LLM for Search Engines

Chenyan Xiong

11-667

Disclaimer:

All the discussions in this lecture about commercial systems are based on public information, plus educated guesses from the instructor

Outline

Overview of Modern Information Retrieval Systems

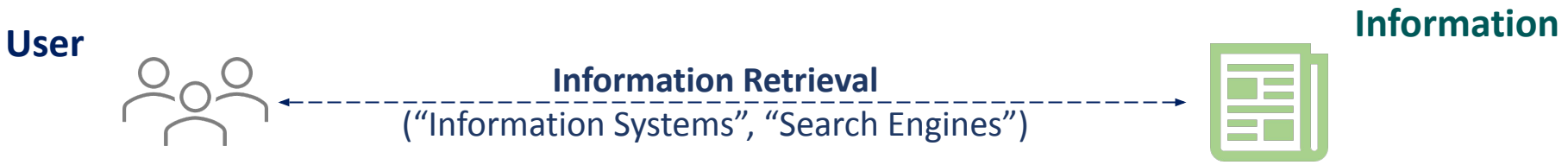
- An example search component being updated by LLMs
- Glances of other components using LLMs

Dense Retrieval, a revolution of search with LLMs

- End-to-end learned retrieval
- Notable extensions
- Pretraining retrieval representations

Overview of Modern Information Retrieval Systems

Information Retrieval Systems



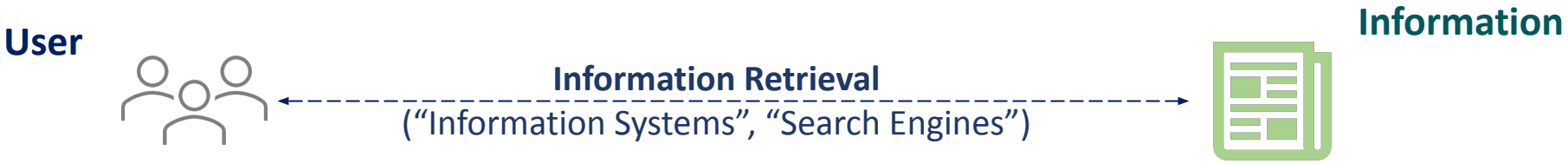
General Definition: Anything system that finds information user needed

- Search Systems, QA Systems, Recommendation Systems, etc.

Specific Definition: Search engines that retrieve documents for user queries

- Explicit Query: User expressed information needs via texts, audios, or conversations
- Target Document: Satisfy user information needs by finding relevant documents

Information Retrieval Systems



General Definition: Anything system that finds information user needed

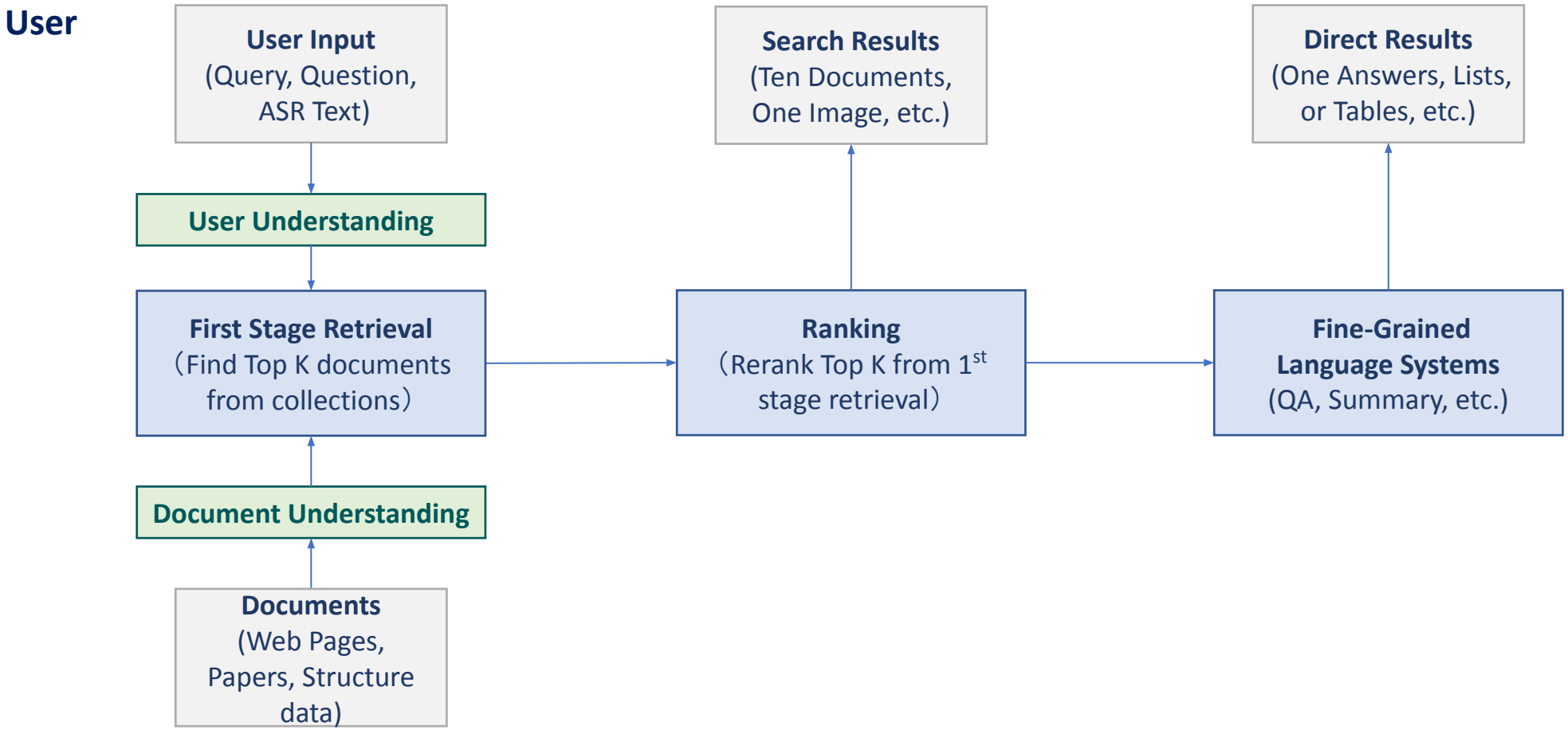
- Search Systems, QA Systems, Recommendation Systems, etc.

Specific Definition: Search engines that retrieve documents for user queries

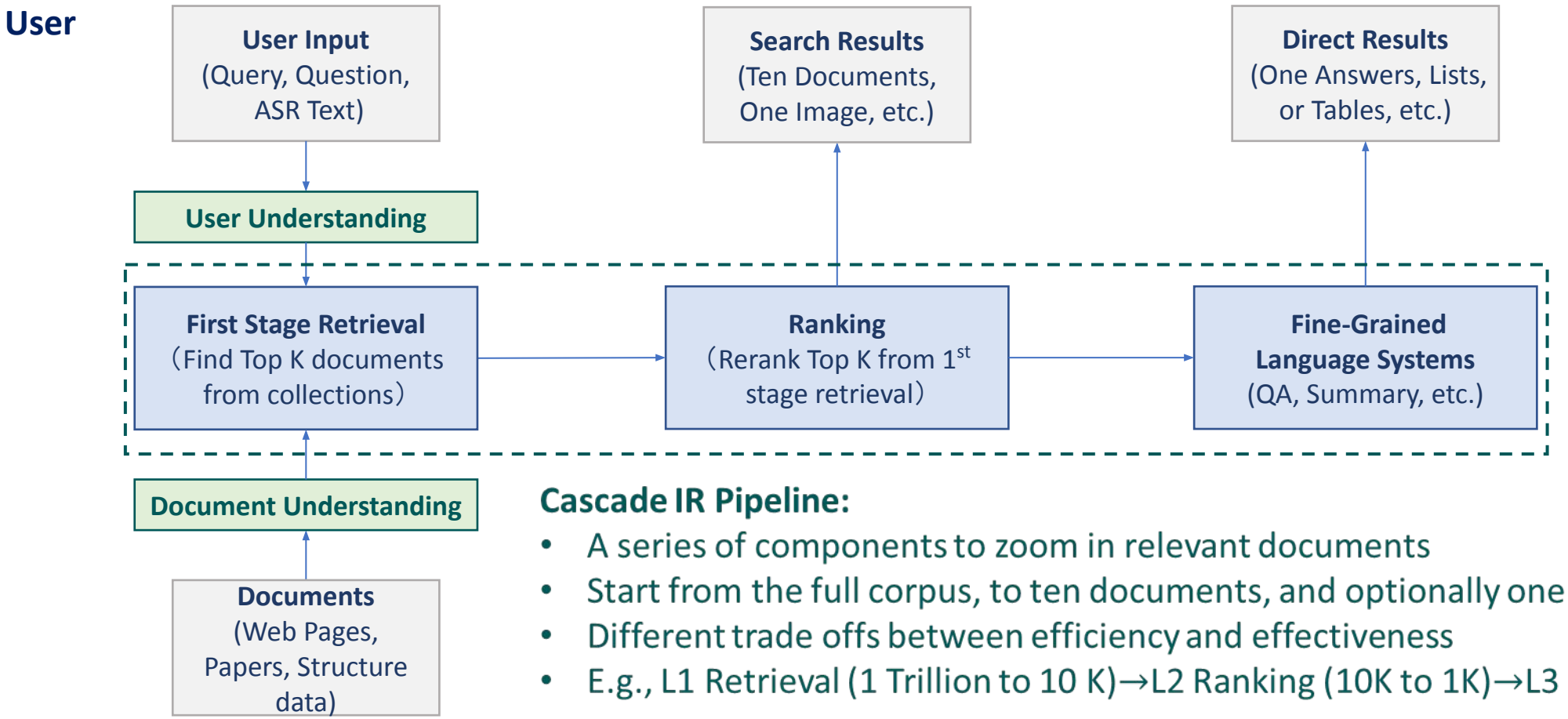
- Explicit Query: User expressed information needs via texts, audios, or conversations
- Target Document: Satisfy user information needs by finding relevant documents

One of the most popular AI applications in past decades

The General Framework of Search Engines



The General Framework of Search Engines



Cascade IR Pipeline:

- A series of components to zoom in relevant documents
- Start from the full corpus, to ten documents, and optionally one answer
- Different trade offs between efficiency and effectiveness
- E.g., L1 Retrieval (1 Trillion to 10 K)→L2 Ranking (10K to 1K)→L3 Ranking (1K to 100)

Outline

Overview of Modern Information Retrieval Systems

- **An example search component being updated by LLMs**
- Glances of other components using LLMs

Dense Retrieval, a revolution of search with LLMs

- End-to-end learned retrieval
- Notable extensions
- Pretraining retrieval representations

Ranking Models

Given a query q, order a set of candidate document D so that relevant document d^+ is on top of the rank

Often done by

- learning a ranking score function $f(q, d)$
- From relevance labels (q, d^+)

Ranking Models

Given a query q, order a set of candidate document D so that relevant document d^+ is on top of the rank

Often done by

- learning a ranking score function $f(q, d)$
- From relevance labels (q, d^+)

With statistics machine learning models:

$$f(q, d) = \text{XGBoost}(\phi(q, d))$$

- $\phi(q, d)$: Ranking features, e.g.,
  - word overlaps between q and d
  - BM25 retrieval scores
  - Page rank of d
  - Freshness of d
- A learned combination of features that manually designed to capture q-d relevance

Ranking Models

Given a query q, order a set of candidate document D so that relevant document d^+ is on top of the rank

Often done by

- learning a ranking score function $f(q, d)$
- From relevance labels (q, d^+)

With neural ranking models:

$$f(q, d) = \mathrm{NN}(\mathrm{M}_{qd})$$
$$\mathrm{M}_{qd}^{ij} = \mathrm{sim}(q^i, d^j)$$

- M_{qd}: The Translation matrix between q and d.
  - Each element is the embedding similarity between a query term q^i and a doc term d^j
- NN: A specifically designed network to pool term level similarities to q-d relevance
- Learning to model soft matches between q d terms, e.g., “pdf” and “reader”

Ranking BERT

- BERT models the relevance of (q, d) by simple classification [1]:

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

- A MLP layer after the last layer's [CLS] representation to learn binary predictions: relevant/irrelevant

Ranking BERT

BERT models the relevance of (q, d) by simple classification [1]:

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

- A MLP layer after the last layer's [CLS] representation to learn binary predictions: relevant/irrelevant

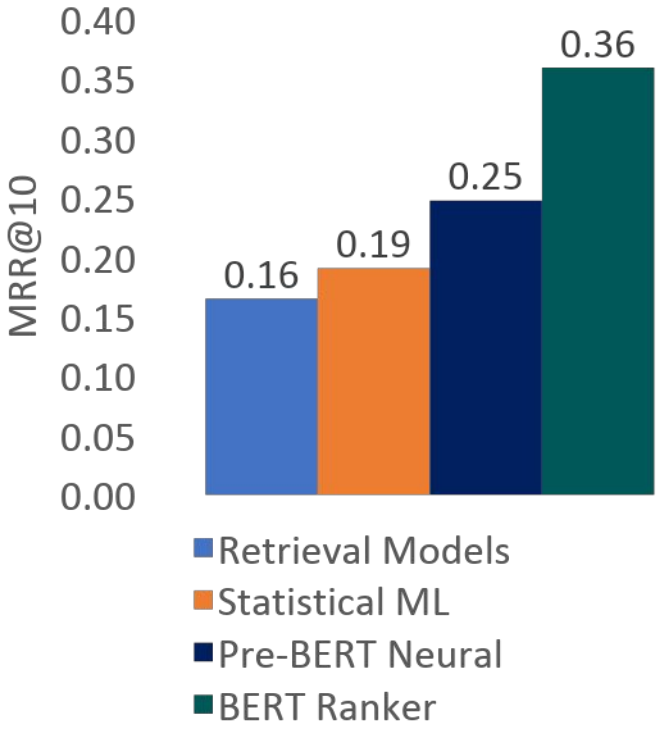


Figure 1: Ranking Performance on MS MARCO Passage Ranking Test [2]

