

Fig credit: Xiaoqing Li, Jiajun Zhang, Chengqing Zong. One sentence one model for neural machine translation. arxiv16.

Standard Model: Sentence-level Finetuning

Tencent AI Lab



Standard NMT model
(RNN, Transformer)

Finetuning objective

$$\max_{\theta_n} \sum_{\langle x,y\rangle \in M_n} \log p(y|x;\theta_n)$$

Standard NMT model
(RNN, Transformer)

- Optimize θ_n
  - Run SGD on M_n
- Decode with θ_n

On-the-fly finetuning and testing

Fig credit: Xiaoqing Li, Jiajun Zhang, Chengqing Zong. One sentence one model for neural machine translation. arxiv16.

Standard Model: Sentence-level Fintuning

- Drawbacks in sentence-level finetuning
  - Low efficiency
    - Relatively large memory size is used to ensure good translations
    - But the efficiency of finetuning is low
  - Setting hyperparameters is not trivial
    - Hyperparameters are sensitive to different test sentences.

Standard Model: Input Augmentation

Tencent AI Lab

Training

$\mathcal{D}:$ $(\mathbf{x}^i, \mathbf{y}^i)$

$\mathcal{M}$

Retrieval

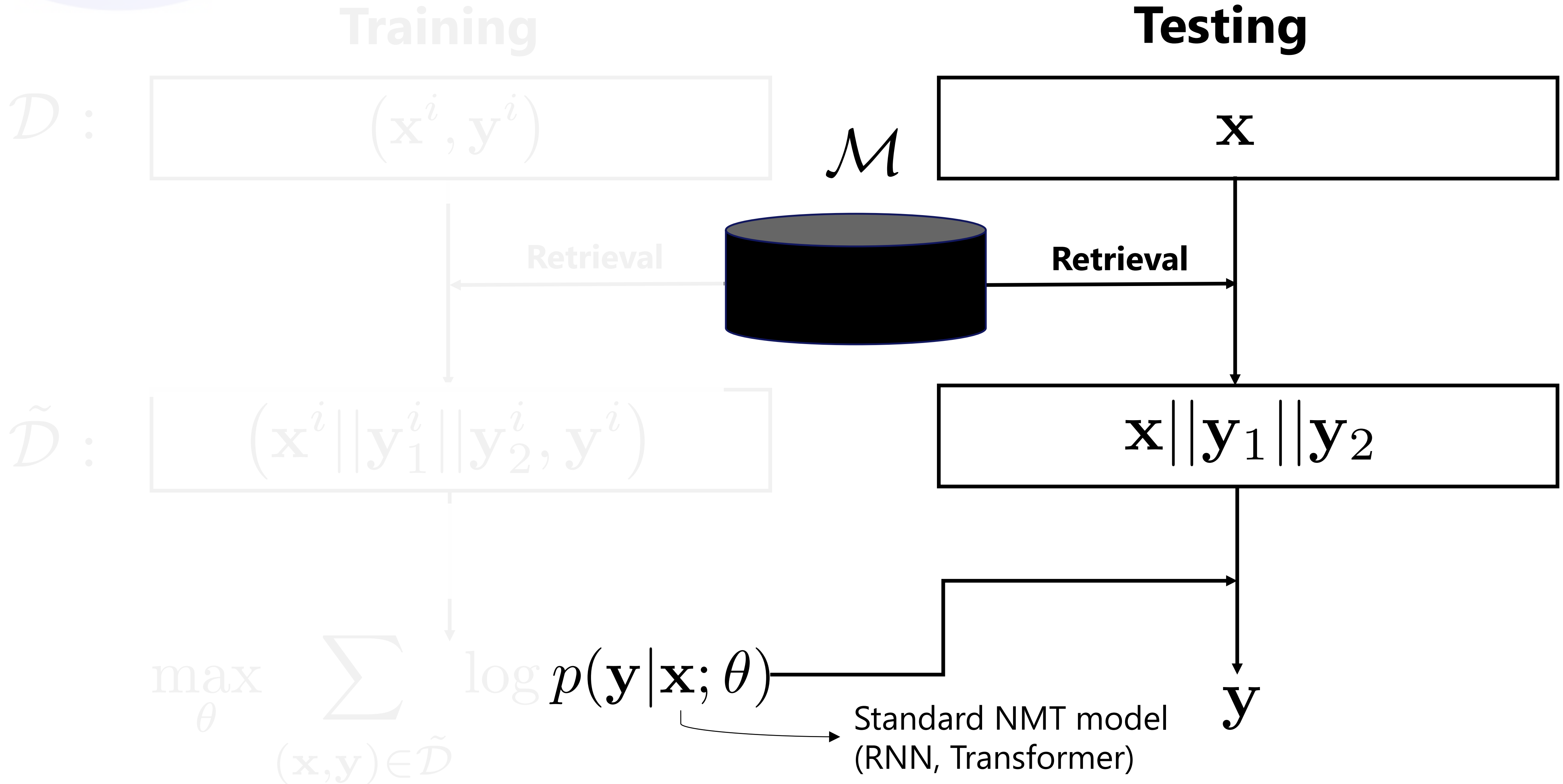
$\tilde{\mathcal{D}}:$ $(\mathbf{x}^i || \mathbf{y}_1^i || \mathbf{y}_2^i, \mathbf{y}^i)$

$$\max_\theta \sum_{(\mathbf{x},\mathbf{y})\in\tilde{\mathcal{D}}} \log p(\mathbf{y}|\mathbf{x};\theta)$$

Standard NMT model
(RNN, Transformer)

Bram Bulte, Arda Tezcan. Neural Fuzzy Repair: Integrating Fuzzy Matches into Neural Machine Translation. ACL19.

Standard Model: Input Augmentation



Bram Bulte, Arda Tezcan. Neural Fuzzy Repair: Integrating Fuzzy Matches into Neural Machine Translation. ACL19.

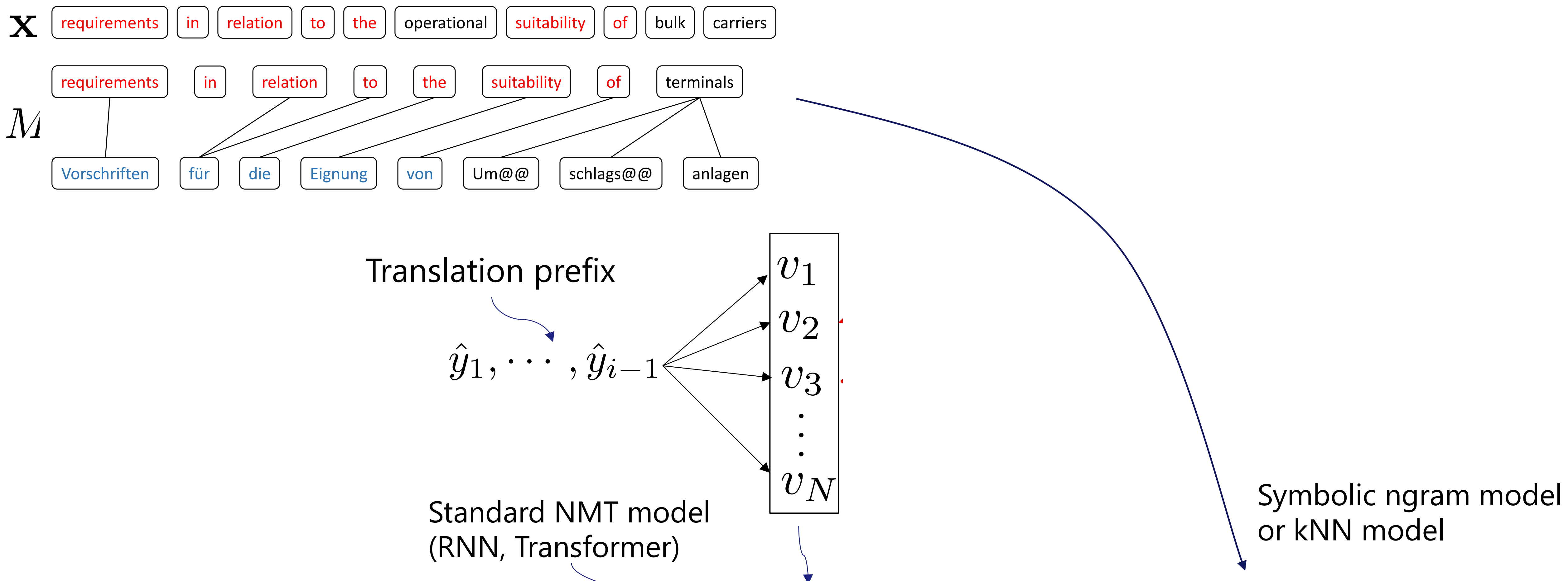
Pros and Cons: Both standard models for TM

- Pros
  - Both sentence-level finetuning and input augmentation are easy to implement
  - Both are general to be applied to any NMT models
- Cons
  - Their Model architecture is not customized for translation memory
  - They can not make full use of translation memory
  - Limited translation quality

Outline

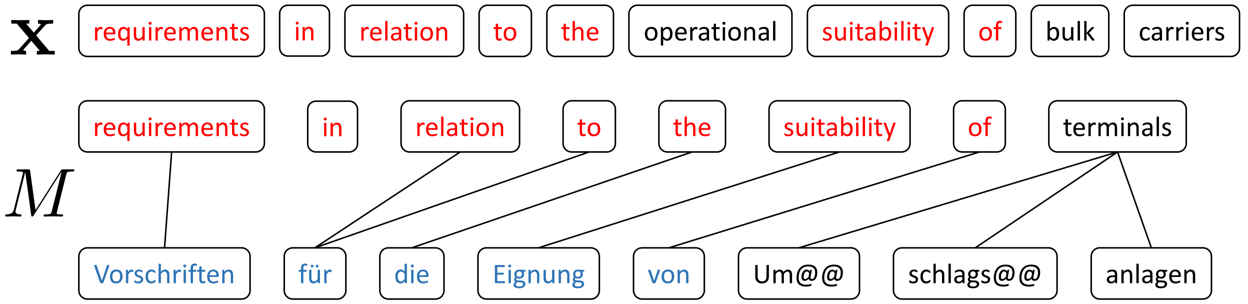
- Background and Introduction
- Language Modeling
- Open-Domain Dialogue Systems
- **Neural Machine Translation**
  - Motivation
  - TM-augmented NMT Framework
  - **TM-augmented Models**
    - Standard model
    - **Dual model**
    - Unified model
- Conclusion and Outlook

Dual Model: Key Idea



$$p(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) = p_{\mathrm{NMT}}(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) + \lambda \times p_{\mathrm{TM}}(y_i)$$

Dual Model by Ngram Model



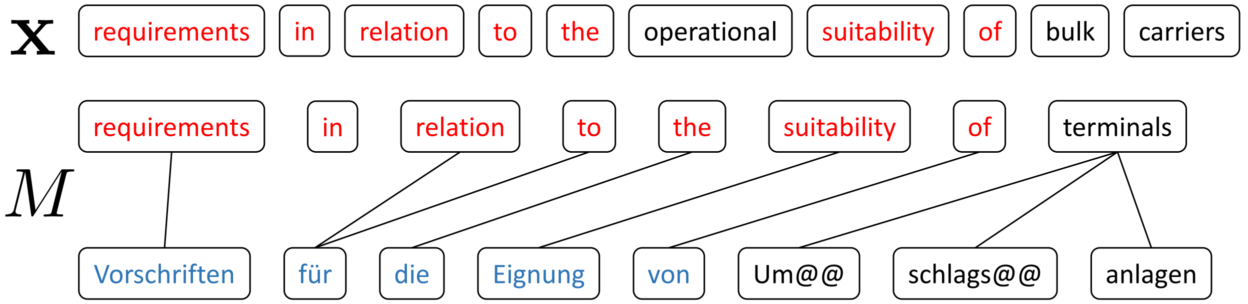
Weighted n-gram

(Vorschriften, 0.8)
(fur, 0.8)
(die, 0.8)
(Eignung, 0.8)
(von, 0.8)

(Vorschriften fur, 0.8)
(fur die, 0.8)
… …
(Vorschriften fur die Eignung, 0.8)
(fur die Eignung von, 0.8)

Fig credit: J. Zhang, M. Utiyama, E. Sumita, G. Neubig, S. Nakamura. Guiding Neural Machine Translation with Retrieved Translation Pieces. NAACL18.