Task: Rank answer passages for Bing questions from BM25 top 1000

- ~1M queries/labels from MS MARCO
- ~10M passages

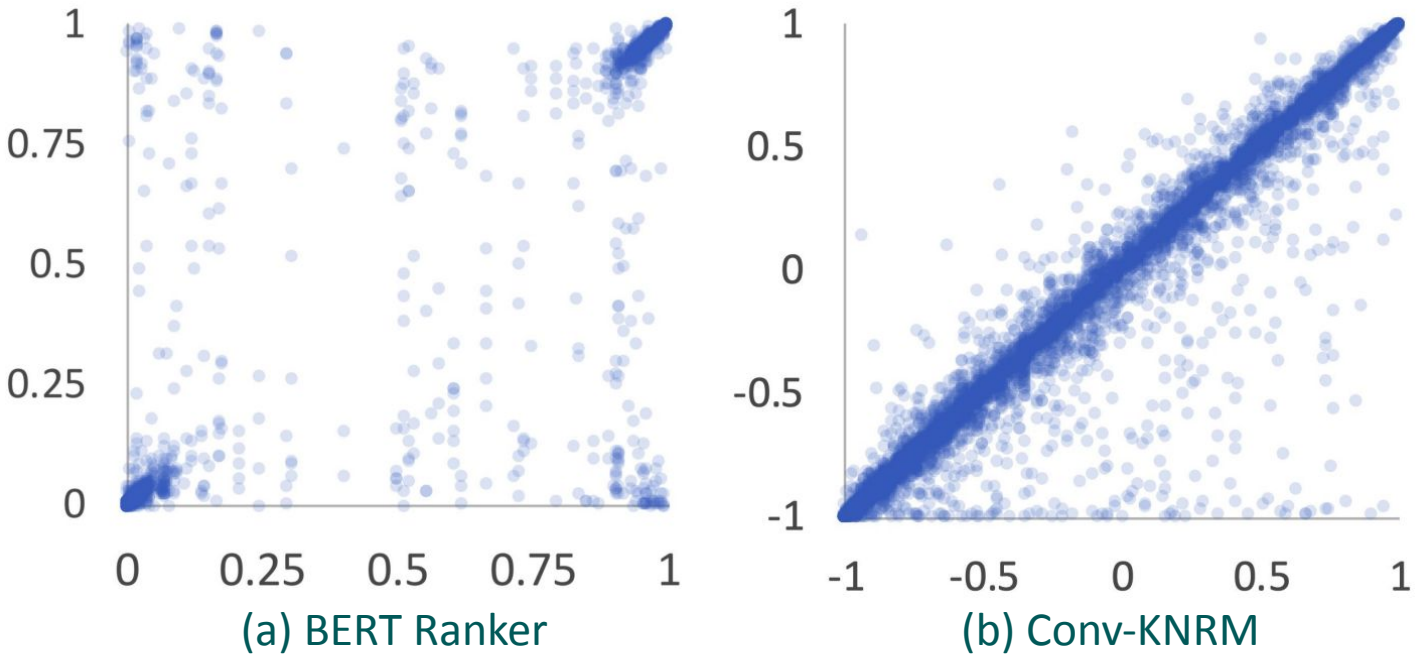
Significant gains from retrieval to neural ranker to BERT ranker

- Relevant doc moved from position 6 (1/0.16 MRR) to 4 to above 3

Also require far fewer supervisions

[1] Nogueira Et al. "PASSAGE RE-RANKING WITH BERT." Arxiv 2019
[2] Qiao Et al. "Understanding the Behaviors of BERT in Ranking". Arxiv 2019

Ranking BERT



(a) BERT Ranker (b) Conv-KNRM

Figure 2: Ranking score of BERT and Pre-BERT Neural ranker (n-gram soft match) before (x) and after (y) removing a random document term [2]

BERT is more confident:

- Most produced ranking scores are close to 0 or 1

BERT is more global:

- Removing most document terms does not matter
- But some dramatically changed BERT's decision
- Most of these terms are crucial for relevance matches

ChatGPT for Ranking

Challenging to ask ChatGPT to generate a reasonable numerical ranking score for each document

One solution is to ask ChatGPT to rank a set of documents for a query

- E.g., input: q + p1, p2, p3, p4, ask ChatGPT to rank p1-p4

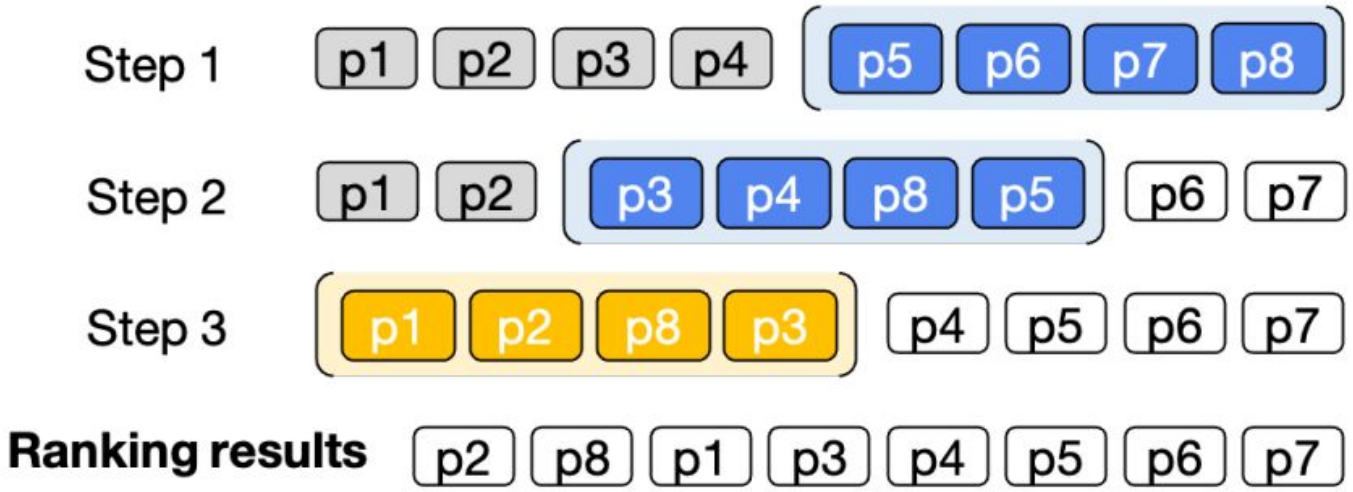


Figure 3: “Bubble Sorting” documents by prompting LLMs [3]

ChatGPT for Ranking

Challenging to ask ChatGPT to generate a reasonable numerical ranking score for each document

One solution is to ask ChatGPT to rank a set of documents for a query

- E.g., input: q + p1, p2, p3, p4, ask ChatGPT to rank p1-p4

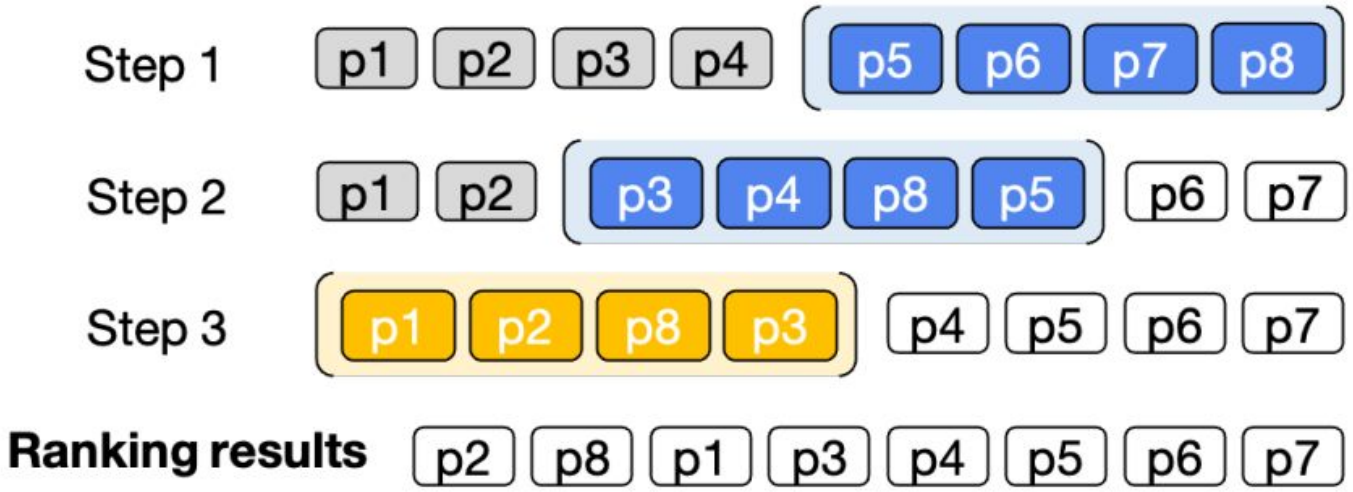
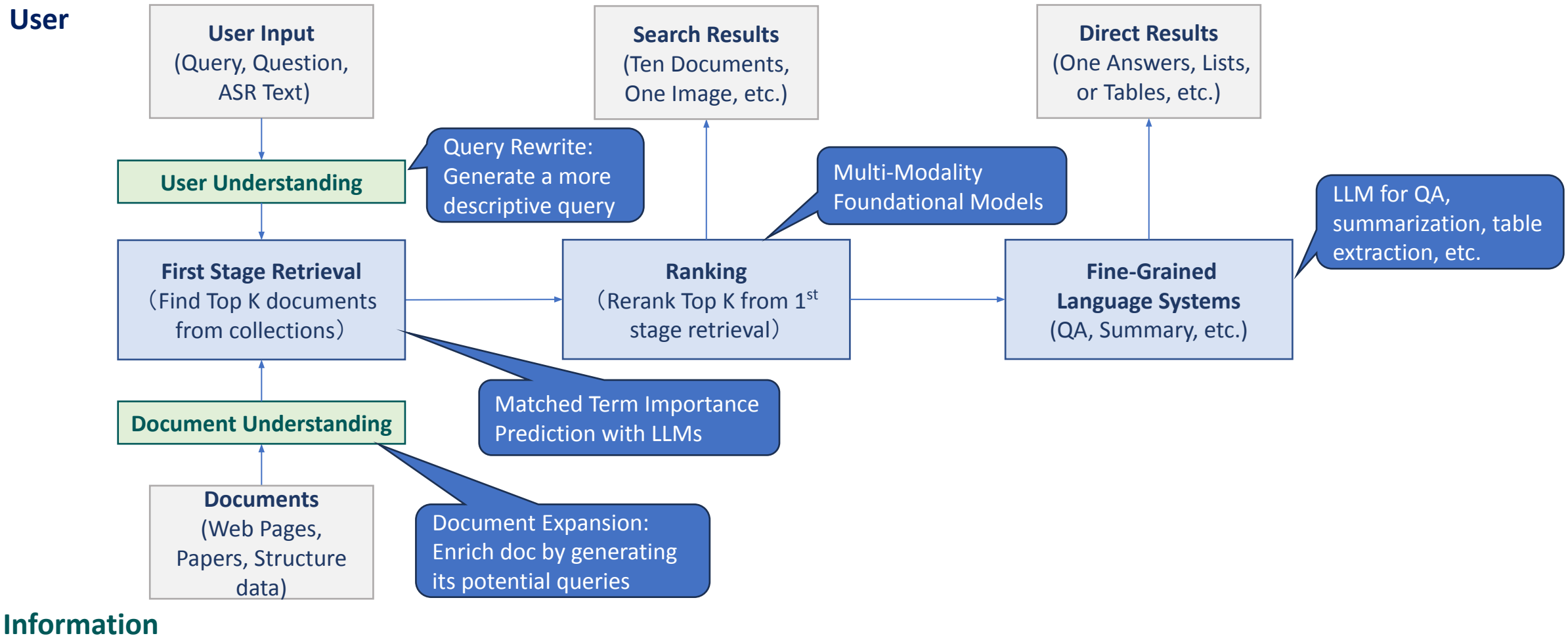


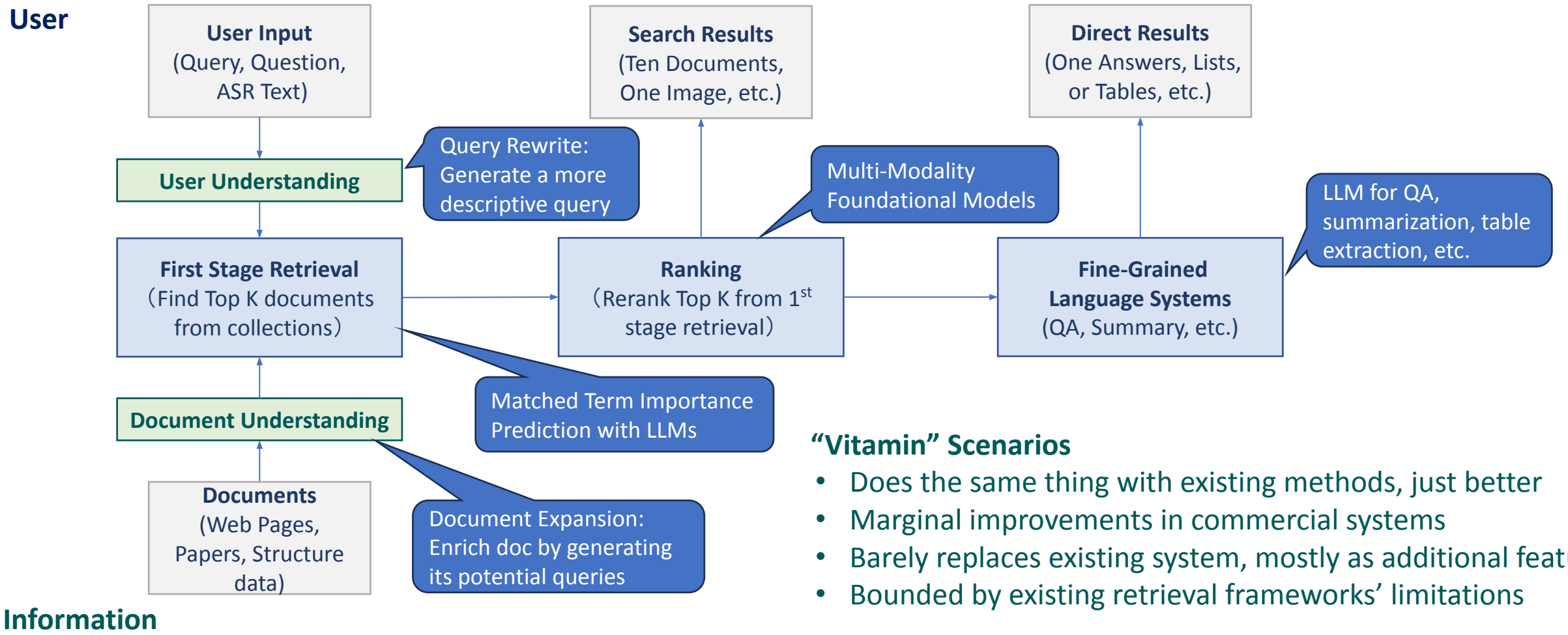
Figure 3: "Bubble Sorting" documents by prompting LLMs [3]

- About 3-5% accuracy+ from GPT-4 over T5 ranker
- Can be distilled to smaller models
- As search mainly cares about top positions, no need to bubble sort all