Dual Model by Ngram Model ry



Weighted n-gram

(Vorschriften, 0.8)	(Vorschriften fur, 0.8)
(fur, 0.8)	(fur die, 0.8)
(die, 0.8)	… …
(Eignung, 0.8)	(Vorschriften fur die Eignung, 0.8)
(von, 0.8)	(fur die Eignung von, 0.8)

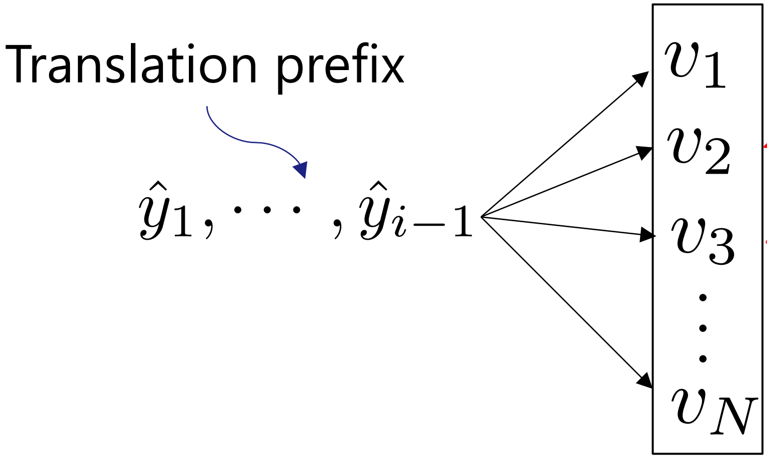
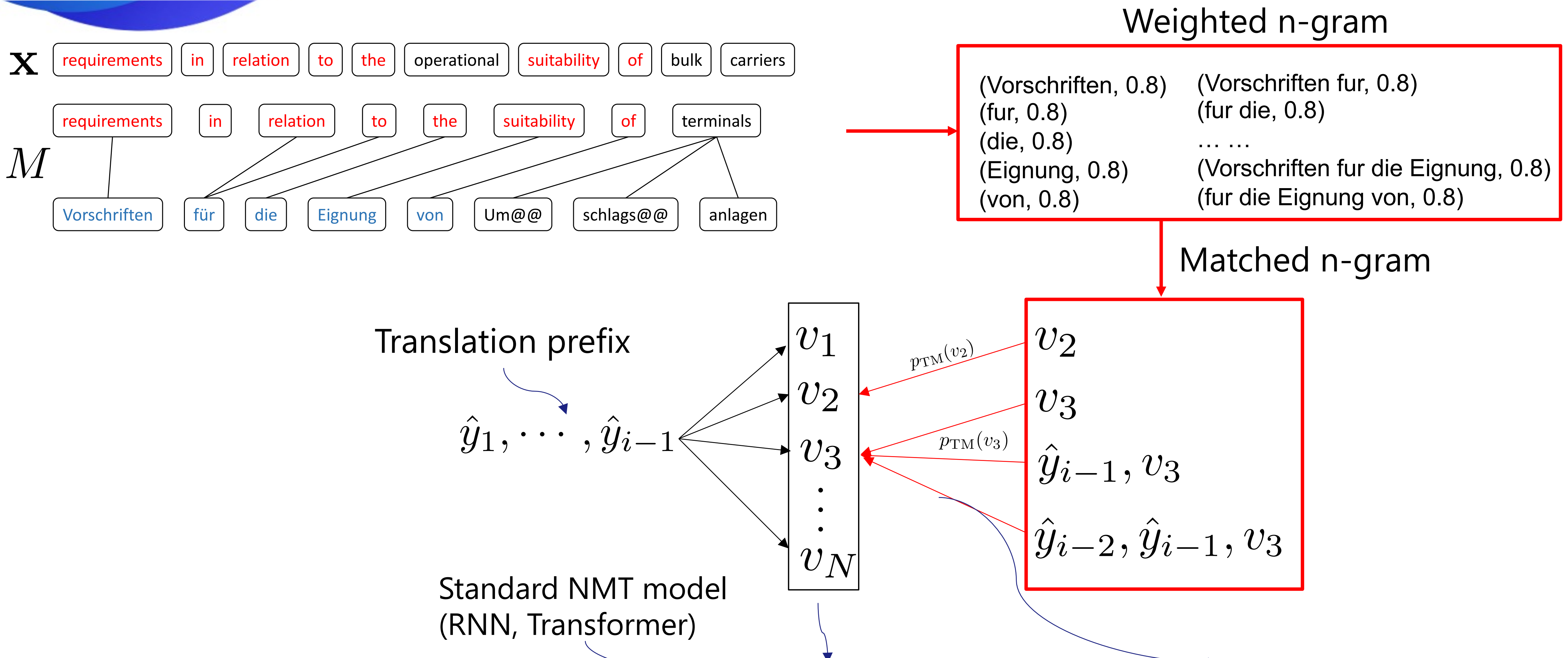


Fig credit: J. Zhang, M. Utiyama, E. Sumita, G. Neubig, S. Nakamura. Guiding Neural Machine Translation with Retrieved Translation Pieces. NAACL18.

Dual Model by Ngram Model



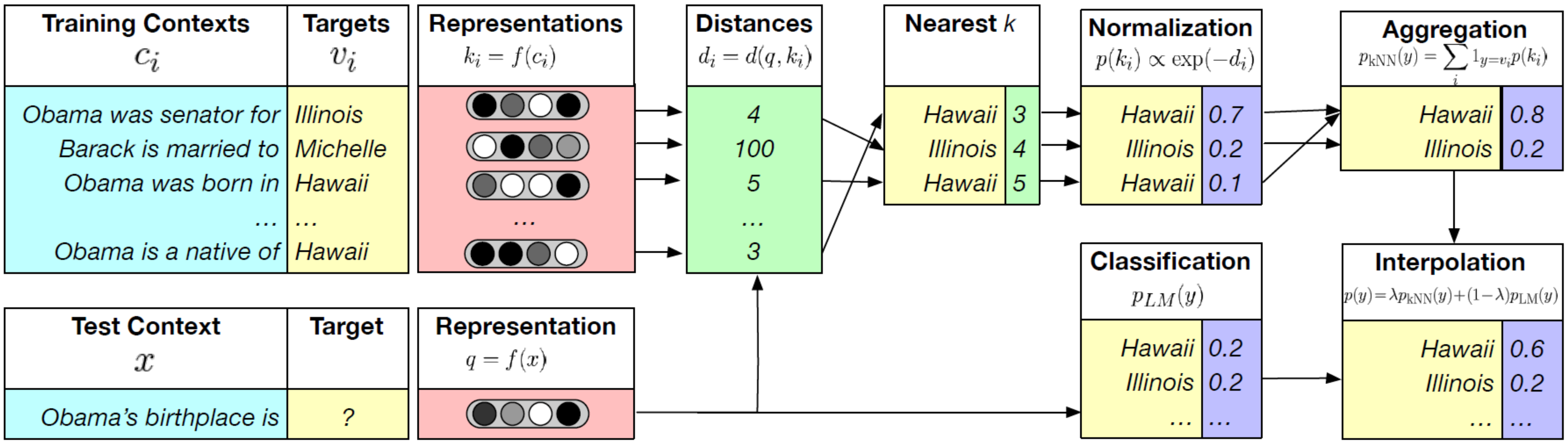
$$p(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) = p_{\mathrm{NMT}}(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) + \lambda \times p_{\mathrm{TM}}(y_i)$$

Fig credit: J. Zhang, M. Utiyama, E. Sumita, G. Neubig, S. Nakamura. Guiding Neural Machine Translation with Retrieved Translation Pieces. NAACL18.

Pros and Cons of Ngram Model

- Pros
  - The idea is intuitive
  - The prediction is interpretable
- Cons
  - Relying on exact matches of n-grams
  - Sensitive to interpolation coefficient

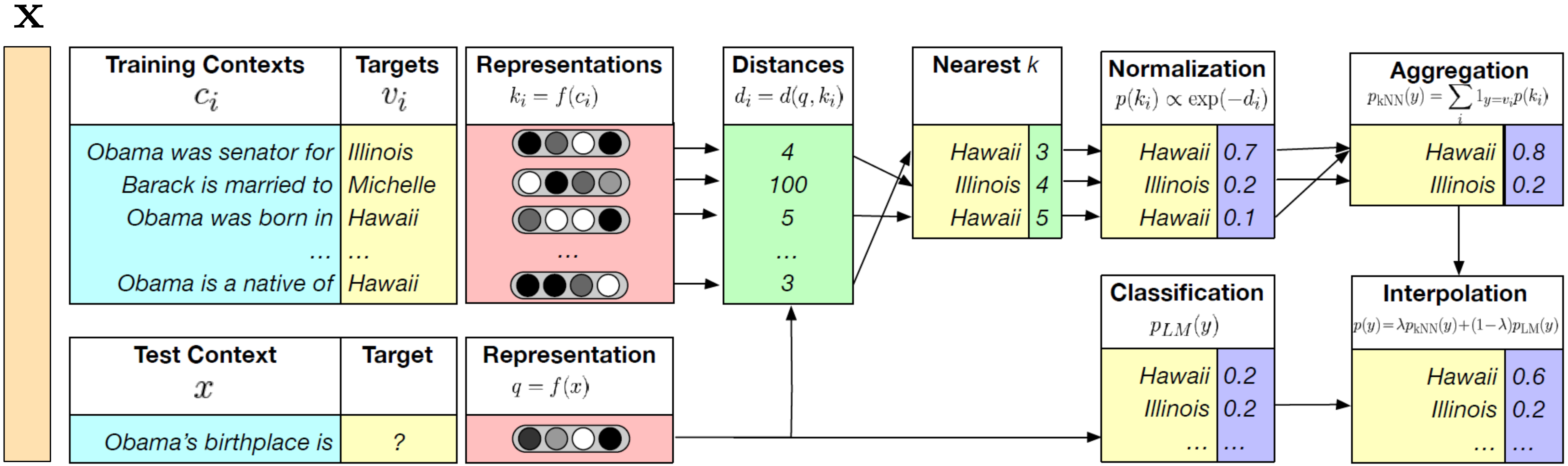
Dual model: KNN-NMT Extended from KNN-LM



KNN-LM

Fig. Credit: U. Khandelwal, O. Levy, D. Jurafsky, L. Zettlemoyer, M. Lewis. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 20.

Dual model: KNN-NMT Extended from KNN-LM



KNN-LM

Fig. Credit: U. Khandelwal, O. Levy, D. Jurafsky, L. Zettlemoyer, M. Lewis. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 20.

Dual model: KNN-NMT

Training Translation Contexts $(s^{(n)}, t_{i-1}^{(n)})$		Datastore: Representation $k_j = f(s^{(n)}, t_{i-1}^{(n)})$	Datastore: Target $v_j = t_i^{(n)}$
J'ai été à Paris.	*I have*		*been*
J'avais été à la maison.	*I had*		*been*
J'apprécie l'été.	*I enjoy*		*summer*
…	…	…	…
J'ai ma propre chambre.	*I have*		*my*

Fig. Credit: Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. ICLR21.

Dual model: KNN-NMT

Training Translation Contexts $(s^{(n)}, t_{i-1}^{(n)})$		Datastore: Representation $k_j = f(s^{(n)}, t_{i-1}^{(n)})$	Datastore: Target $v_j = t_i^{(n)}$
J'ai été à Paris.	*I have*		*been*
J'avais été à la maison.	*I had*		*been*
J'apprécie l'été.	*I enjoy*		*summer*
…	…	…	…
J'ai ma propre chambre.	*I have*		*my*

Test Input x	Generated tokens $\hat{y}_{1:i-1}$	Representation $q = f(x, \hat{y}_{1:i-1})$	Target y_i
J'ai été dans ma propre chambre.	*I have*		*?*

Fig. Credit: Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. ICLR21.