LLMs in Many Places of Search Engines



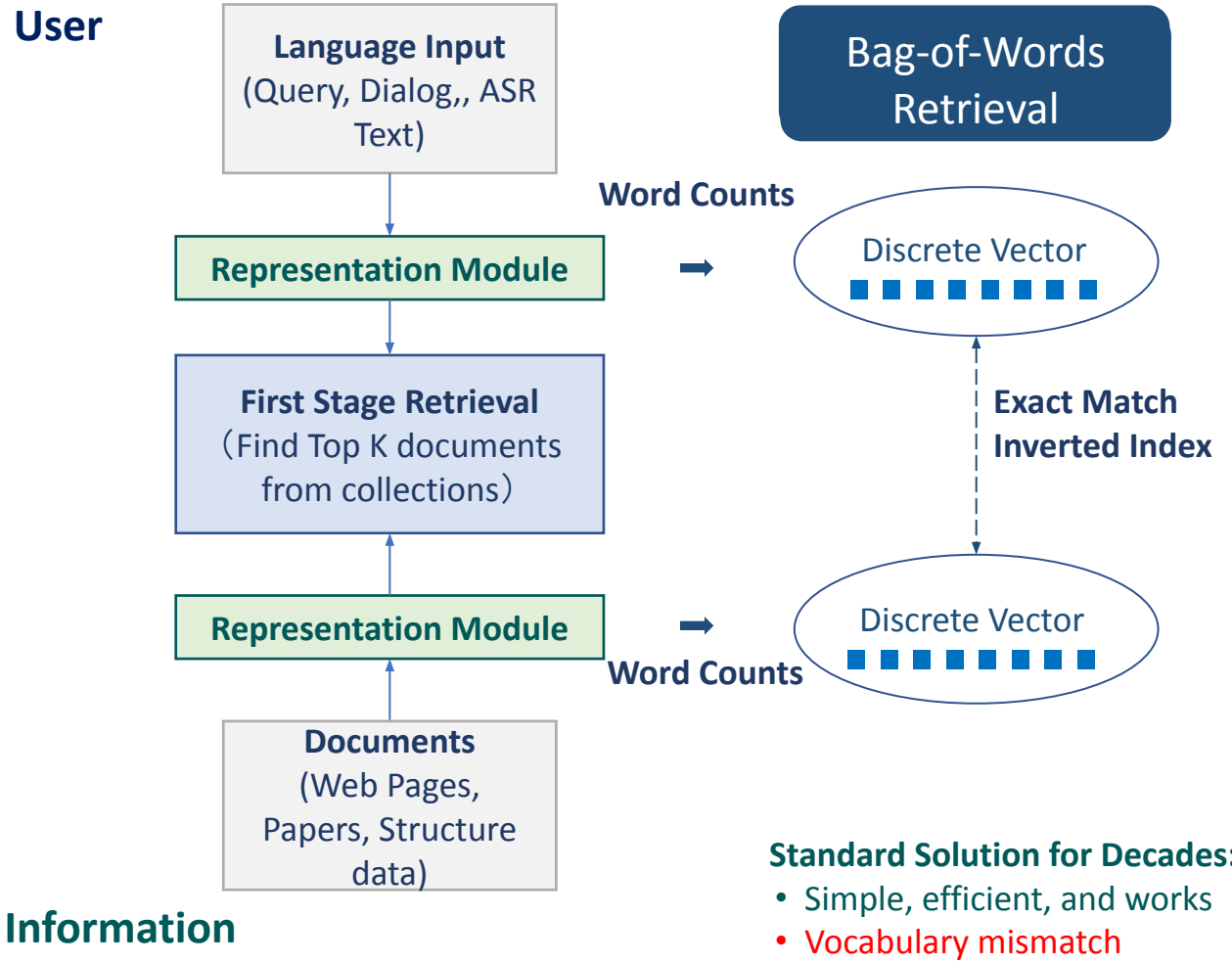
LLMs in Many Places of Search Engines



"Vitamin" Scenarios

- Does the same thing with existing methods, just better
- Marginal improvements in commercial systems
- Barely replaces existing system, mostly as additional features
- Bounded by existing retrieval frameworks' limitations

Biggest Pain Points in Previous Search Systems



Standard Solution for Decades:

- Simple, efficient, and works
- Vocabulary mismatch

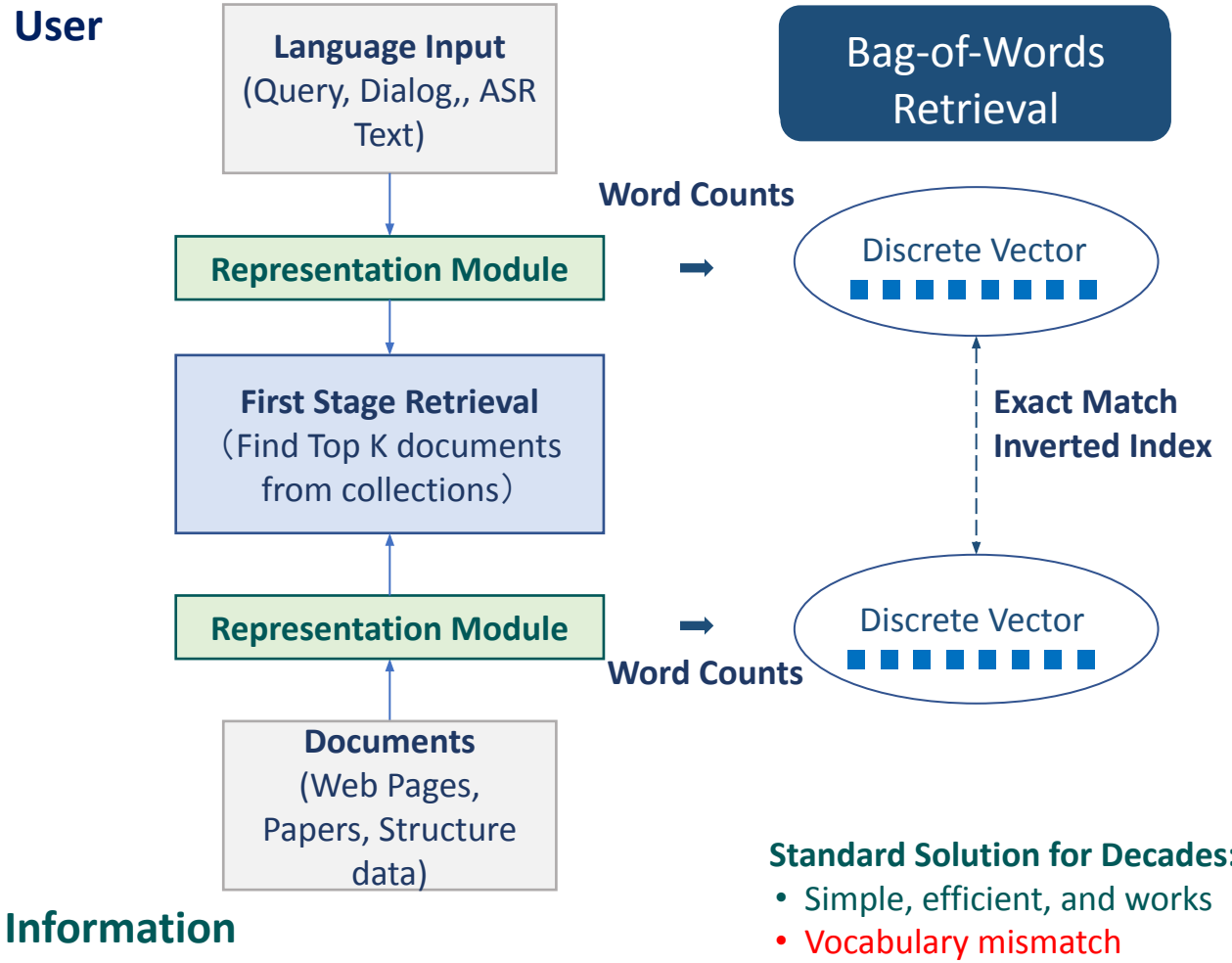
Bag-of-Words first stage retrieval

- Retrieve documents that contain exact query terms

Intrinsic challenge: Vocabulary Mismatch

- Query and Relevant documents may not have term overlap

Biggest Pain Points in Previous Search Systems



Standard Solution for Decades:

- Simple, efficient, and works
- Vocabulary mismatch

Bag-of-Words first stage retrieval

- Retrieve documents that contain exact query terms

Intrinsic challenge: Vocabulary Mismatch

- Query and Relevant documents may not have term overlap

A huge pain point for IR systems

- Discrete representations hard to optimize
- Bounds ranking performance
- Very ways to mitigate, making systems complicated
- More effective way: expand document with clicked queries
  - Requires ton of user traffics
  - Impossible for public domain
  - Lower coverage even in commercial search

Outline

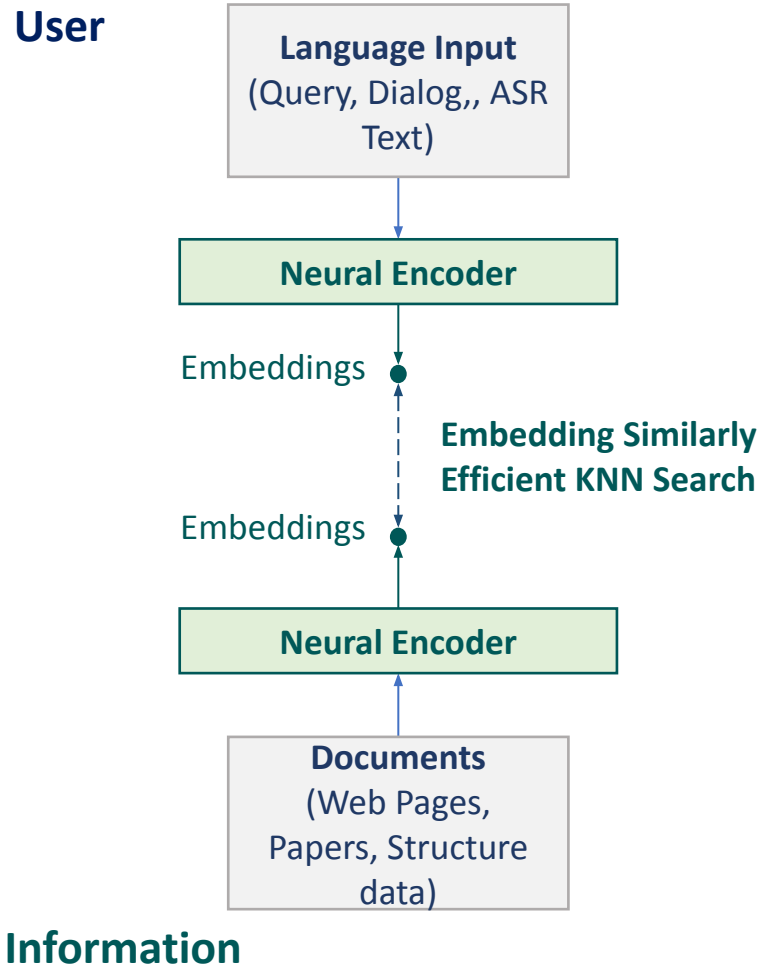
Overview of Modern Information Retrieval Systems

- An example search component being updated by LLMs
- Glances of other components using LLMs

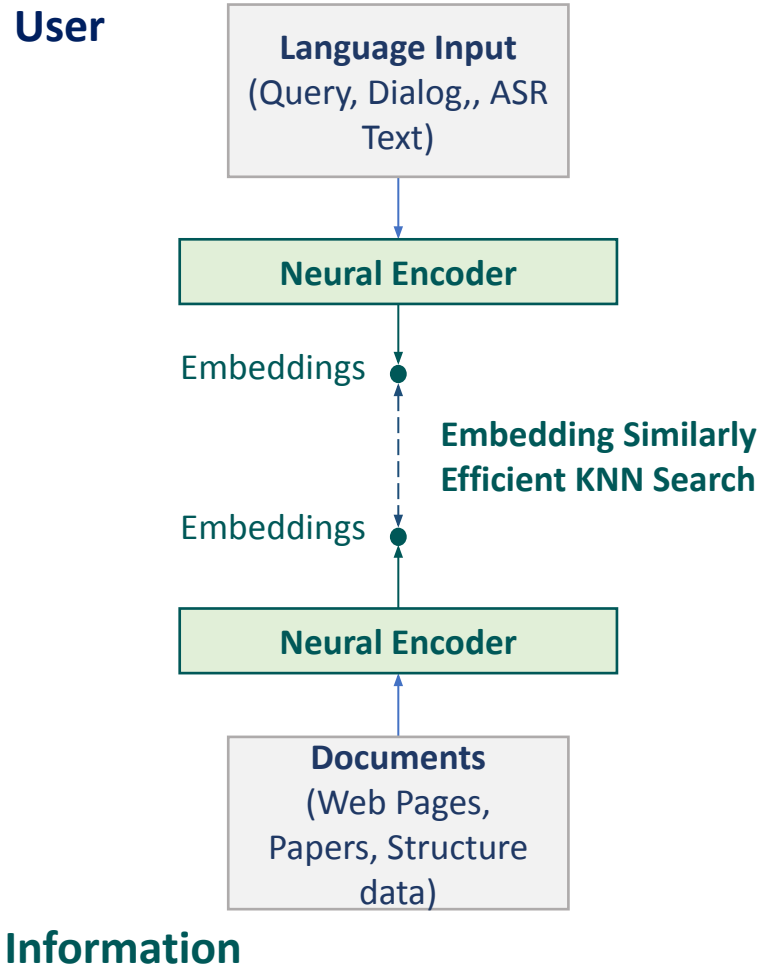
Dense Retrieval, a revolution of search with LLMs

- End-to-end learned retrieval
- Notable extensions
- Pretraining retrieval representations

Dense Retrieval: Matching with Fully Learned Embeddings



Dense Retrieval: Matching with Fully Learned Embeddings



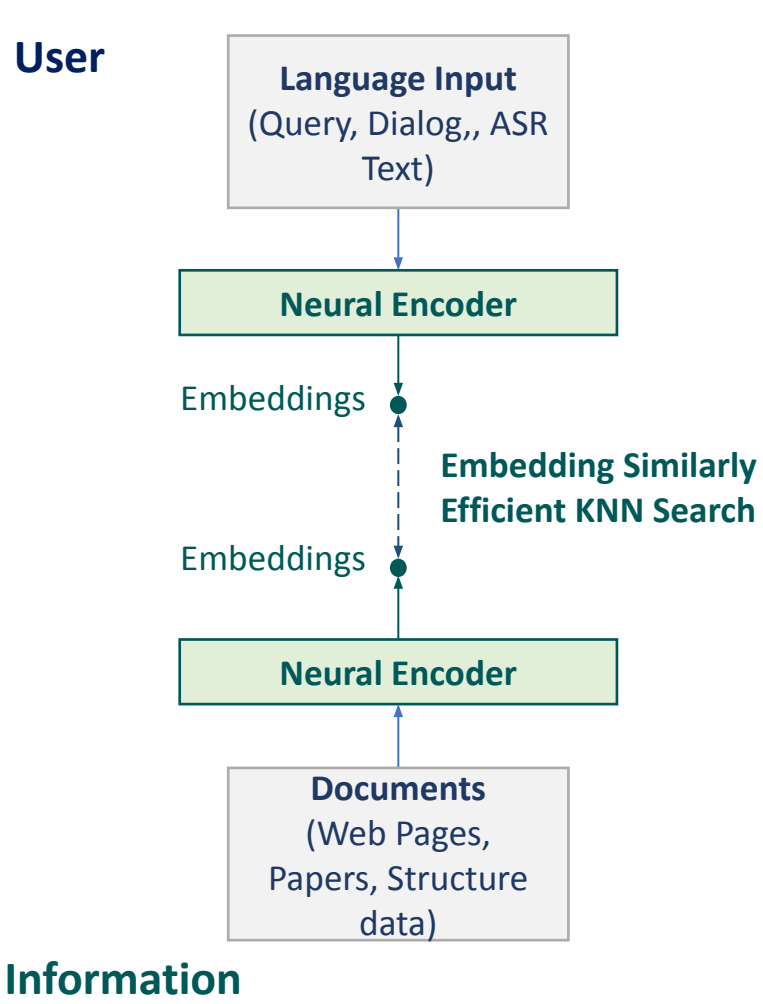
Matching with learned semantic representations instead of bag-of-words