Dual model: KNN-NMT

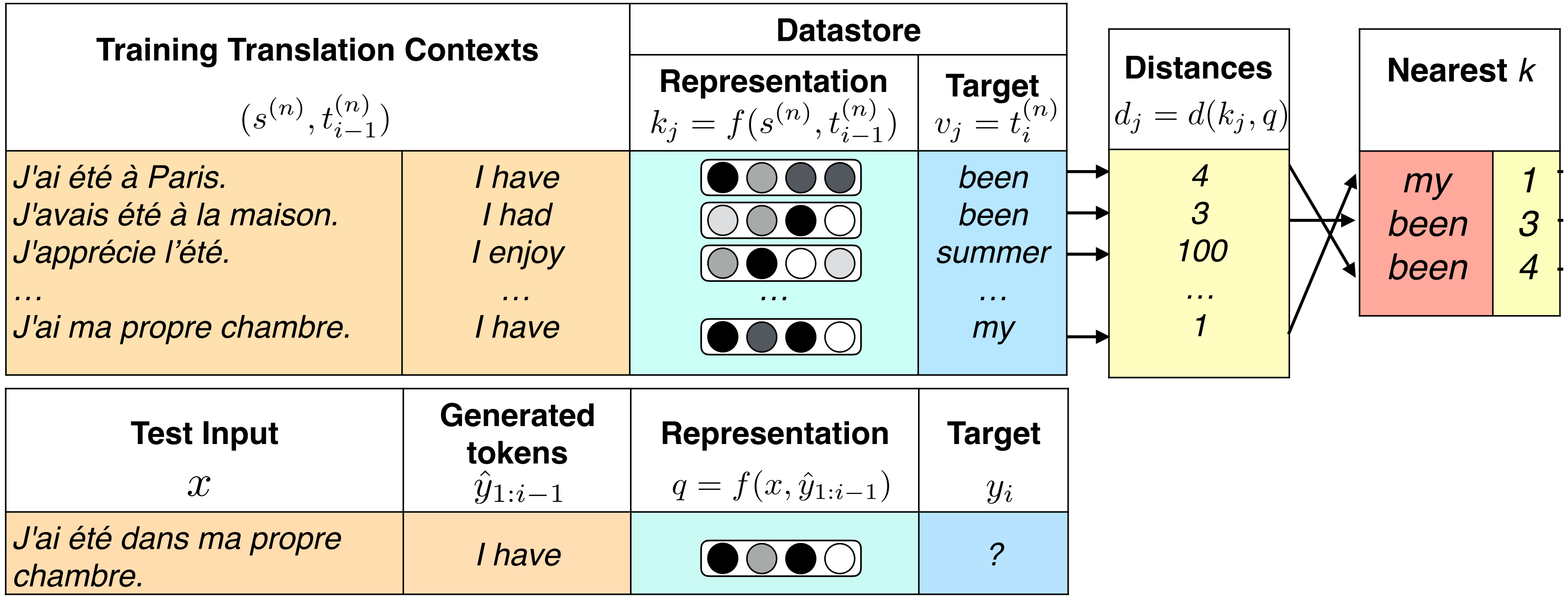


Fig. Credit: Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. ICLR21.

Dual model: KNN-NMT

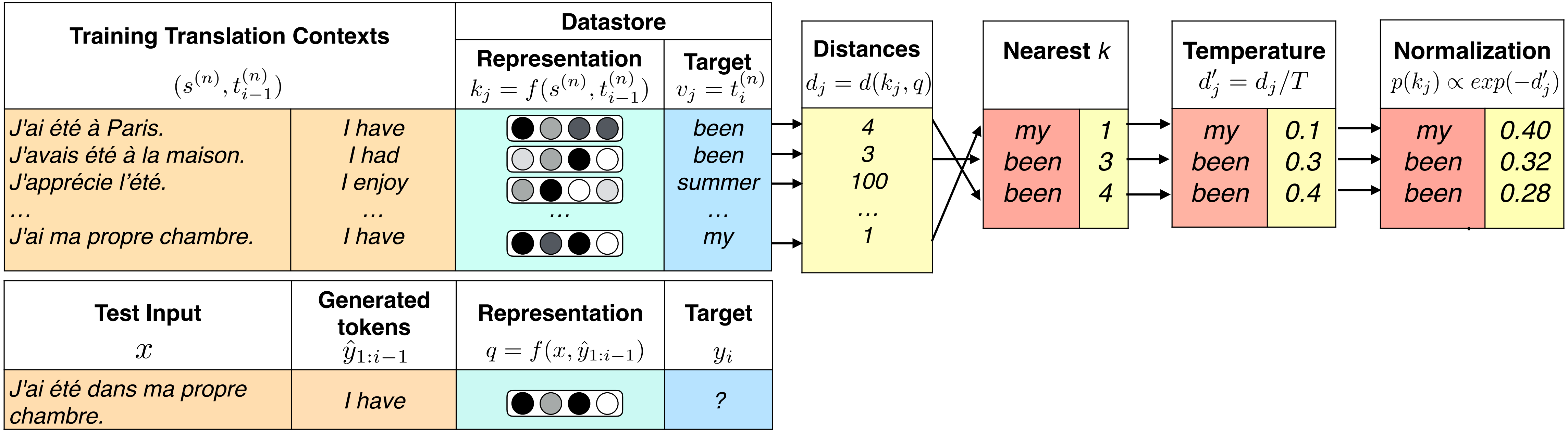


Fig. Credit: Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. ICLR21.

Dual model: KNN-NMT

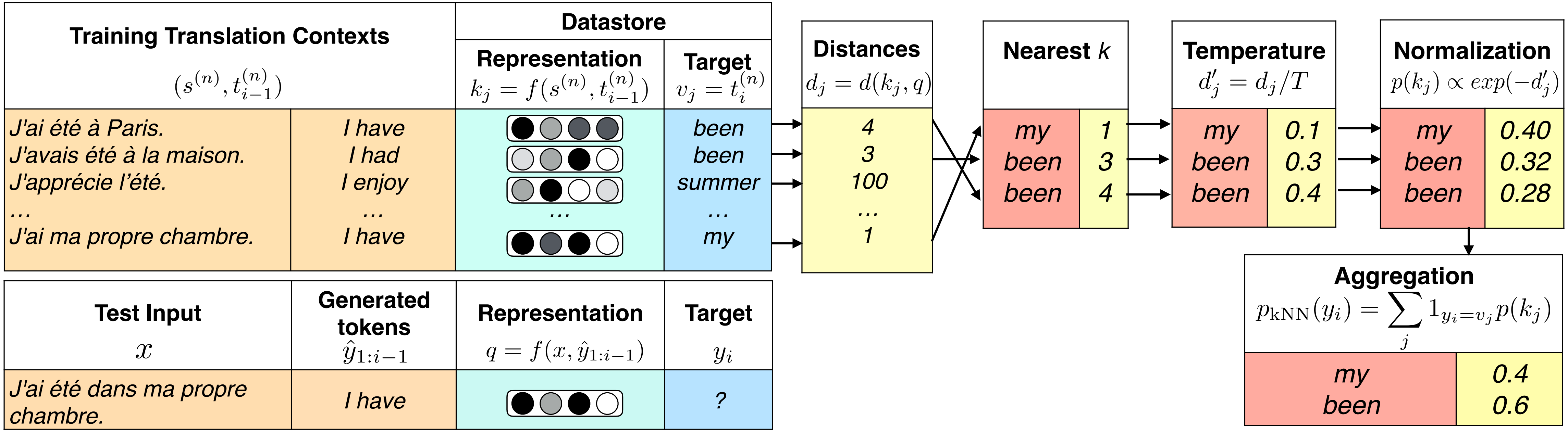
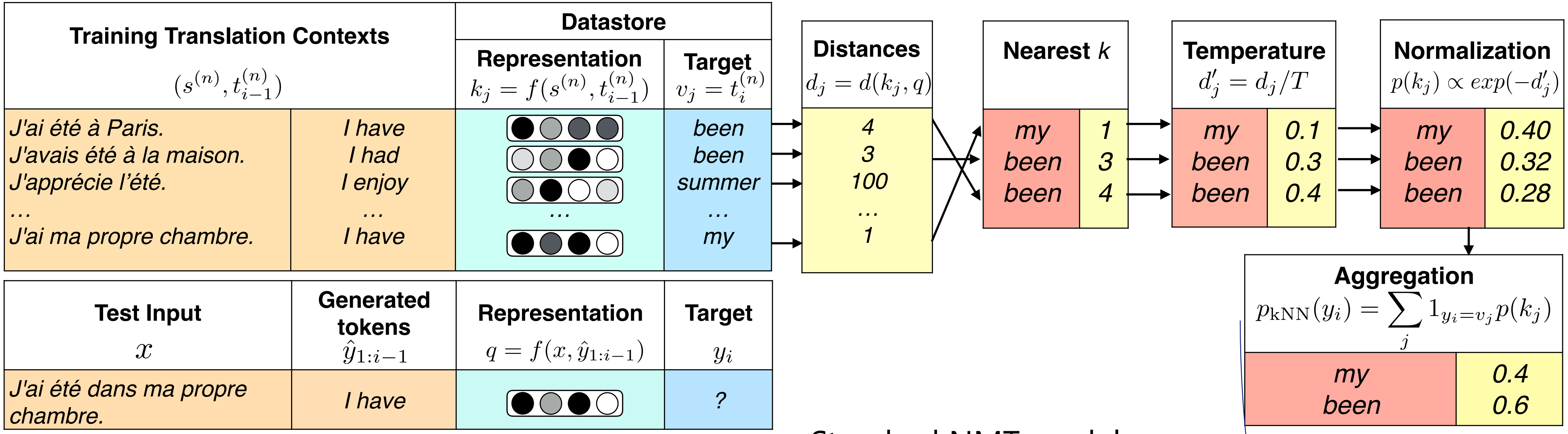


Fig. Credit: Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. ICLR21.

Dual model: KNN-NMT



Standard NMT model
(RNN, Transformer)

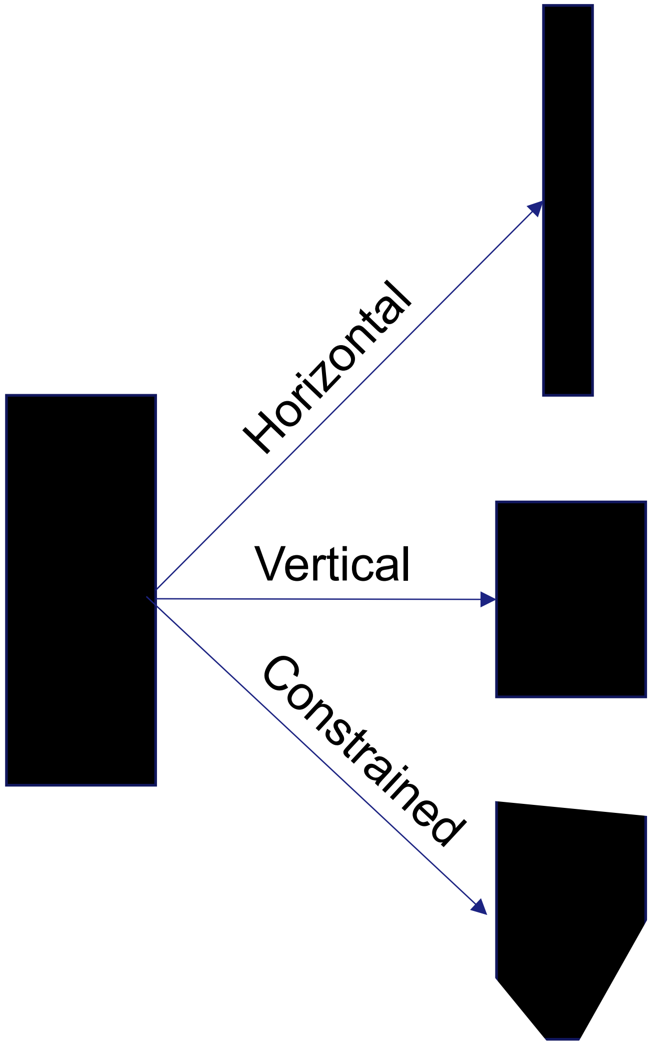
$$p(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) = p_{\text{NMT}}(y_i|x, \hat{\mathbf{y}}_{1:i-1}) + \lambda \times p_{k\text{NN}}(y_i)$$

Fig. Credit: Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. ICLR21.