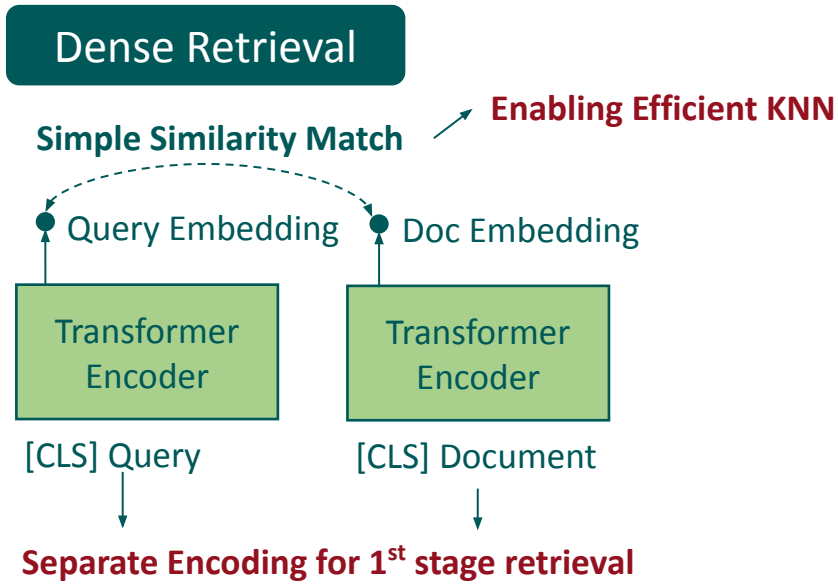
A pipe dream of IR, with lots of attempts for half a century

- Controlled vocabularies
- Ontologies
- Latent Semantic Index
- Topic Models
- Knowledge Graphs
- Shallow Neural Networks

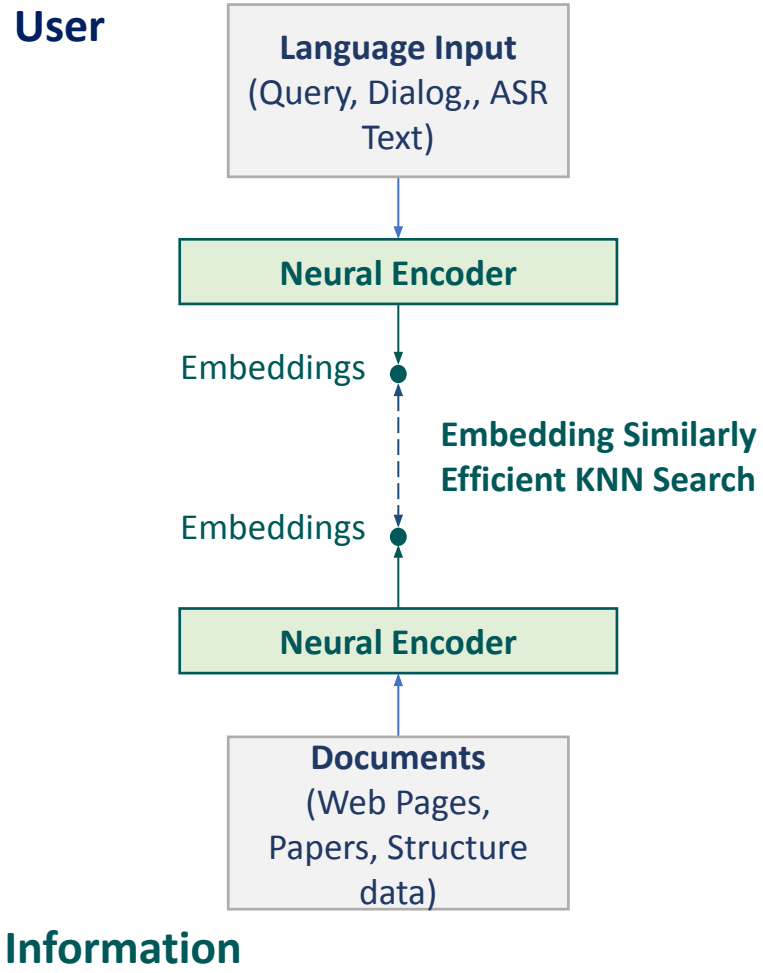
None achieved general success

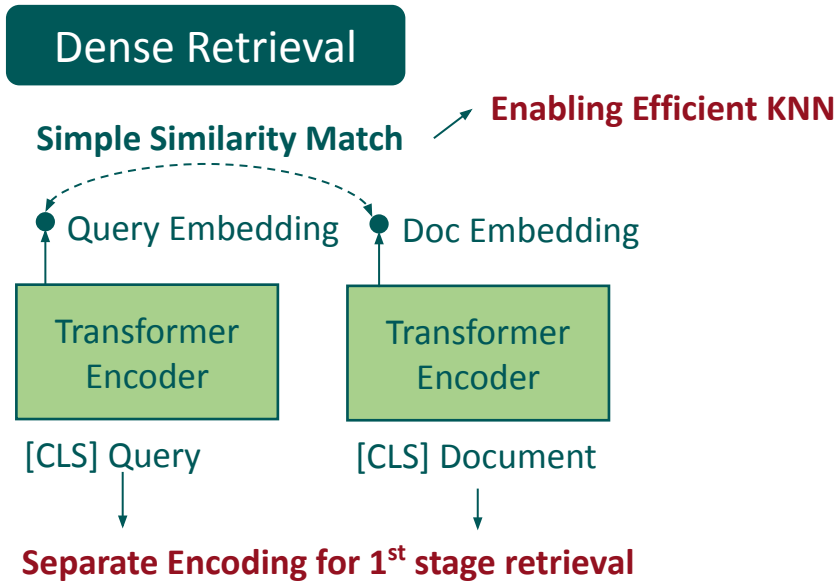
Dense Retrieval: Matching with Fully Learned Embeddings





Dense Retrieval: Matching with Fully Learned Embeddings





A representation-centric approach:

- All system capacity from encoders, only simple vector operations afterwards

Dense Retrieval: Formulation

A standard setup with BERT Encoders [4]

Retrieval Function: (Dual Encoder)

$$f(q,d) = \text{BERT}(q) \cdot \text{BERT}(d) = \text{MLP}\left(\overrightarrow{[\text{CLS}]}_q\right) \cdot \text{MLP}\left(\overrightarrow{[\text{CLS}]}_d\right)$$

Inference: (Approximate KNN Search)

$D_q = \text{ANN}_{f(q,\circ)}$ Finding K nearest neighbor in the corpus with approximate nearest neighbor search.

Dense Retrieval: Formulation

A standard setup with BERT Encoders [4]

Retrieval Function: (Dual Encoder)

$$f(q,d) = \text{BERT}(q) \cdot \text{BERT}(d) = \text{MLP}\left(\overrightarrow{[\text{CLS}]}_q\right) \cdot \text{MLP}\left(\overrightarrow{[\text{CLS}]}_d\right)$$

Inference: (Approximate KNN Search)

$$D_q = \text{ANN}_{f(q,\circ)}$$

Finding K nearest neighbor in the corpus with approximate nearest neighbor search.

Approximate nearest neighbor search (ANNS): Gain (sub-linear) efficiency by slightly scarifying KNN accuracy

- Partition-based methods: Split the space into regions and only search sub regions
  - E.g., hierarchical K-means trees
- Hash-based methods: Map data points by hashing functions and only search certain hash codes
  - E.g., Locality sensitive hash
- Graph-based methods: Connect data points by similarity edges and greedily traverse the graph
  - E.g., K-nearest neighborhood graph

Can achieve similar cost/efficiency as inverted index (not yet in standard open-source toolkits)

Dense Retrieval: Training

Representation learning using standard query-relevant document pairs

Learning: (Contrastive Learning / Learning to Rank)

$$\theta^* = \text{argmin}_\theta \sum_{q} \sum_{d^+} \sum_{d^- \sim P_{D^-}} l(f(q, d^+), f(q, d^-))$$

Relevant q-d pairs (given)

Negative Sampling

Standard Ranking Loss

Dense Retrieval: Challenge

Standard random negatives too weak for retrieval

Learning: (Contrastive Learning / Learning to Rank)

$$\theta^* = \text{argmin}_\theta \sum_{q} \sum_{d^+} \sum_{d^- \sim P_{D^-}} l(f(q, d^+), f(q, d^-))$$

Relevant q-d pairs (given)

Negative Sampling

Standard Ranking Loss

Figure 4: Dense Retrieval Training Loss with Randomly Sampled Negatives on MSMARCO

Dense Retrieval: Challenge

Standard random negatives too weak for retrieval

Learning: (Contrastive Learning / Learning to Rank)

$$\theta^* = \text{argmin}_\theta \sum_{q} \sum_{d^+} \sum_{d^- \sim P_{D^-}} l(f(q, d^+), f(q, d^-))$$

Relevant q-d pairs (given)

Negative Sampling

Standard Ranking Loss

Figure 4: Dense Retrieval Training Loss with Randomly Sampled Negatives on MSMARCO

A severe problem because of unique properties of retrieval

- Corpus size is **huge**: millions, billions, or trillions
- 99.99% are trivially irrelevant
- Retrieval is to distinguish a **small number** of hard negatives

Dense Retrieval: Training with Sparse Retrieval Negatives

Sampling negatives from top results of existing sparse retrieval systems

$$\theta^* = \text{argmin}_\theta \sum_q \sum_{d^+} \sum_{d^- \sim P_{D^-}} l(f(q, d^+), f(q, d^-))$$

- Negatives from existing inverted index (industry's sparse retrieval). (Bing Vector Search) [Waldburger 2019]
- Sampling from BM25 Top K. (DPR) [Karpukhin et al. 2020]
- Offline hard negative mining from production system (Facebook Embedding Search) [Huang et al. 2020]

Dense Retrieval: Training with Sparse Retrieval Negatives

Sampling negatives from top results of existing sparse retrieval systems

$$\theta^* = \text{argmin}_\theta \sum_q \sum_{d^+} \sum_{d^- \sim P_{D^-}} l(f(q, d^+), f(q, d^-))$$

- Negatives from existing inverted index (industry's sparse retrieval). (Bing Vector Search) [Waldburger 2019]
- Sampling from BM25 Top K. (DPR) [Karpukhin et al. 2020]
- Offline hard negative mining from production system (Facebook Embedding Search) [Huang et al. 2020]

Pros:

- Bootstrap upon an existing system with meaningful negatives

Cons:

- Often negatives from sparse retrieval are still too trivial
- Weaker generalization ability

Dense Retrieval: Training with Self Negatives

Sampling negatives globally from the entire corpus using the dense retriever itself (ANCE [5]).

$$\theta^* = \text{argmin}_\theta \sum_q \sum_{d^+} \sum_{d^- \sim \text{ANN}_{f(q,\circ)}} l(f(q,d^+), f(q,d^-))$$

- Sampling from the top retrieved results of the dense retrieval model
- Periodically refresh the dense retrieval index to keep negatives updated
- Start from sparse retrieval negatives to warm up

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

Dense Retrieval: Training with Self Negatives

Sampling negatives globally from the entire corpus using the dense retriever itself (ANCE [5]).

$$\theta^* = \text{argmin}_\theta \sum_q \sum_{d^+} \sum_{d^- \sim \text{ANN}_{f(q,\circ)}} l(f(q, d^+), f(q, d^-))$$

- Sampling from the top retrieved results of the dense retrieval model
- Periodically refresh the dense retrieval index to keep negatives updated
- Start from sparse retrieval negatives to warm up

Pros:

- Aligned training and testing distribution
- Strong performance in-domain and out-of-domain

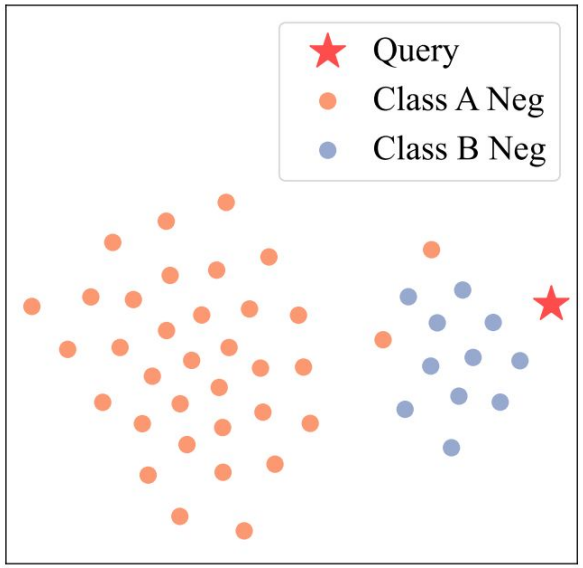
Cons:

- Overhead cost in refreshing the corpus index for negative sampling
- Instabilities from negative refreshes

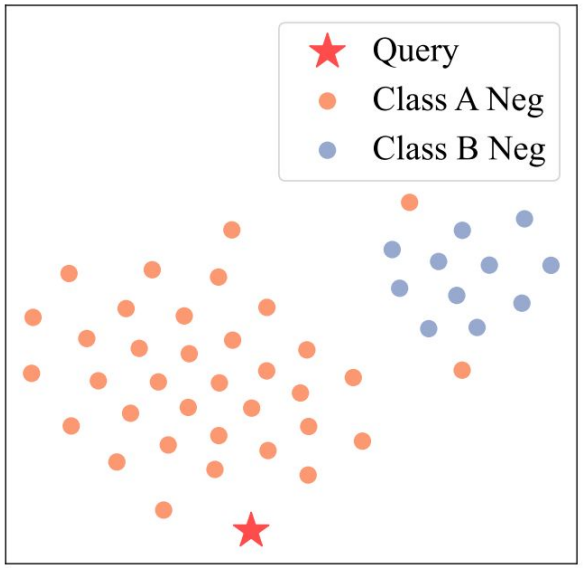
Dense Retrieval: Instabilities from Negative Sampling

Dense retriever swings between several groups of negatives [6]

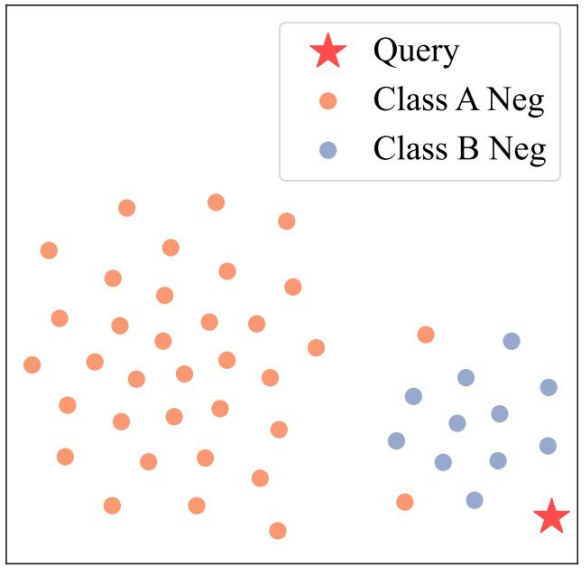
Query	Class A Negatives	Class B Negatives
most popular breed of rabbit	The Golden Retriever is one of the most **popular breeds** in the United States…	**Rabbit** habitats include meadows, woods, forests, grasslands, deserts and wetlands…



After Negative Refresh #1



After Negative Refresh #2



After Negative Refresh #3

Figure 5: T-SNE plots of a query and its two negative groups during ANCE training [6]

Dense Retrieval: Training with Teleportation Negatives

Smooth training by combining negatives from past samples and potential future samples (ANCE-Tele [7])

$$\theta^* = \text{argmin}_\theta \sum_q \sum_{d^+} \sum_{d^- \sim \text{Tele}_i} l(f(q, d^+), f(q, d^-))$$

$$\text{Tele}_i = \text{ANN}_{f_i(q,\circ)} + \text{Tele}_{i-1} + \text{ANN}_{f_i(d^+,\circ)}$$

Self-Negatives from current (i-th) training episode

Negatives from previous episode (Momentum)

Approximation of future negatives using neighbors of d^+ (Lookahead)

Dense Retrieval: Training with Teleportation Negatives

Smooth training by combining negatives from past samples and potential future samples (ANCE-Tele [7])

$$\theta^* = \text{argmin}_\theta \sum_q \sum_{d^+} \sum_{d^- \sim \text{Tele}_i} l(f(q, d^+), f(q, d^-))$$

$$\text{Tele}_i = \text{ANN}_{f_i(q,\circ)} + \text{Tele}_{i-1} + \text{ANN}_{f_i(d^+,\circ)}$$

Self-Negatives from current (i-th) training episode

Negatives from previous episode (Momentum)

Approximation of future negatives using neighbors of d^+ (Lookahead)

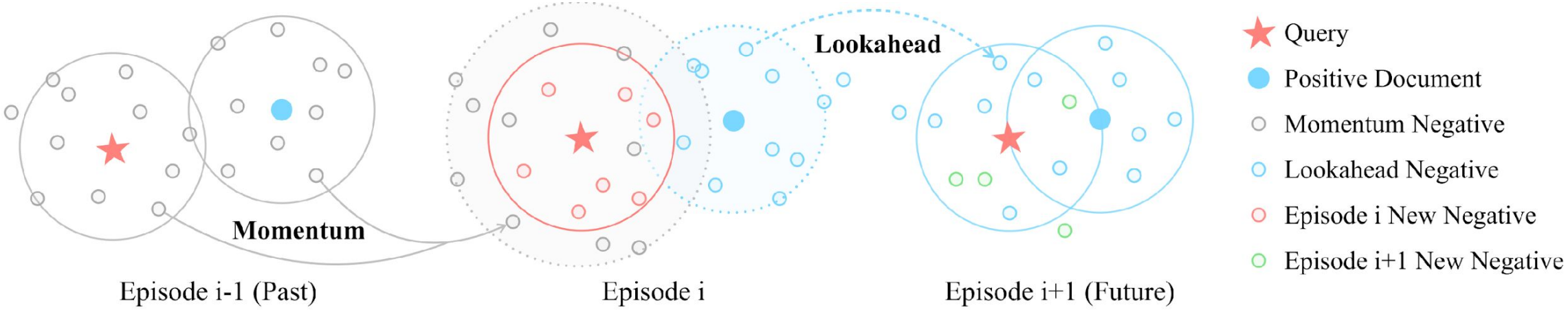


Figure 6: Smooth Negative Sampling with Momentum and Lookahead [6]

Dense Retrieval: Performances

Evaluation on supervised retrieval: MS MARCO Passage Task.

- Retrieve answer passages for Bing questions from a corpus of ~10M passages
- All dense retrievers start from RoBERTa base.

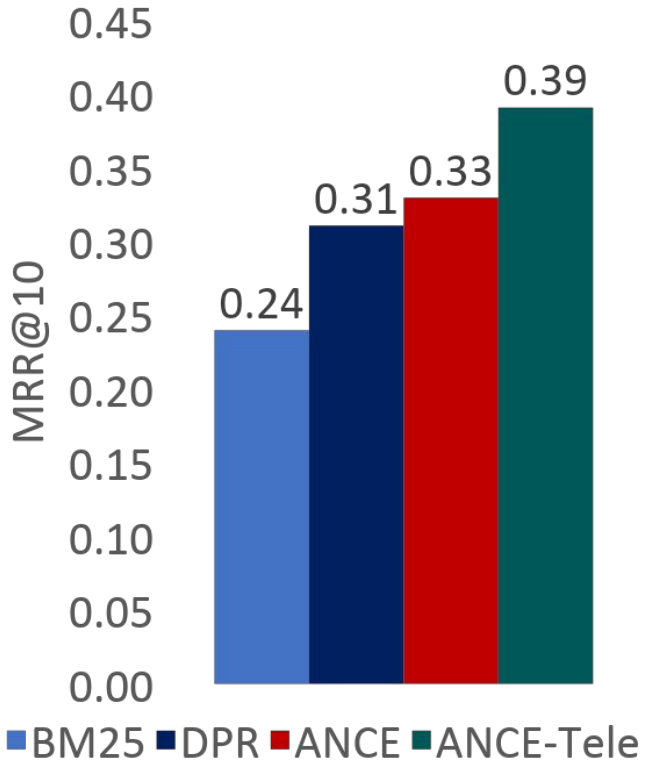


Figure 7: Supervised Retrieval Performances on MS MARCO.

BM25: Standard sparse bag-of-words based retrieval

DPR: Trained with BM25 negatives + random negatives

ANCE: Trained with self-negatives (warmed up by BM25 negative)

ANCE-Tele: Trained with momentum and lookahead global negatives

Dense Retrieval: Examples

Retriever	Query	Bad Case	Relevant Document
BM25	What is the most popular food in Switzerland	Answers.com: Most popular traditional food dishes of Mexico	Wikipedia: Swiss cuisine
ANCE	How long to hold bow in yoga	Yahoo Answer: How long should you hold a yoga pose for	yogaoutlet.com: How to do bow pose in yoga

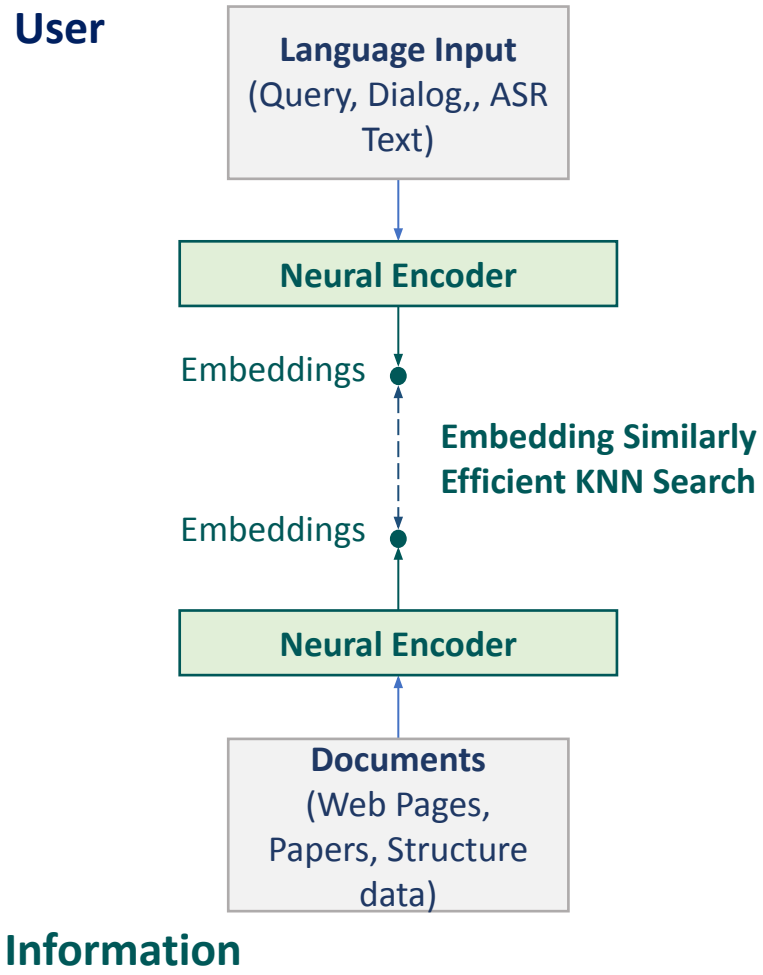
Table 1: Error Cases of BM25 and ANCE in TREC Deep Learning Track Document Retrieval 2019 [5]

Sparse retrieval and dense retrieval behave quite differently.

- BM25 and ANCE only agree on 20% of their top 100 rankings
- But both find relevant document in top 3

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

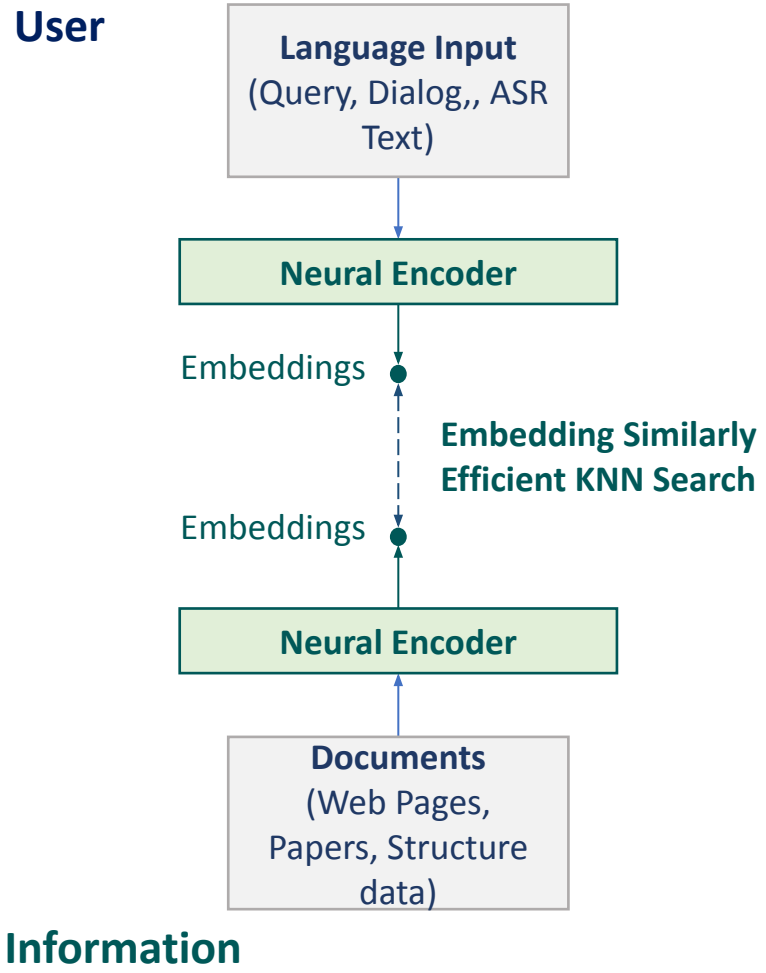
Dense Retrieval: Summary



A long-desired goal, finally achieved because of two advancements:

1. Representation power of LLMs (Major)
2. Retrieval-oriented fine-tuning (Last Mile)

Dense Retrieval: Summary



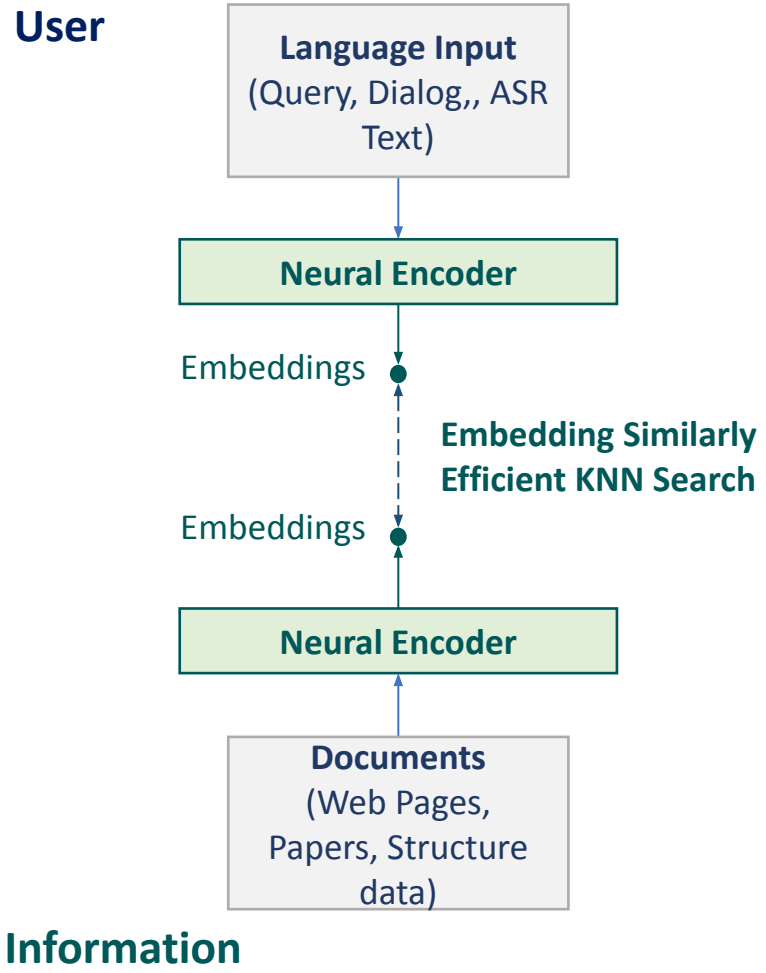
A long-desired goal, finally achieved because of two advancements:

1. Representation power of LLMs (Major)
2. Retrieval-oriented fine-tuning (Last Mile)

A "Painkiller" solution

- Eliminate a major bottleneck of existing search solutions
- A fundamental solution of an intrinsic challenge in status quo

Dense Retrieval: Summary



A long-desired goal, finally achieved because of two advancements:

1. Representation power of LLMs (Major)
2. Retrieval-oriented fine-tuning (Last Mile)

A "Painkiller" solution

- Eliminate a major bottleneck of existing search solutions
- A fundamental solution of an intrinsic challenge in status quo

Enabled lots of potentials

- Democratize state-of-the-art search
- Ride the generalization power of LLMs
- Unify many modalities and scenarios in one embedding space
- Many vector-based search startups and heavy investments

Outline

Overview of Modern Information Retrieval Systems

- An example search component being updated by LLMs
- Glances of other components using LLMs

Dense Retrieval, a revolution of search with LLMs

- End-to-end learned retrieval
- **Notable extensions**
- Pretraining retrieval representations

Dense Retrieval Extensions: Stronger Foundation Models

Sentence T5 Encoder-Decoder: bringing in benefits of T5

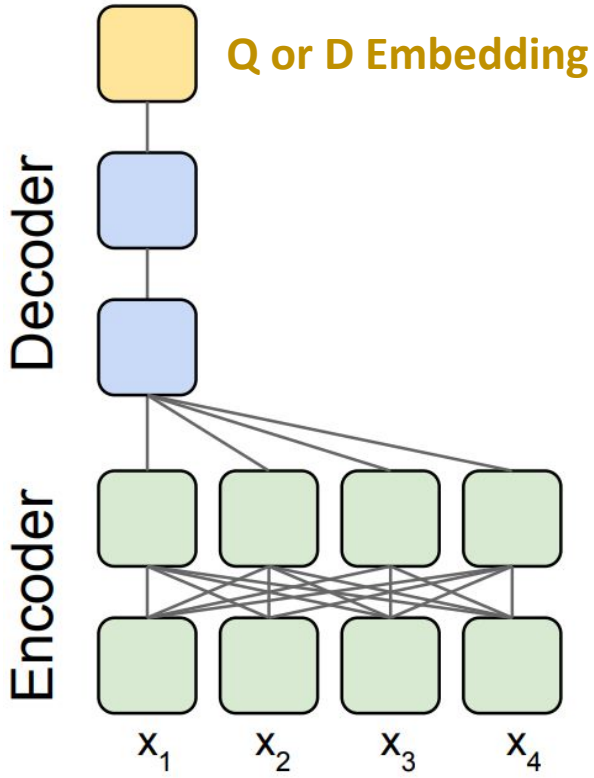


Figure 8: Architecture of SentenceT5 Encoder-Decoder [7].