Dual model: Improving KNN-NMT

- Issues in KNN-NMT
  - Low efficiency
  - Large Storage

- Three directions to improve KNN-NMT
  - (**Horizontal**) Dimension reduction
    Jahnson et al.(2021)
    Wang et al. (2022)
  - (**Vertical**) Example reduction
    He et al. (2021)
  - **Constrained** Search
    Meng et al. (2022)



Outline

- Background and Introduction
- Language Modeling
- Open-Domain Dialogue Systems
- **Neural Machine Translation**
  - Motivation
  - TM-augmented NMT Framework
  - **TM-augmented Models**
    - Standard model
    - Dual model
    - **Unified model**
- Conclusion and Outlook

Unified Model: Key idea to CopyNet for TM

Dual model $$p(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) = p_{\mathrm{NMT}}(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1}) + \lambda \times p_{\mathrm{TM}}(y_i)$$

- Three components: standard NMT, sub-model from tm, and interpolation
- The neural network is not learnable, and its parameters are directly taken from a well-trained standard NMT

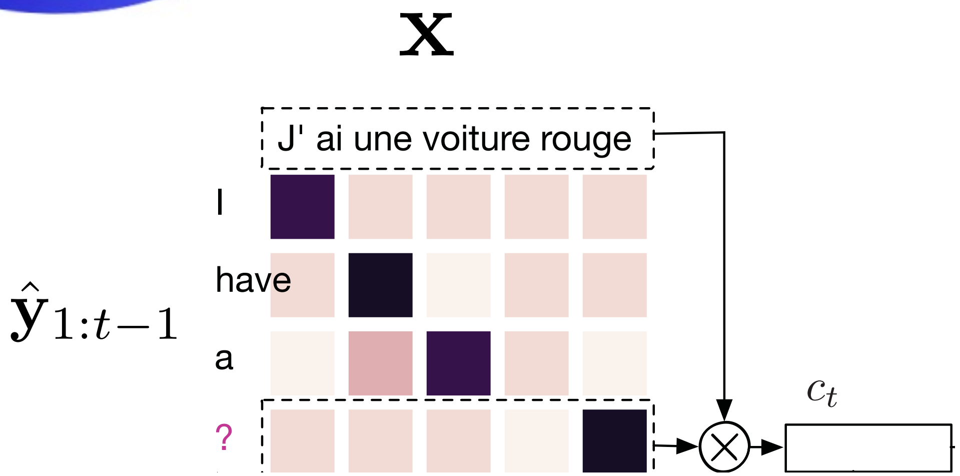
$$p(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1};\theta) = \zeta_t(\theta)\boxed{p_{\mathrm{NMT}}(y_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i-1};\theta)} + (1 - \boxed{\zeta_t(\theta)}) \times \boxed{p_{\mathrm{TM}}(y_i;\theta)}$$

- Three components: standard NMT, sub-model from tm, and interpolation
- Three components are modeled by neural networks whose parameters are learnable

How to define three components with neural networks?

Jiatao Gu, Yong Wang, Kyunghyun Cho, Victor O.K. Li. Search Engine Guided Neural Machine Translation. AAAI18.

Unified Model: CopyNet for TM



(a) Query the source sentence,
and the search engine returns
K translation pairs;

Fig. Credit: Jiatao Gu, Yong Wang, Kyunghyun Cho, Victor O.K. Li. Search Engine Guided Neural Machine Translation. AAAI18.

Unified Model: CopyNet for TM

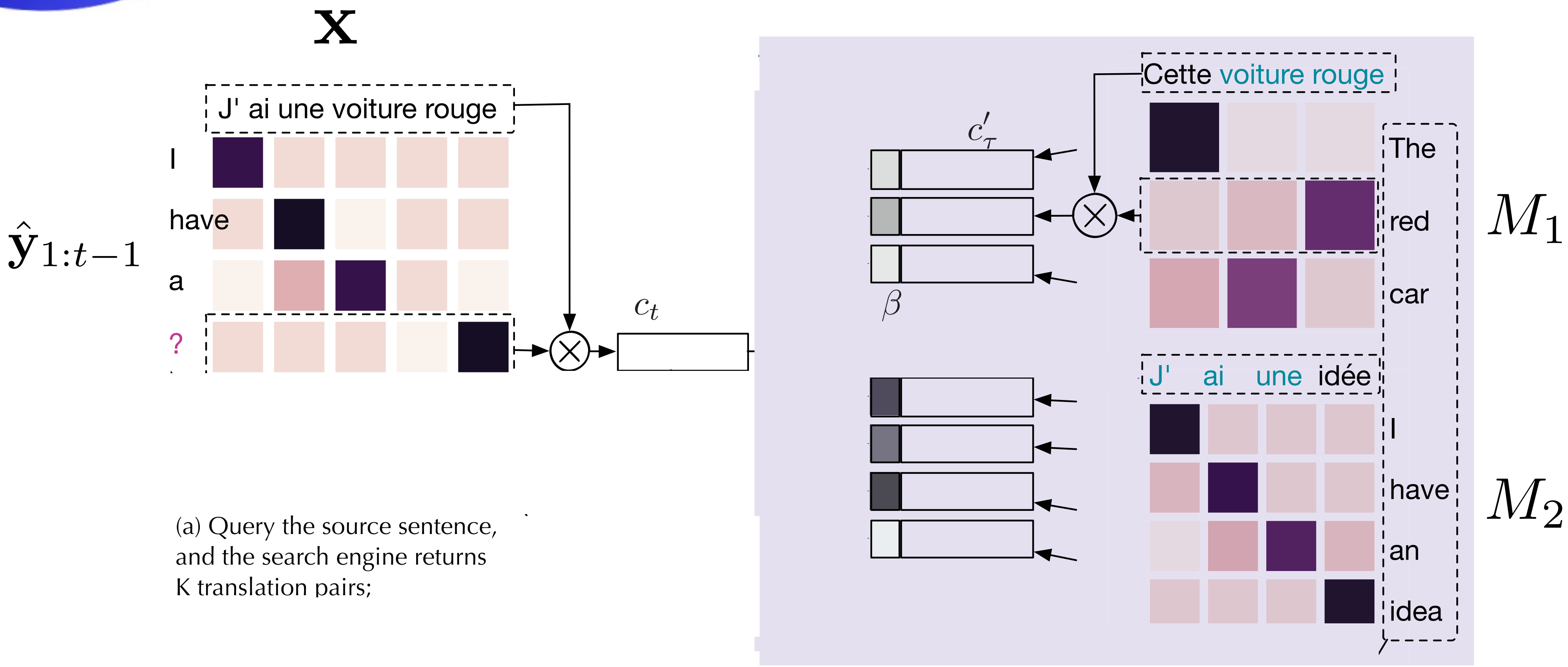
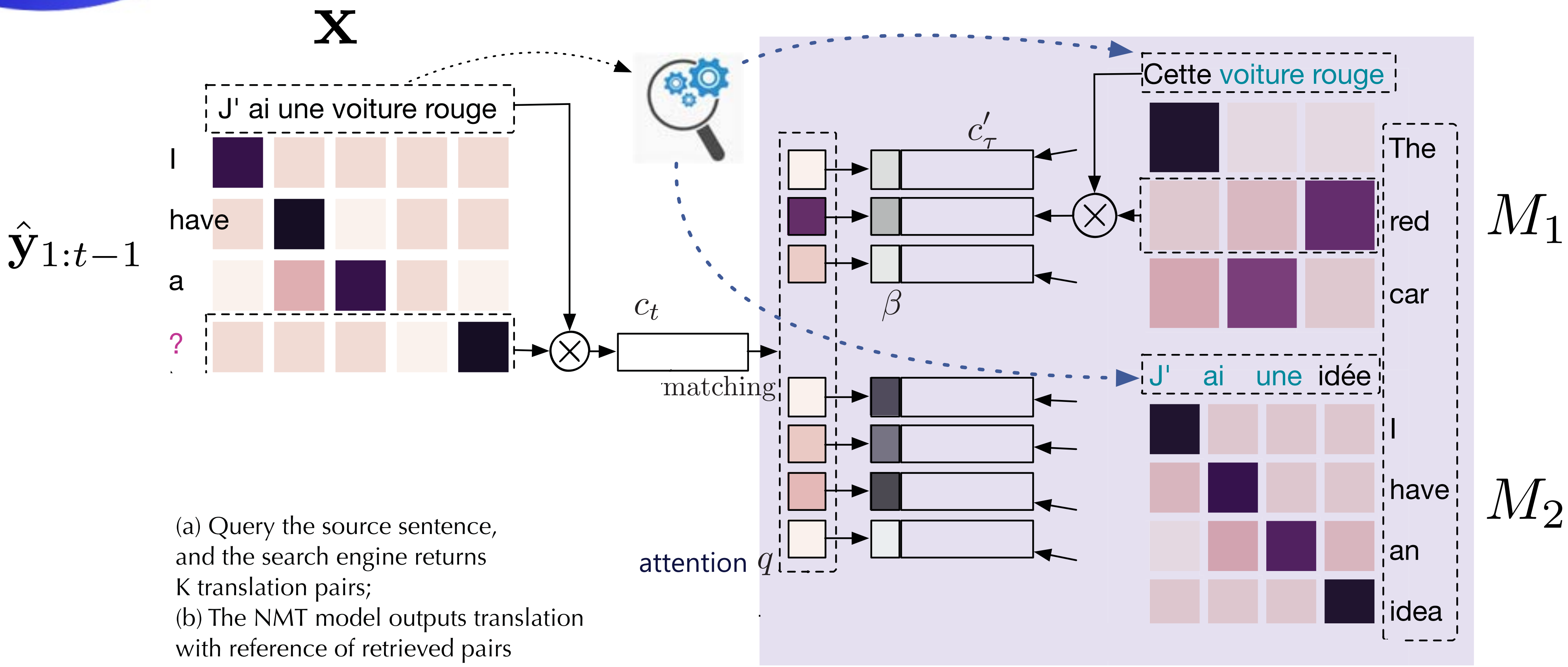


Fig. Credit: Jiatao Gu, Yong Wang, Kyunghyun Cho, Victor O.K. Li. Search Engine Guided Neural Machine Translation. AAAI18.

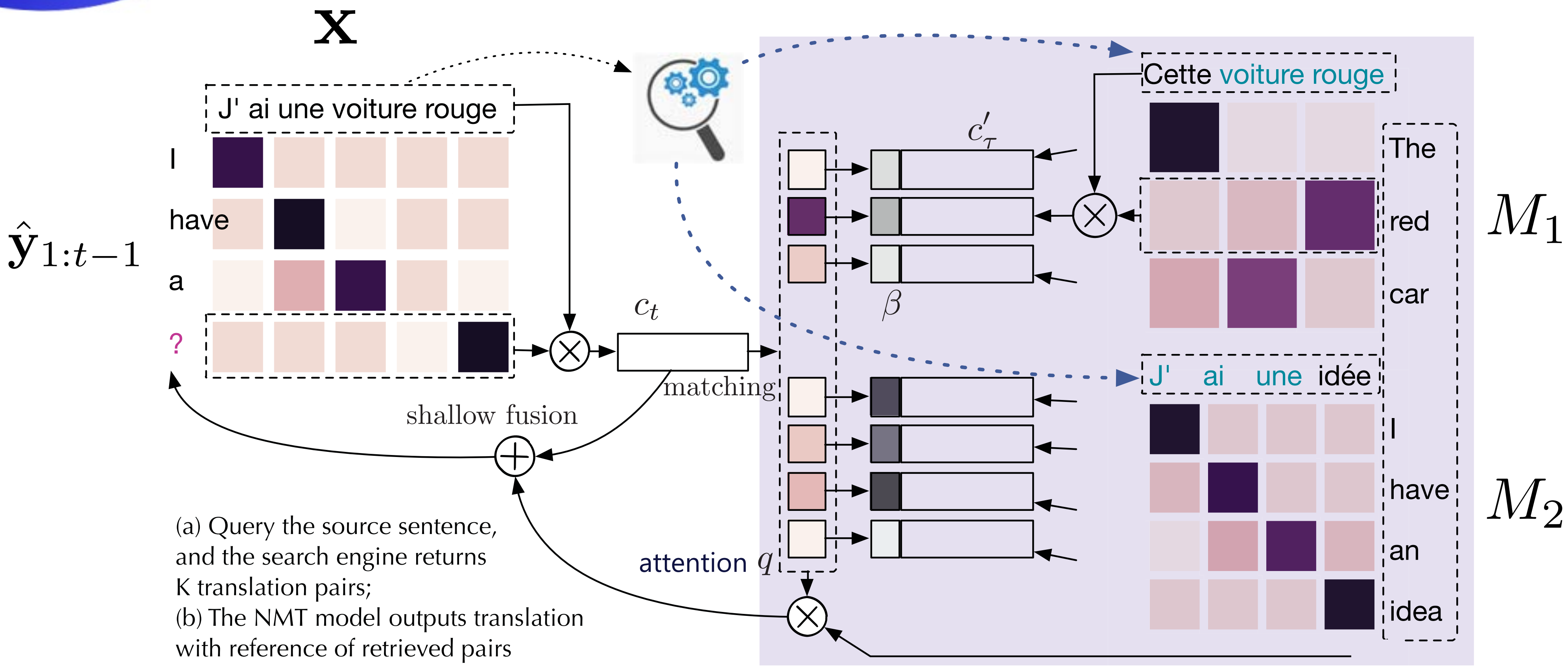
Unified Model: CopyNet for TM



(a) Query the source sentence,
and the search engine returns
K translation pairs;
(b) The NMT model outputs translation
with reference of retrieved pairs

Fig. Credit: Jiatao Gu, Yong Wang, Kyunghyun Cho, Victor O.K. Li. Search Engine Guided Neural Machine Translation. AAAI18.