Benefits:

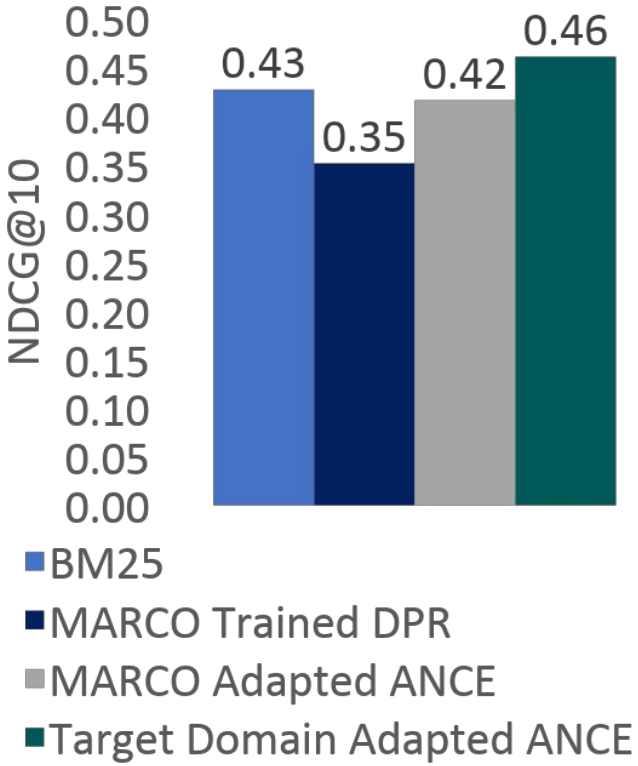
- Better pretrained model (T5 > RoBERTa)
- Easy to introduce prompts/instructions, especially task specific ones in multi-task setting
- Current go-to solution at T5's scales

[7] Ni et al. "Sentence-T5: Scalable Sentence Encoders from Pre-trained Text-to-Text Models". ACL 2022.

Dense Retrieval Extensions: Robust Zero-Shot

Various techniques to make web-trained dense retrievers generalizable to other search domains

- Lots of real-world needs (e.g., OpenAI embedding API, AWS Open Search, Azure Search)
- Most successful techniques are to continuously pretrain underlying LLM in target corpus



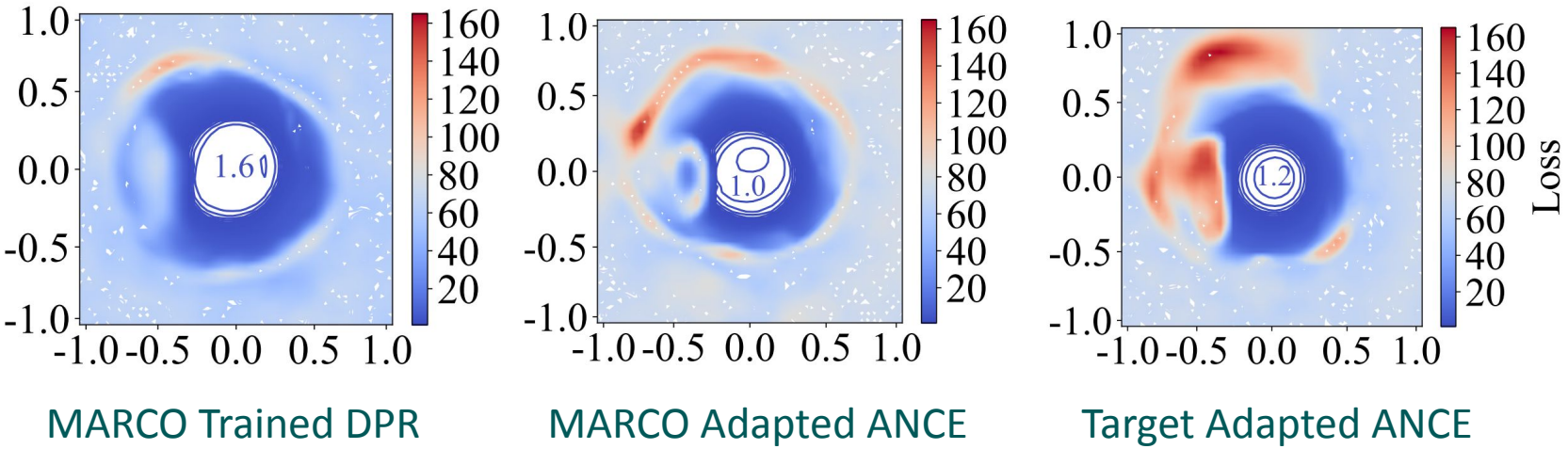
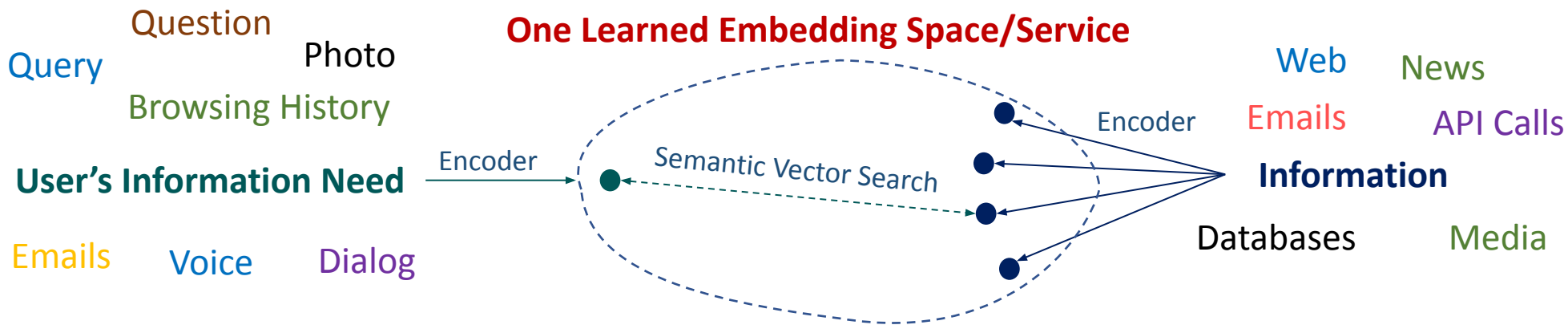


Figure 9: Loss landscape of dense retrieval models on MARCO development set.

Dense Retrieval Extensions: Universal Retrieval

Map queries and documents in variant formats and modalities into one central embedding space

- Enable cross scenario and cross modality information access
- One unified entry for search



Current Solution: Ride the universal representation capability of foundational models

- Linearize structured data with prompts, e.g., Table BERT, and use text model
- Leverage multi-modality foundational models, e.g., CLIP for image-text
- Continuous pretrain LM on other data formats, e.g., code, molecular SMILE

Outline

Overview of Modern Information Retrieval Systems

- An example search component being updated by LLMs
- Glances of other components using LLMs

Dense Retrieval, a revolution of search with LLMs

- End-to-end learned retrieval
- Notable extensions
- **Pretraining retrieval representations**

Mismatches Between LM Pretraining and Retrieval Needs

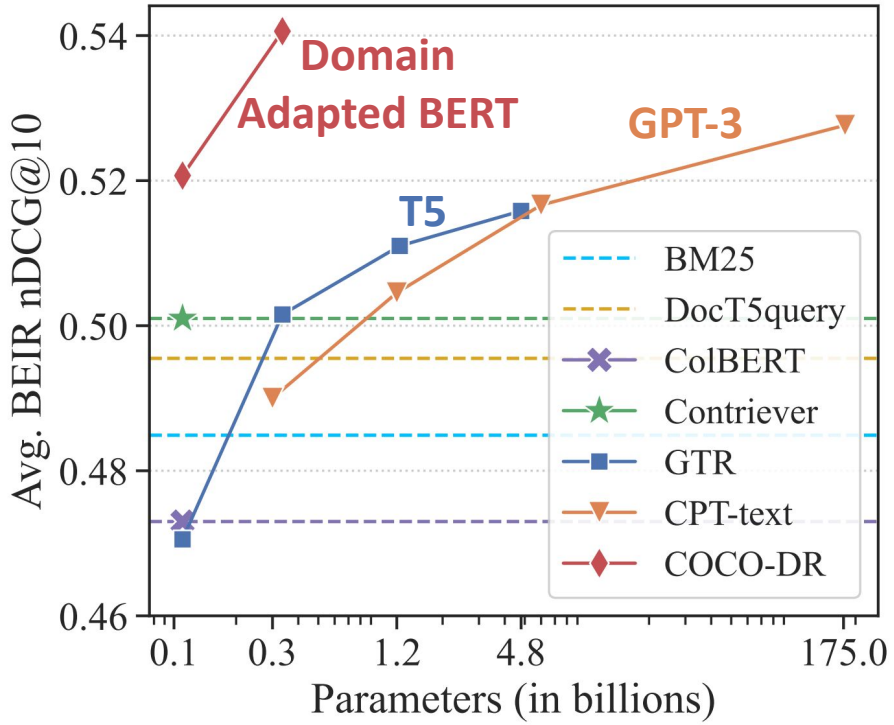
Various observations that pretrained LLMs are not as strong in retrieval than other language tasks

- Zero zero-shot performance from vanilla LMs, e.g., BERT, ELECTRA
- Required more complicated fine-tuning, e.g., ANCE
- Prompting LLMs not really working

Mismatches Between LM Pretraining and Retrieval Needs

Various observations that pretrained LLMs are not as strong in retrieval than other language tasks

- Zero zero-shot performance from vanilla LMs, e.g., BERT, ELECTRA
- Required more complicated fine-tuning, e.g., ANCE
- Prompting LLMs not really working



Much worse scaling law from LLMs in retrieval

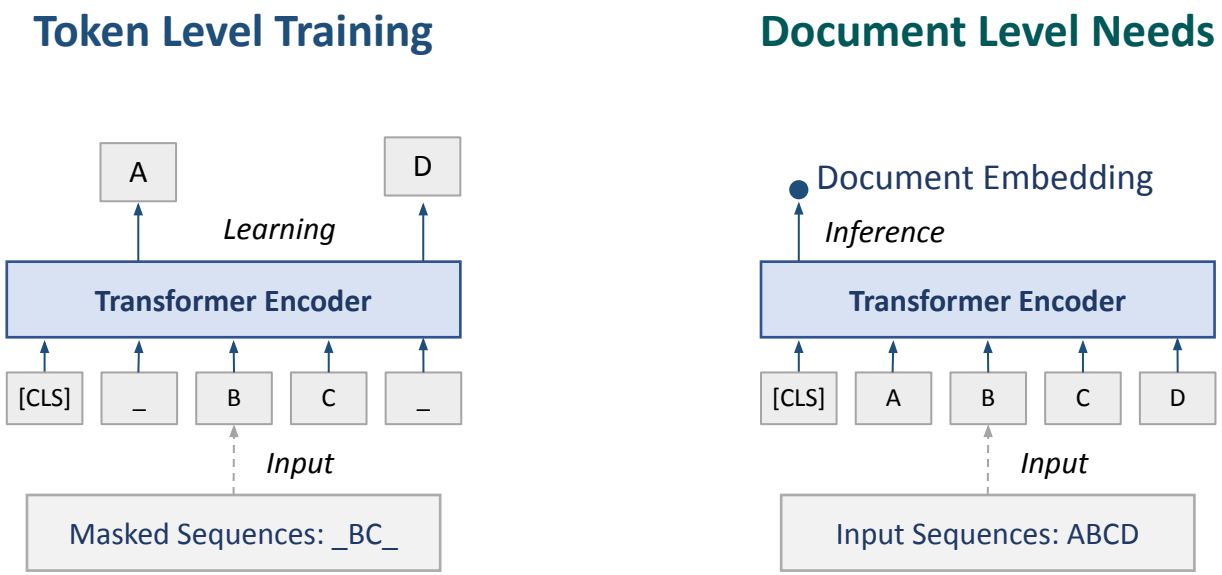
- GPT-3 much worse than T5 at similar scale
- More diminished return when scaling up
- Generalization heavily depends on domain adapted pretraining

Figure 10: Scaling of LLMs on Zero-Shot Dense Retrieval [8]

Mismatch #1: Local versus Global

Language modeling is more about local contexts

Retrieval requires capturing information of the full document

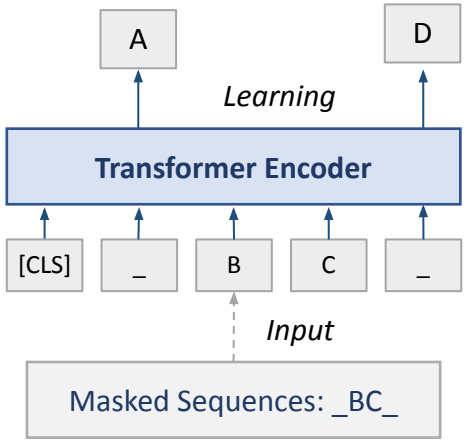


Mismatch #1: Local versus Global

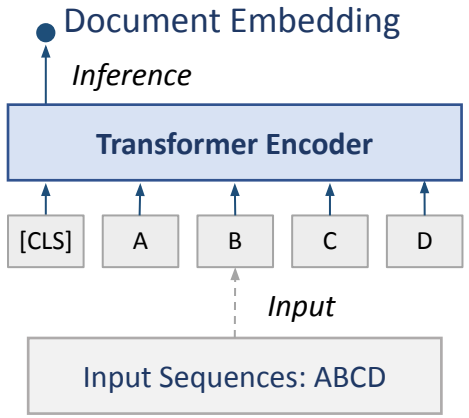
Language modeling is more about local contexts

Retrieval requires capturing information of the full document

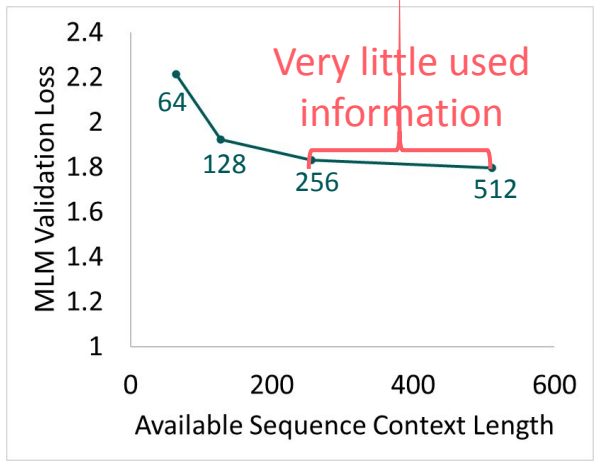
Token Level Training



Document Level Needs



BERT Base MLM Loss



Mismatch #1: Local versus Global

Language modeling is more about local contexts

- Long context methods mainly work on specific long range tasks (not retrieval)
- “the longer context model retains strong performance on various general-purpose tasks” (LLaMA 2 [8])

Context Length	Hella-Swag (0-shot)	NQ (64-shot)	TQA (64-shot)	GSM8K (8-shot)	Human-Eval (0-shot)
2k	75.1	25.5	53.7	4.9	7.9
4k	74.8	25.5	52.2	6.5	7.3

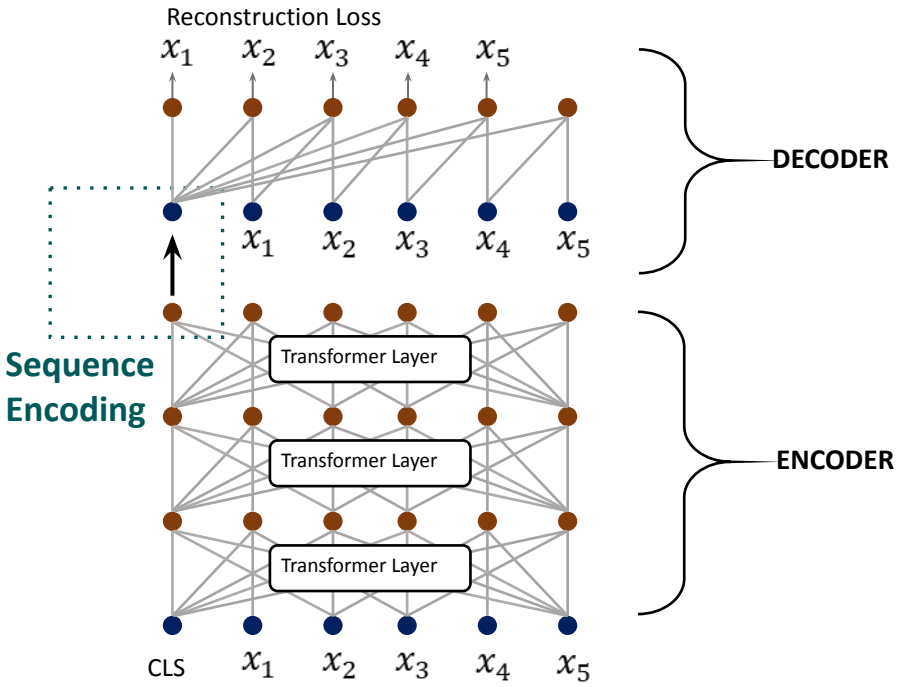
Table 2: LLaMA 2 performance on general-purpose tasks with different pretraining context length [8]

[8] Touvron et al. “Llama 2: Open Foundation and Fine-Tuned Chat Models”. ArXiv 2023.