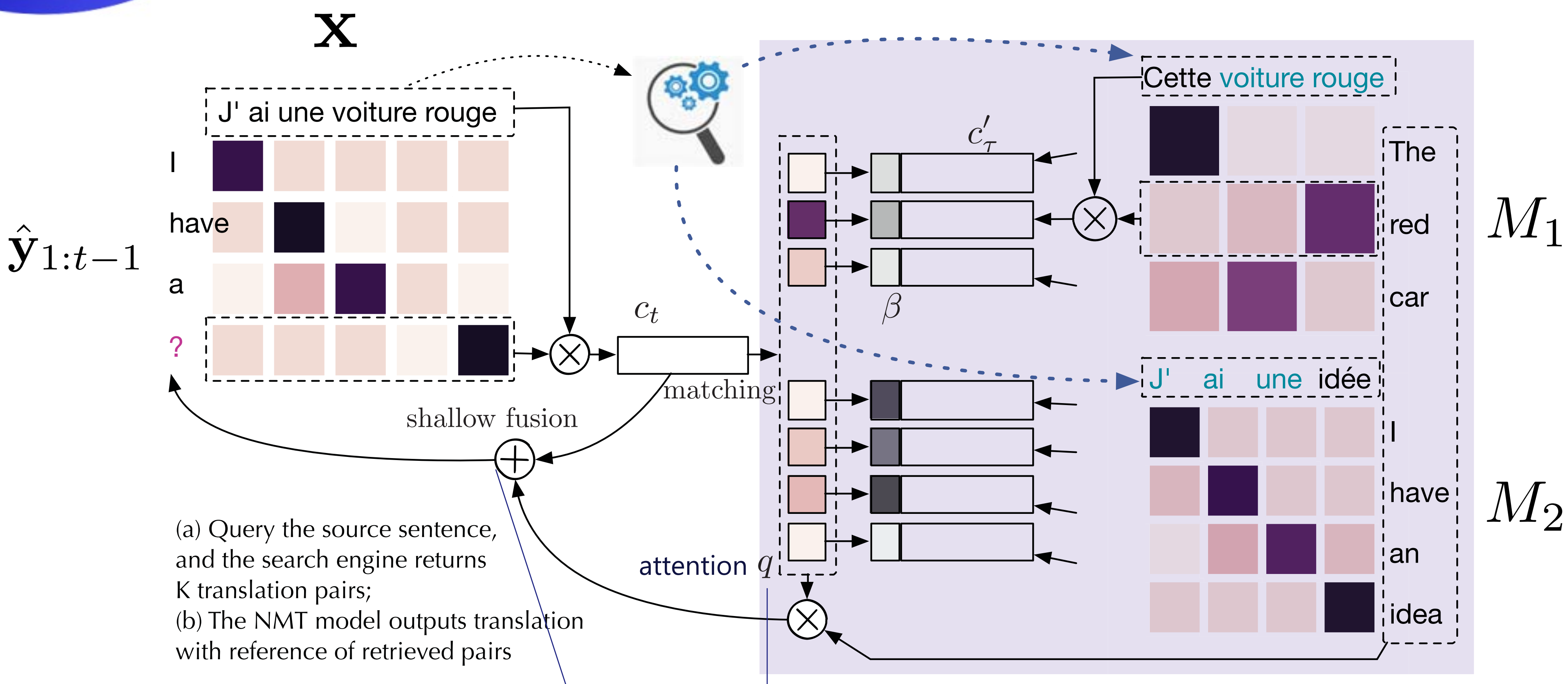
Unified Model: CopyNet for TM



(a) Query the source sentence, and the search engine returns K translation pairs;
(b) The NMT model outputs translation with reference of retrieved pairs

Fig. Credit: Jiatao Gu, Yong Wang, Kyunghyun Cho, Victor O.K. Li. Search Engine Guided Neural Machine Translation. AAAI18.

Unified Model: CopyNet for TM



$$p(y_t|\mathbf{x}, \hat{\mathbf{y}}_{1:t-1}, M; \theta) = \zeta_t(\theta) \times p_{\text{copy}}(y_t; \theta) + (1 - \zeta_t(\theta))p(y_t|\mathbf{x}, \hat{\mathbf{y}}_{1:t-1}; \theta)$$

Fig. Credit: Jiatao Gu, Yong Wang, Kyunghyun Cho, Victor O.K. Li. Search Engine Guided Neural Machine Translation. AAAI18.

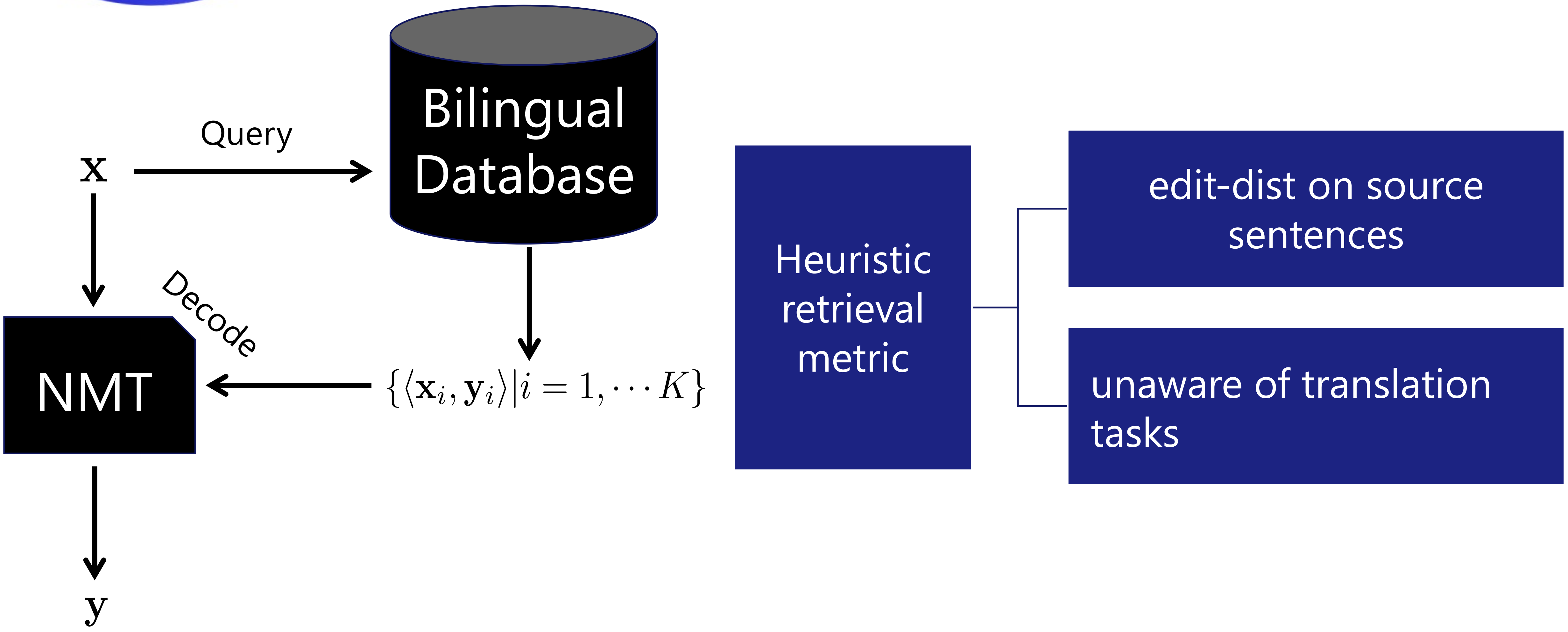
Pros and Cons of CopyNet for TM

- Pros
  - Model capacity is good
  - Translation quality is good

- Cons
  - Encoding all words from tm needs considerable GPU memory
  - Attention over all target words from tm is not efficient

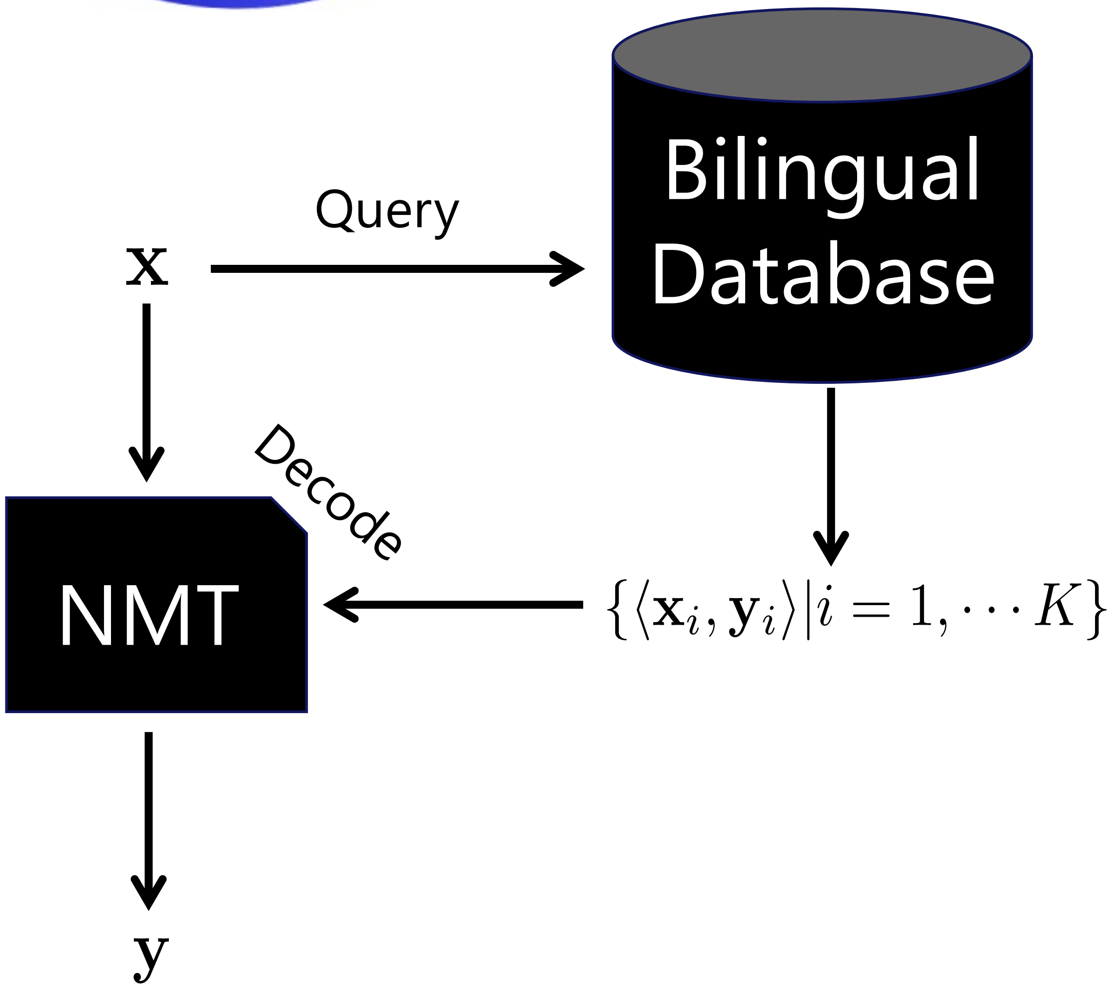
- Improvements
  - A compact graph structure to organize translation memory (Xia et al., 2019)
  - Customized TM augmented model with a small translation memory (He et al., 2021)

Limitations in conventional TM framework



Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Limitations in conventional TM framework

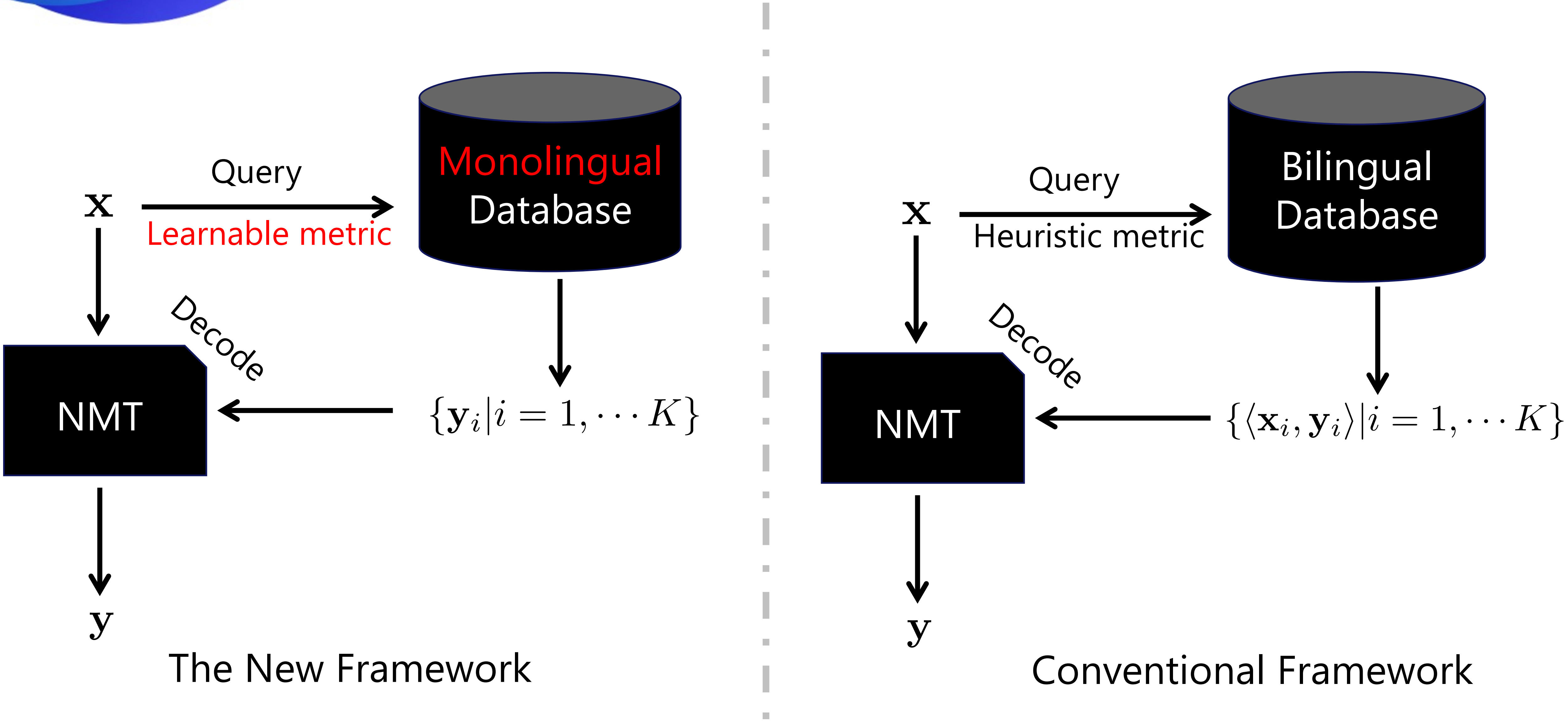


Limiting Applications

- Search over bilingual database
- Bilingual data is too expensive

Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Monolingual translation memory



Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Challenge

Query in Chinese

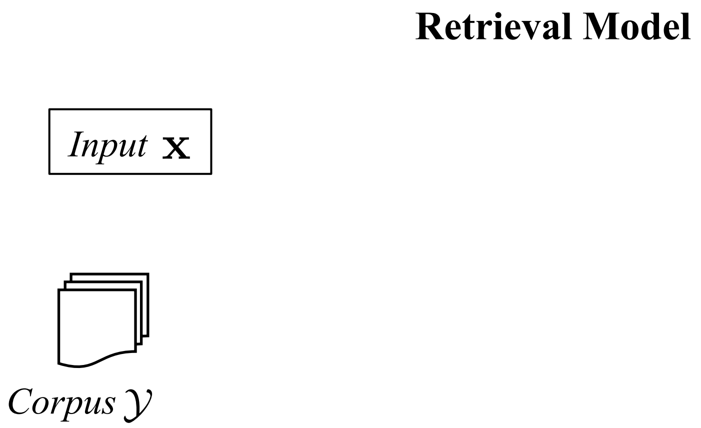
获取 或 设置 与 批注 关联 的 对象

Cross-lingual retrieval

gets an object that is associated with the annotation label
obtains an annotated label from an object
… …

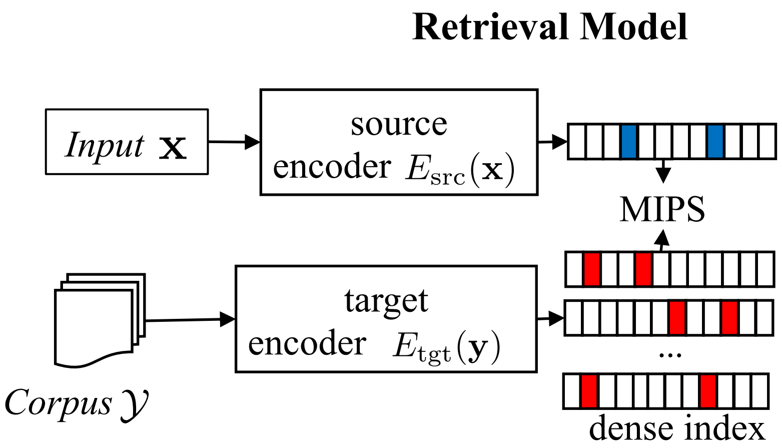
The database in English

Cross-lingual Retrieval Metric Definition



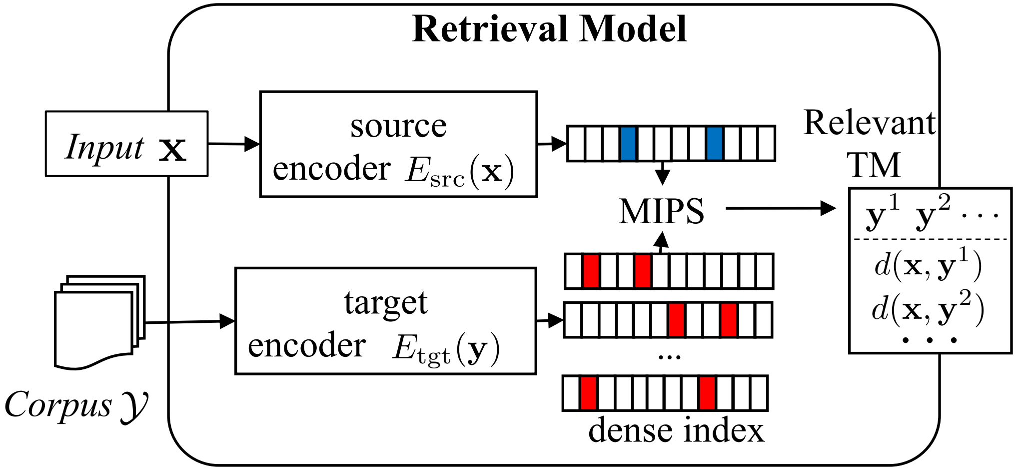
Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Cross-lingual Retrieval Metric Definition



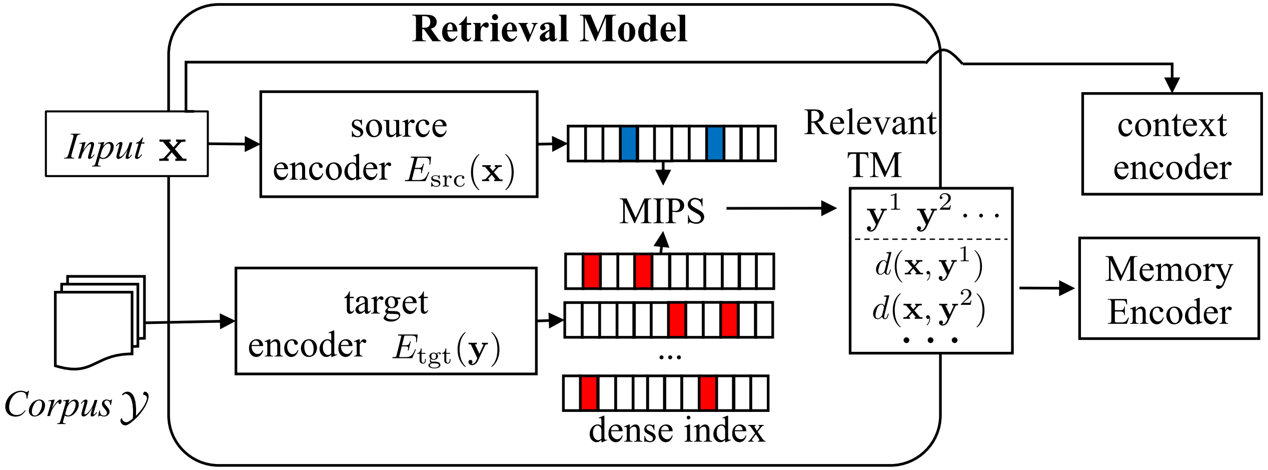
Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Cross-lingual Retrieval Metric Definition



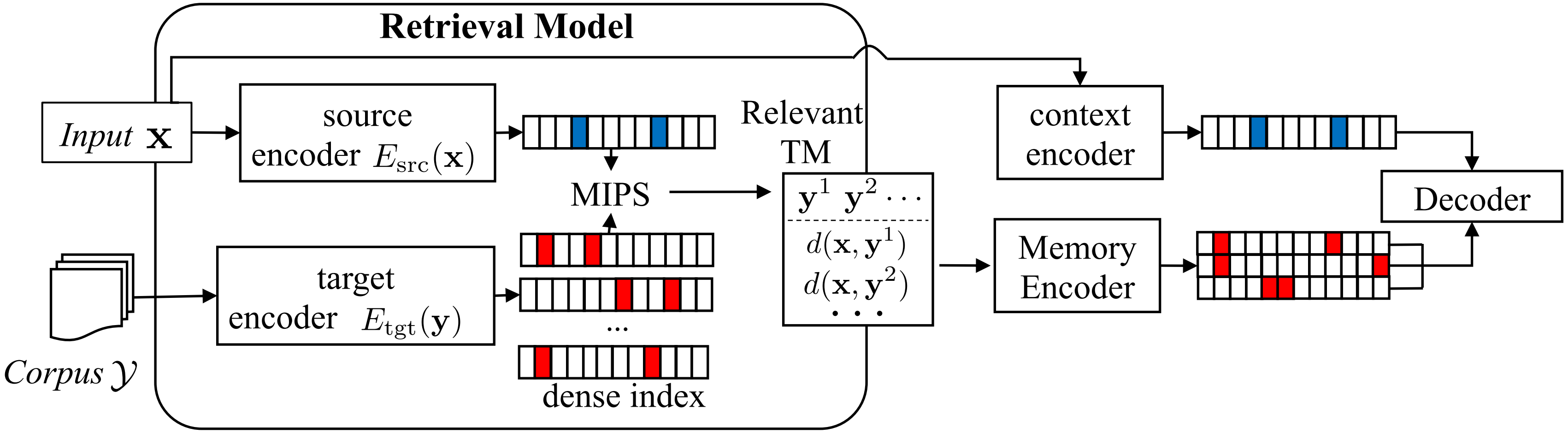
Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Retrieval augmented translation model