Mismatch #1 Solution: Auto-Encoder Training

Reintroduce self-reconstruction loss on sequence embeddings to capture full sequence information [9]

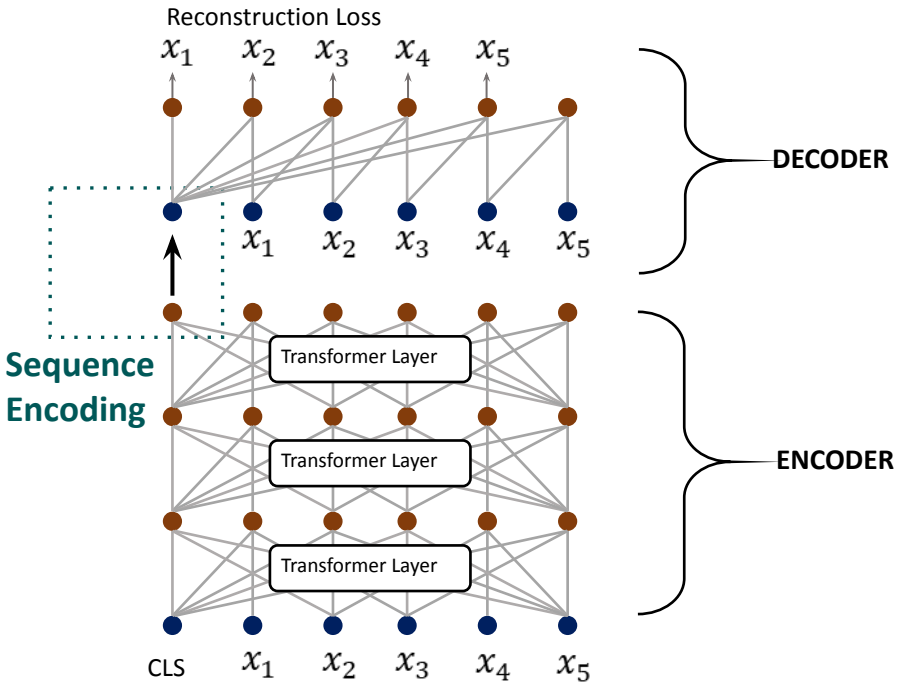
Information bottleneck on Document Encoding



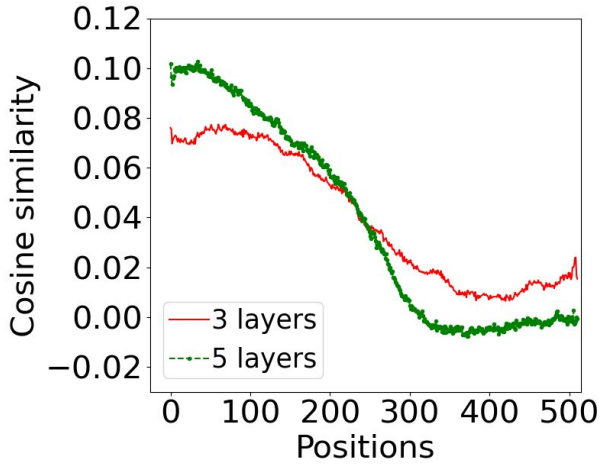
Mismatch #1 Solution: Auto-Encoder Training

Reintroduce self-reconstruction loss on sequence embeddings to capture full sequence information [9]

Information bottleneck on Document Encoding

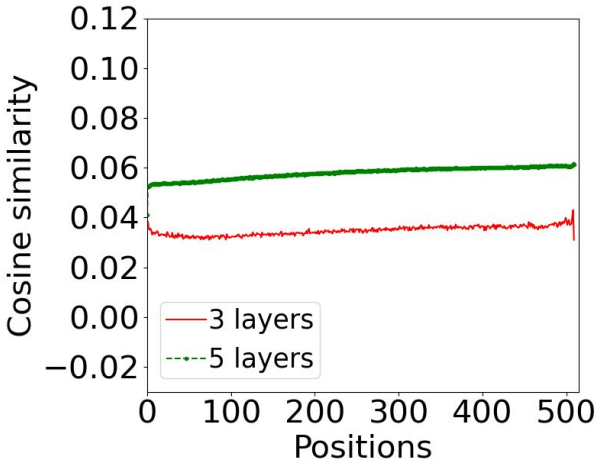


Restrict decoder capacity to enforce dependency on encoder



Without restricting decoder:

- Encoding only cares first half tokens
- Later decoder has sufficient contexts & does not care about encoding quality



With restricted decoder:

- Encoding captures latter tokens too
- Better starting point for dense retrieval
- Better few-shot ability too

- **Many similar ways to configure this, e.g., see condenser, TSDAE, etc.**
- **T5 implicitly has this global information from its pretraining setup and has benefit of large scale pretraining**

[9] Lu et al. "Less is More: Pre-train a Strong Text Encoder for Dense Retrieval Using a Weak Decoder". EMNLP 2021.

Mismatch #2: Anisotropy/Non-Uniformity

The sequence embedding space of many pretrained LLMs are highly non-uniform

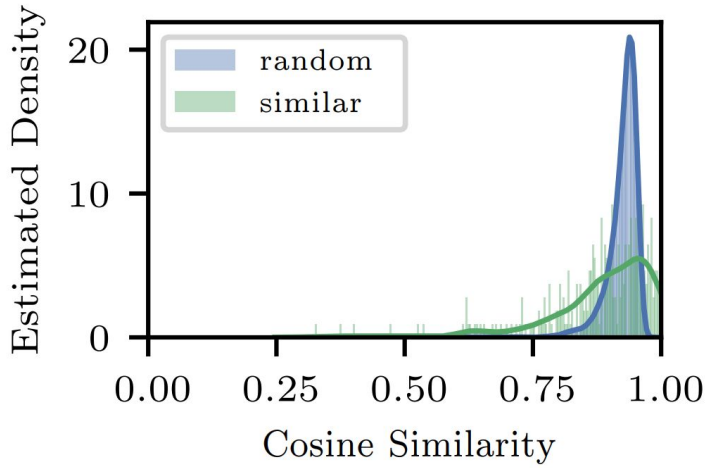


Figure 11: Similarity of RoBERTa $\overrightarrow{[\mathrm{CLS}]}$ on semantically similar and random pairs from STS-S [10]

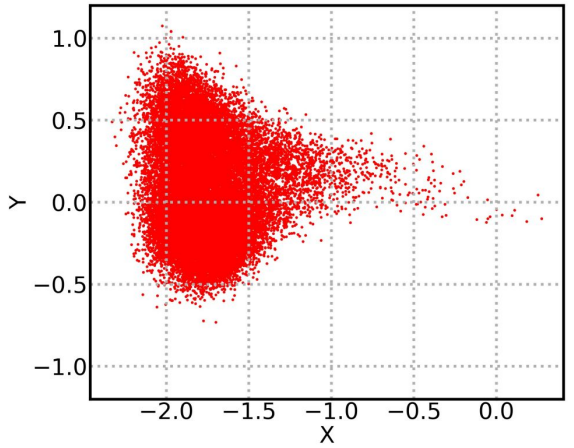


Figure 12: SVD 2-D mapping of word embeddings from Transformer trained on EN→DE [11]

Mismatch #2: Anisotropy/Non-Uniformity

The sequence embedding space of many pretrained LLMs are highly non-uniform

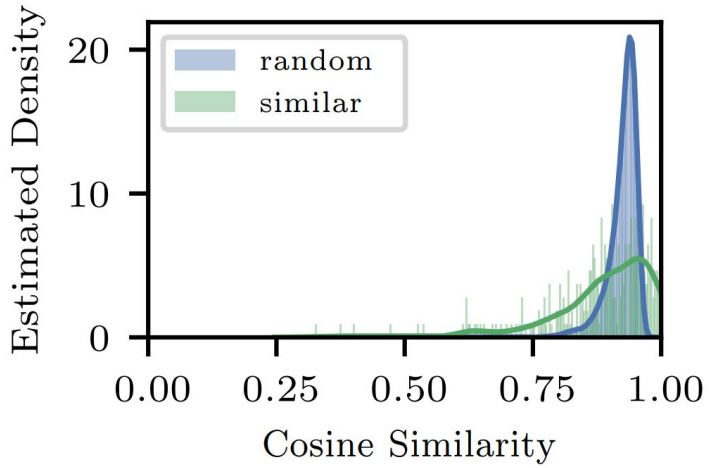


Figure 11: Similarity of RoBERTa $\overrightarrow{[\mathrm{CLS}]}$ on semantically similar and random pairs from STS-S [10]

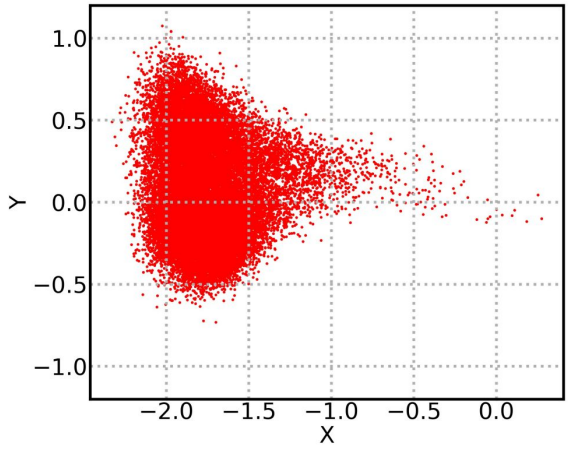


Figure 12: SVD 2-D mapping of word embeddings from Transformer trained on EN→DE [11]

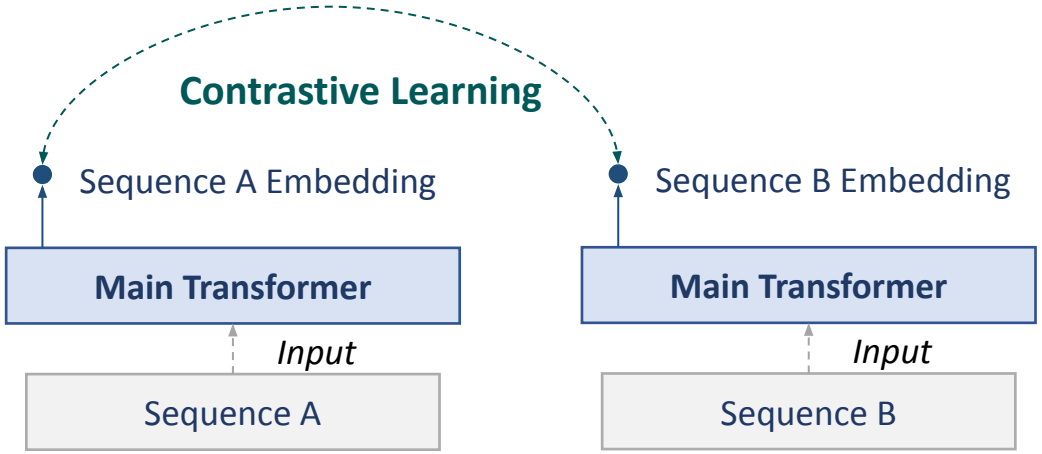
Most rare tokens are pushed to a narrow cone in the space, and [CLS] is a rare token in learning

- Every training signal pushes all negatives away from the positive
- Rare tokens (without much or any positive pulls) are pushed away from all positives, into a narrow cone
- Theoretical intuitions in Gao et al. [11]

Mismatch #2 Solution: Sequence Contrastive Learning

- Pretraining sequence representations with Sequence Contrastive Learning (SCL)

Adding pretraining task: $L_{\text{SCL}} = \text{E}\left(\frac{\exp(\cos(\boldsymbol{s},\boldsymbol{s}^{+}))}{\exp(\cos(\boldsymbol{s},\boldsymbol{s}^{+}))+\sum_{s^{-}}\exp(\cos(\boldsymbol{s},\boldsymbol{s}^{-}))}\right)$



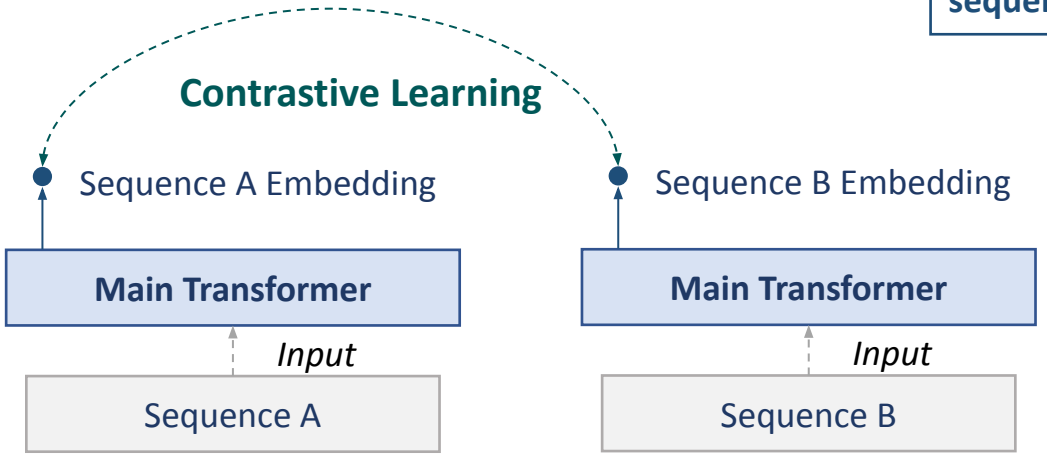
Mismatch #2 Solution: Sequence Contrastive Learning

- Pretraining sequence representations with Sequence Contrastive Learning (SCL)

Adding pretraining task: $L_{\text{SCL}} = \text{E}\left(\frac{\exp(\cos(\boldsymbol{s},\boldsymbol{s}^{+}))}{\exp(\cos(\boldsymbol{s},\boldsymbol{s}^{+}))+\sum_{s^-}\exp(\cos(\boldsymbol{s},\boldsymbol{s}^{-}))}\right)$

Embeddings of positive contrast sequence pairs

Embeddings of negative sequence pairs



Construction of positive contrast sequence pairs:

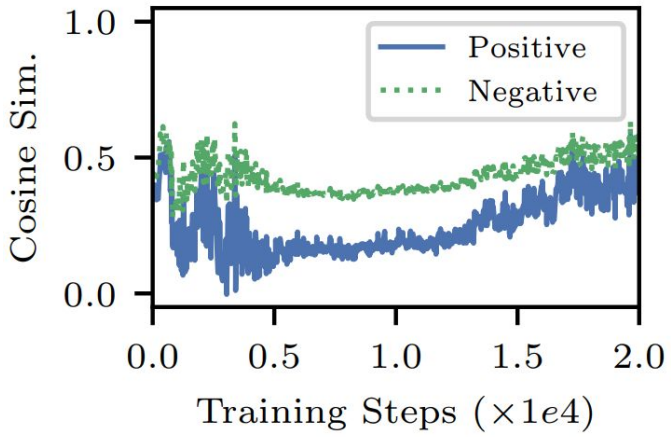
- *Data augmentation*: cropping [10], random replacement, back translation, different dropout (SimCSE), etc.
- *Unsupervised pairs*: next sentence prediction, etc.
- *Supervisions*: Web QA pairs, search query-clicked docs…

[10] Meng et al. "COCO-LM: Correcting and Contrasting Text Sequences for Language Model Pretraining". NeurIPS 2021.