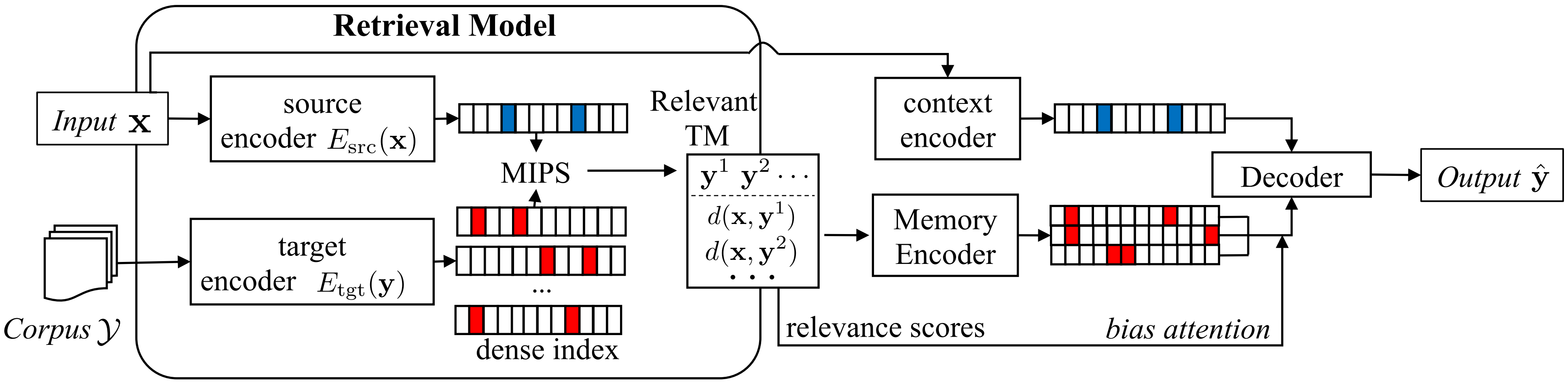
Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Retrieval augmented translation model



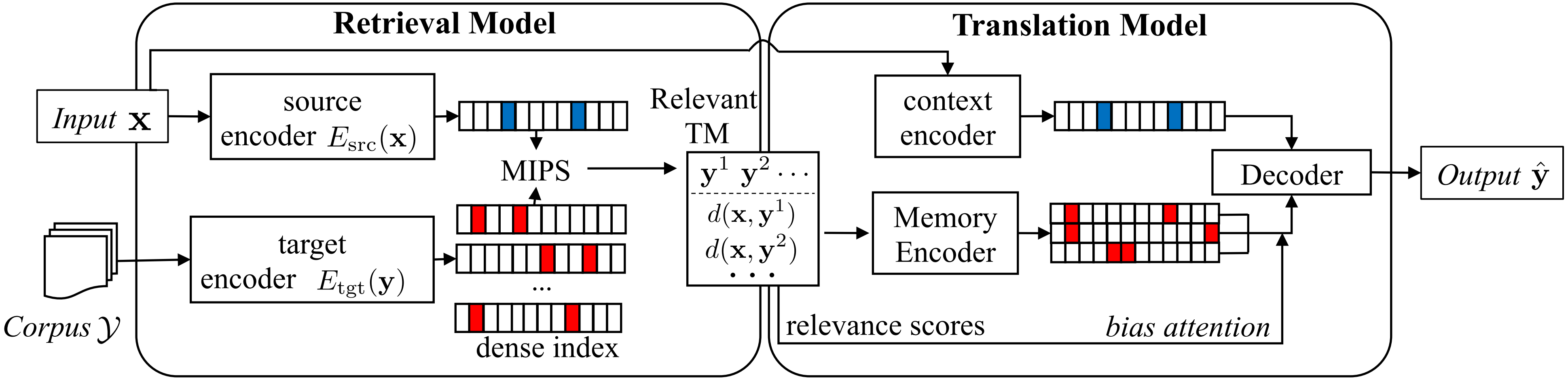
Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Retrieval augmented translation model



Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

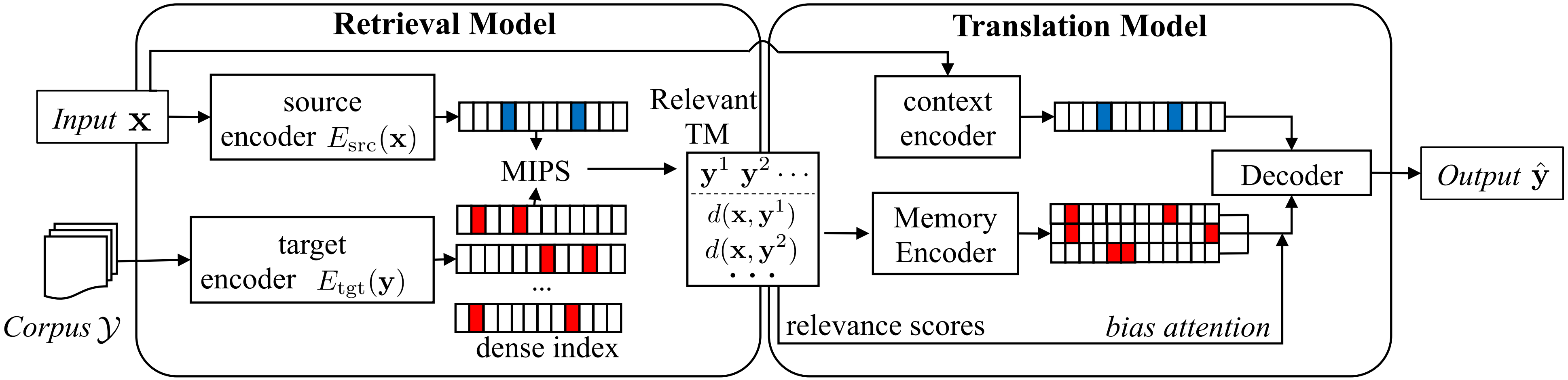
Joint learning retrieval and translation models



$$\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y} \rangle} \log P(\mathbf{y} | \mathbf{x}, \mathbf{y}^1, d_1, \cdots, \mathbf{y}^k, d_k; \theta)$$

Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Joint learning retrieval and translation models



$$\max_{\theta} \sum_{\langle \mathbf{x}, \mathbf{y} \rangle} \log P(\mathbf{y} | \mathbf{x}, \mathbf{y}^1, d_1, \cdots, \mathbf{y}^k, d_k; \theta)$$

- Challenge: joint training by MLE leads to a trivial retrieval metric.
    - Solution: two pre-training subtasks as regularization

Deng Cai, Yan Wang, Huayang Li, Wai Lam, Lemao Liu. Neural Machine Translation with Monolingual Translation Memory. ACL21.

Pros and Cons of monolingual translation memory

- Pros
  - The metric is optimized towards translation quality
  - The framework is general to any translation scenarios because monolingual database is easy to access
- Cons
  - Joint training the retrieval metric and translation model requires additional overheads in computation

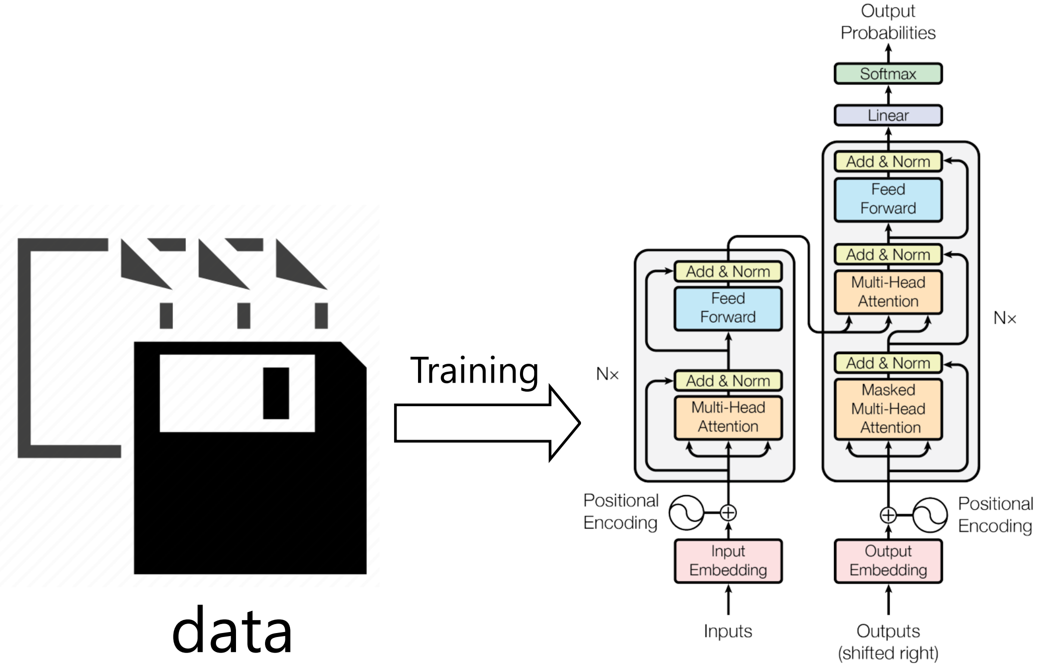
Outline

- Background and Introduction
- Language Modeling
- Open-Domain Dialogue Systems
- Neural Machine Translation
  - Motivation
  - TM-augmented NMT Framework
  - TM-augmented Models
    - Standard model
    - Dual model
    - Unified model
- **Conclusion and Outlook**

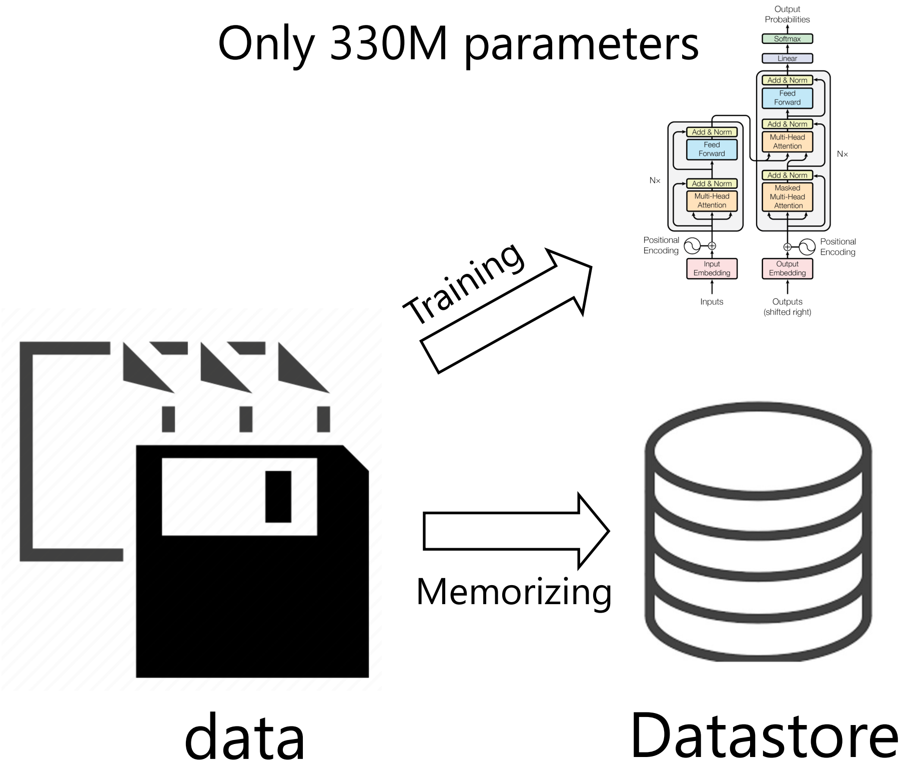
Advantages of retrieval-augmented model

- Compact model with less parameters
  - The knowledge is not implicitly stored in model parameters but in memory

T5 with 11318M parameters



Only 330M parameters



Advantages of retrieval-augmented model

- Better interpretability
  - Some prediction results can be explained through the cues in memory.

From Wikipedia, the free encyclopedia

This article is about the capital city of Spain. For the autonomous community, see Community of (disambiguation).

Madrid (/məˈdrɪd/ *mə-DRID*, Spanish: [maˈðɾið])[n. 1] is the capital and most populous city of Spain. The city has almost 3.4 million[7] inhabitants and a metropolitan area population of approximately 6.7 million. It is the second-largest city in the European Union (EU), surpassed only by Berlin in its administrative limits, and its monocentric metropolitan area is the second-largest in the EU, surpassed only by Paris.[8][9][10] The municipality covers 604.3 km^2 (233.3 sq mi) geographical area.[11]

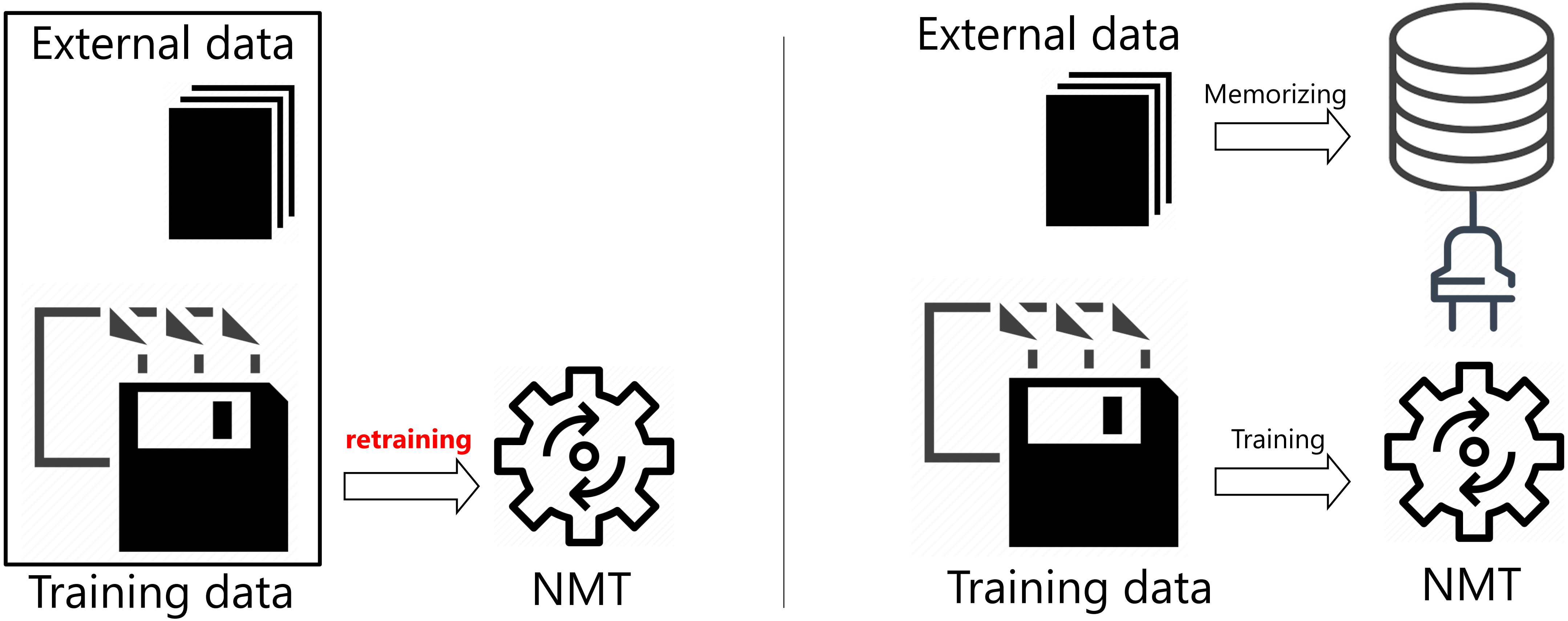
Memory

SIGIR 2022 will be held in Mardrid , **which is the capital and the largest city of Spain .**

Text Generation by retrieval augmented LM

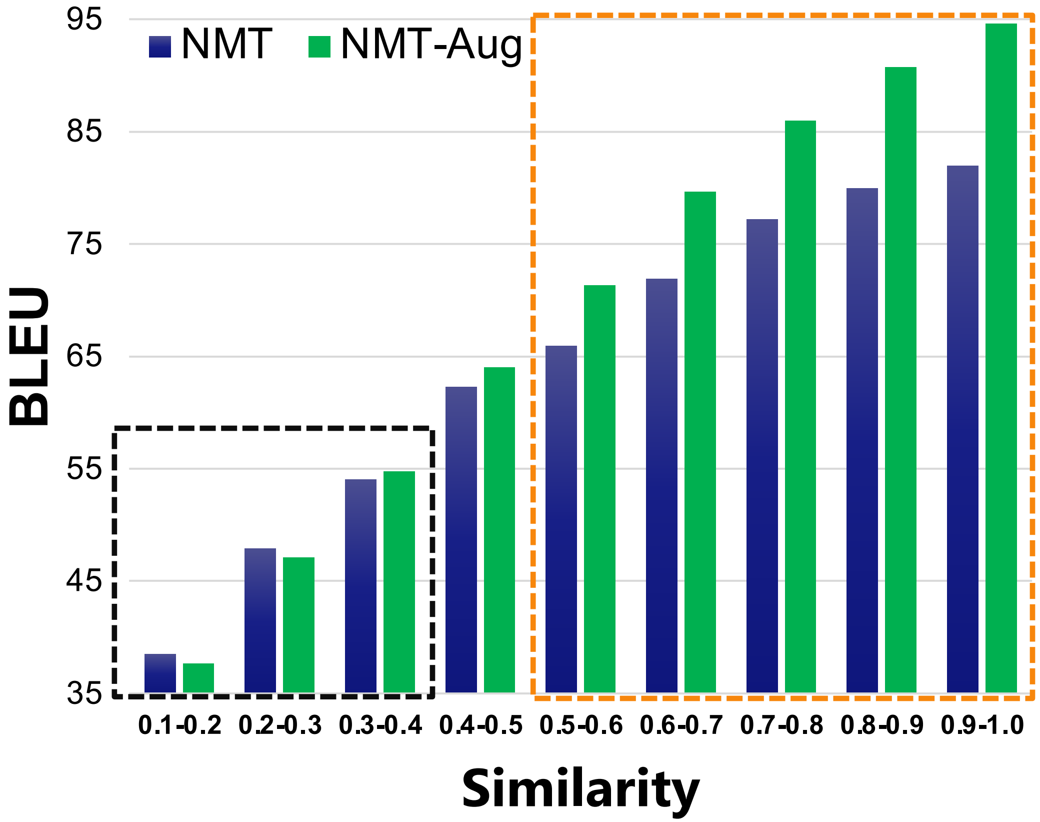
Advantages of retrieval-augmented model

- Better scalability
  - External data can be used as memory in a plug-and-play manner, leading to great scalability



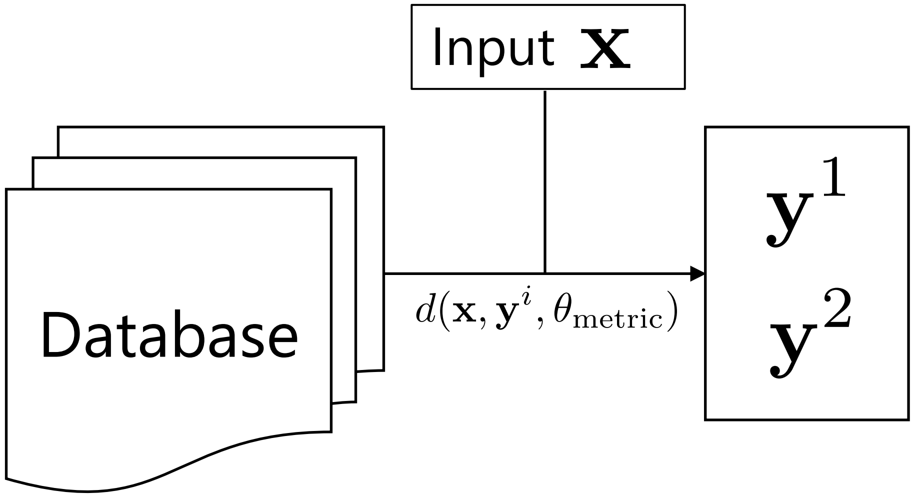
Future Directions

- Retrieval sensitivity
  - Substantial gains for test sentences with high quality memory
  - No gains for those with low quality memory
  - How to alleviate the sensitivity issue?

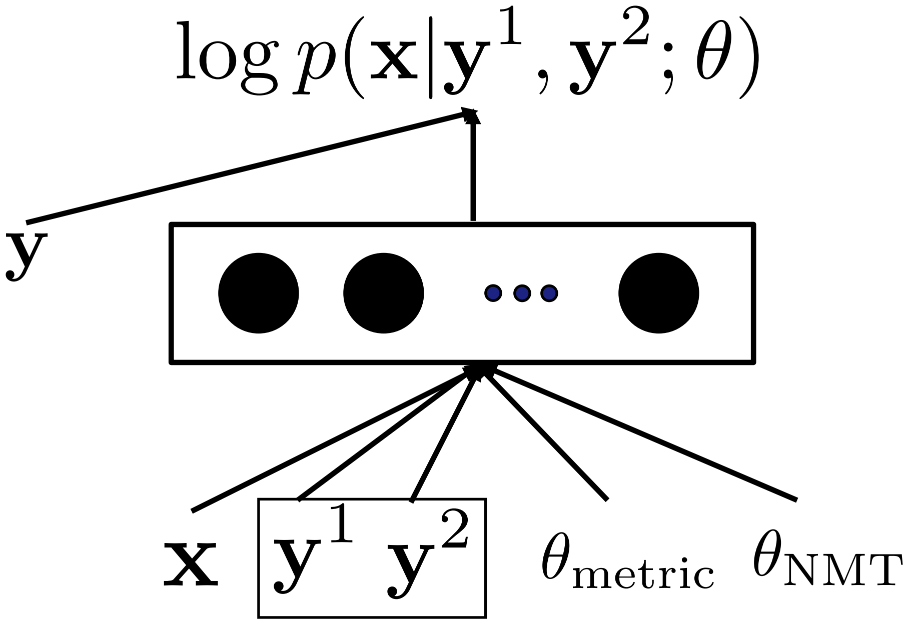


Future Directions

- Gap when jointly learning a retrieval metric towards translation quality
  - Global retrieval: retrieval is globally conducted in the entire database
  - Local optimization: the parameters are locally optimized with respect to a tiny fraction of database.



Global Retrieval



Local optimization

Future Directions

Tencent AI Lab

- Retrieval from multi-modality database
  - Most existing works focus on generation models augmented by text memory
  - Multi-modality information can provide complementary information for text generation

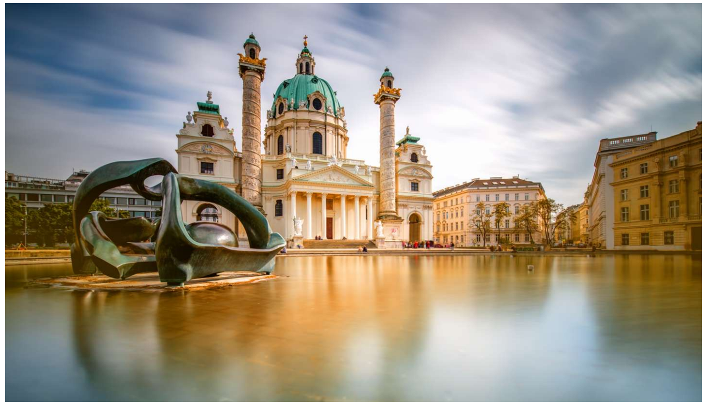
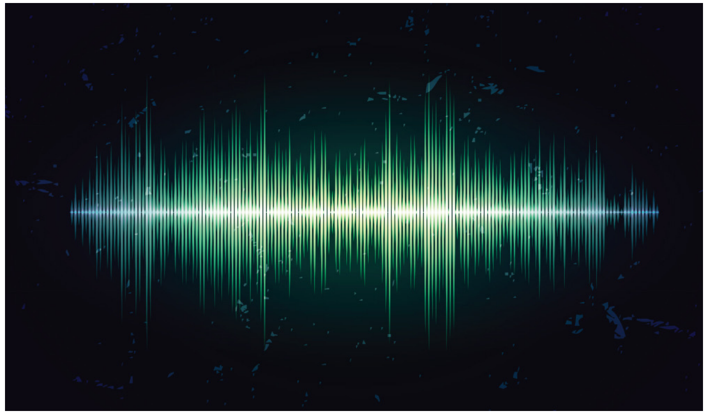


Image database



Audio database

Video database

Q&A

Tencent AI Lab

Thanks