Mismatch #2 Solution: Sequence Contrastive Learning

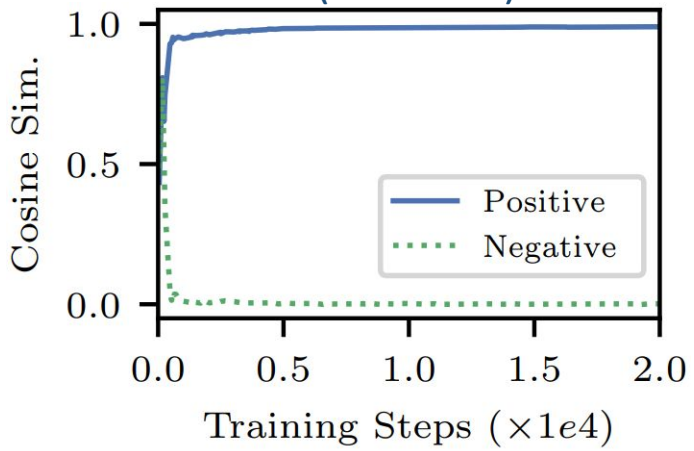
Recalibration of the embedding space, e.g., using cropped sequence pairs (90% overlap)

Training Pair Similarity
(Without SCL)



Failed without SCL
(Although 90% overlap!)

Training Pair Similarity
(With SCL)

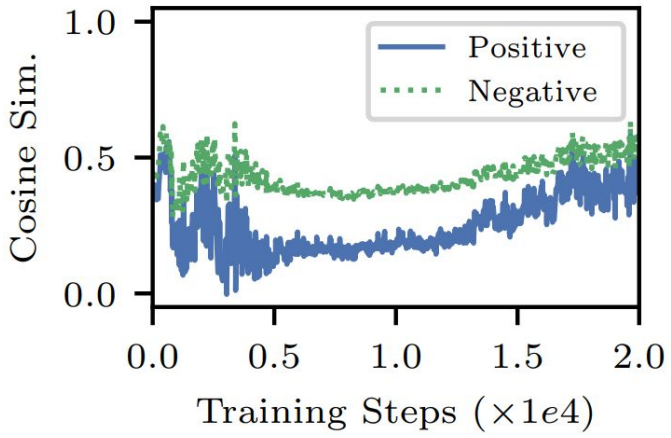


Easy-to-Learn Task
(90% overlap, after all)

Mismatch #2 Solution: Sequence Contrastive Learning

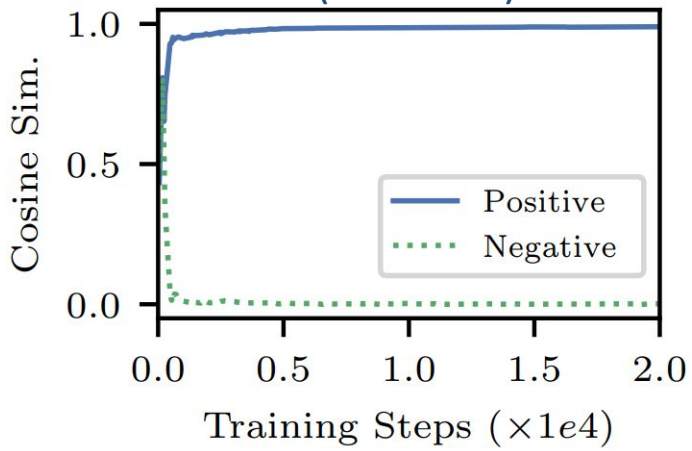
Recalibration of the embedding space, e.g., using cropped sequence pairs (90% overlap)

Training Pair Similarity
(Without SCL)



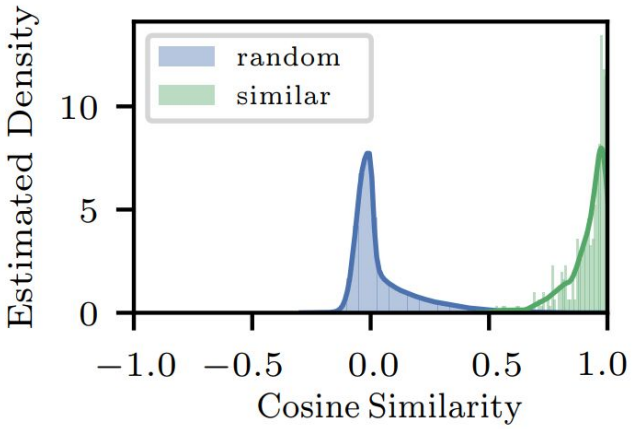
Failed without SCL
(Although 90% overlap!)

Training Pair Similarity
(With SCL)



Easy-to-Learn Task
(90% overlap, after all)

STS-B Similarity
(With SCL)



Effective Calibration
& Good Zero-Shot Ability

Decent zero-shot performance on many sequence similarity tasks and non-random performance on retrieval

Deeper Look into Contrastive Learning

Two forces in contrastive learning: Alignment and Uniformity [12]

$$L_{\text{SCL}} = \text{E}\left(\frac{\exp(\cos(\boldsymbol{s},\boldsymbol{s}^+))}{\exp(\cos(\boldsymbol{s},\boldsymbol{s}^+))+\sum_{s^-}\exp(\cos(\boldsymbol{s},\boldsymbol{s}^-))}\right)$$

$$\sim \underbrace{\cos(\boldsymbol{s},\boldsymbol{s}^+)}_{\text{Align}} + \underbrace{\log(\exp(\cos(\boldsymbol{s},\boldsymbol{s}^+)) + \sum_{s^-}\exp(\cos(\boldsymbol{s},\boldsymbol{s}^-)))}_{\text{Uniformly}}$$

Align positive pairs together

Uniformly spread random pairs in the space

- Proof in Wang et al. [12] that, if exist, perfectly aligned/uniform encoders minimize the two terms
- Note: here negatives are sampled uniformly, not from a long tail distribution

[12] Wang et al. "Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere". ICML 2020.

Deeper Look into Contrastive Learning

Two forces in contrastive learning: Alignment and Uniformity [12]

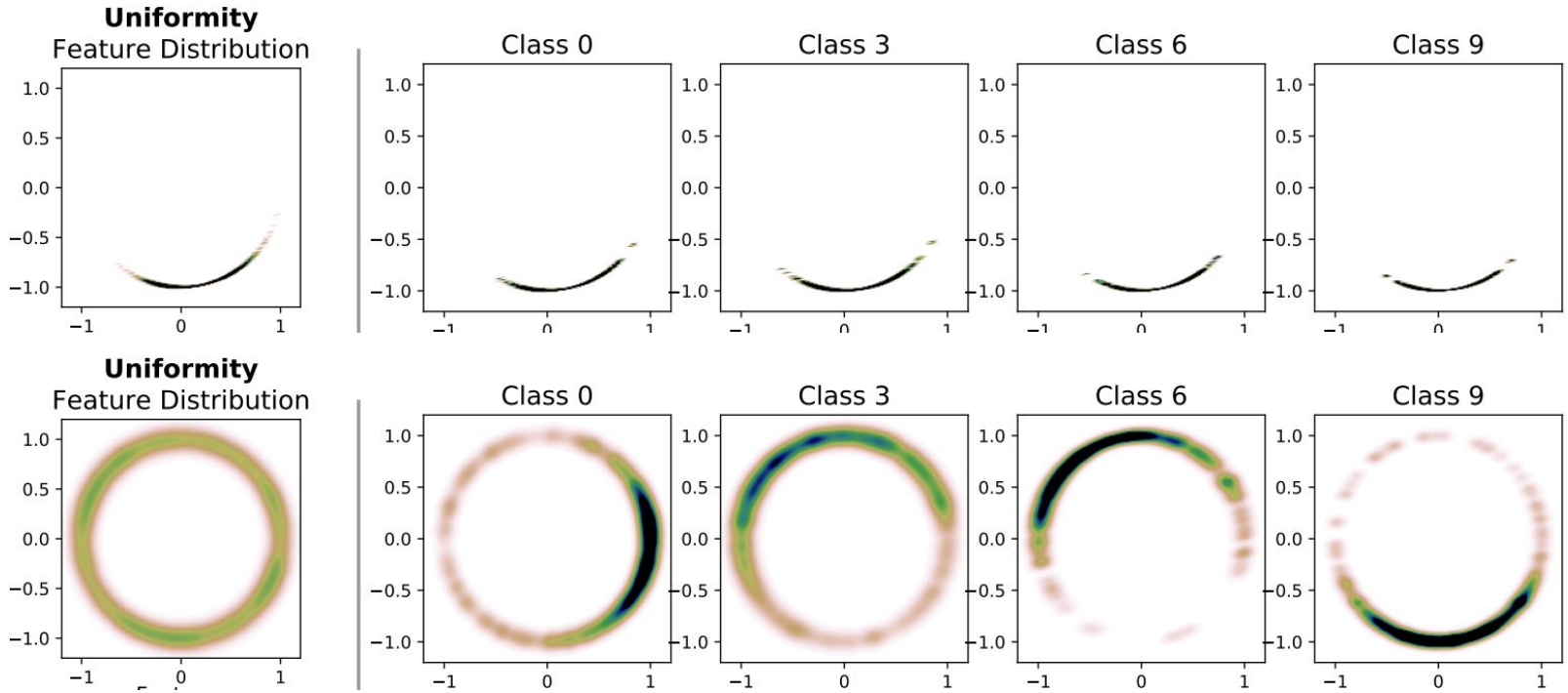


Figure 13: Uniformity of image features in CIFAR-10 from random network (top) and unsupervised contrastive learning (bottom) [12]

Mismatch #3: Alignments

What information does unsupervised contrastive pairs bring in to align the embedding space?

Method	Sequence A	Sequence B
SimCSE	The Steelers enjoy a large, widespread fanbase nicknamed Steeler Nation.	The Steelers enjoy a large, widespread fanbase nicknamed Steeler Nation.
Inverse Cloze Task (ICT)	The Steelers enjoy a large, widespread fanbase nicknamed Steeler Nation.	They currently play their home games at Acrisure Stadium on Pittsburgh's North Side in the North Shore neighborhood,
Cropping Augmentation	The Steelers enjoy a large, widespread fanbase nicknamed ___	____ enjoy a large, widespread fanbase nicknamed Steeler Nation.
Co-document	The Steelers enjoy a large, widespread fanbase nicknamed Steeler Nation.	In the NFL's "modern era" (since the AFL–NFL merger in 1970) the Steelers have posted the best record in the league.

Very limited semantic signals in the alignment

- Either trivial paraphrasing or loosely correlated
- Pretty far away from search relevance
- 10% worse than BM25 after unsupervised contrastive learning (might just approximate BM25)

Mismatch #3 Solution: Weak Supervision from Web Graph

Leverage Anchor Texts and the document they point to pseudo query-relevant document pairs

Method	Sequence A	Sequence B
Anchor Document	Vegetarian Society of Ireland	The Vegetarian Society of Ireland is a registered charity. Our aim is to increase awareness of vegetarianism in relation to health,
Actual Argument Retrieval Data	Becoming a vegetarian is an environmentally friendly thing to do.	Health general weight philosophy ethics You don't have to be vegetarian to be green. Many special environments have been created by

Mismatch #3 Solution: Weak Supervision from Web Graph

Leverage Anchor Texts and the document they point to pseudo query-relevant document pairs

Method	Sequence A	Sequence B
Anchor Document	Vegetarian Society of Ireland	The Vegetarian Society of Ireland is a registered charity. Our aim is to increase awareness of vegetarianism in relation to health,
Actual Argument Retrieval Data	Becoming a vegetarian is an environmentally friendly thing to do.	Health general weight philosophy ethics You don't have to be vegetarian to be green. Many special environments have been created by

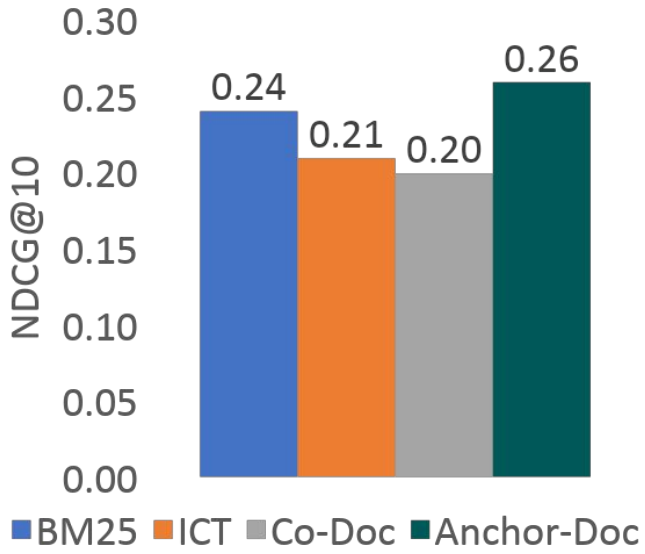


Figure 14: MARCO NDCG@10 of BM25 and dense retrievers trained by different unsupervised signals

Anchor-Doc the only unsupervised signal source outperforms BM25

- Other contrastive pairs not providing much semantics for relevance
- Data cleaning required to filter out functional anchors, e.g., "homepage"

A widely useful information in standard web search

- Page Rank, Document Expansion, etc.
- Still merely 10% better than vanilla BM25

Still a weakly supervised method, rather than a pretraining method

- Behavior closer to weak supervision/transfer learning, not pretraining
- Not seeing emergent capabilities

[13] Nie et al. "Unsupervised Dense Retrieval Training with Web Anchors". SIGIR 2023.

Mismatch Between LLM and Retrieval: Recap

Mismatch	Local vs. Global	Non-uniformity	Semantic Alignment
Solution?	Auto-Encoder	Contrastive Loss	Weak Supervision

We are still not seeing the emergent power of LLMs in embedding-based retrieval

- The fact we need these solutions/mitigations shows there is something missing

Auto-regressive LM + Scaling up captured a lot, but not everything

- Web search is perhaps the biggest money-making AI application, yet not fully covered by GPT-X

“Bitter lesson”, more compute and large scale trump specific designs, is deemed to happen

- But that may not solely via current language models

Outline

Overview of Modern Information Retrieval Systems

- An example search component being updated by LLMs
- Glances of other components using LLMs

Dense Retrieval, a revolution of search with LLMs

- End-to-end learned retrieval
- Notable extensions
- Pretraining retrieval representations

Quiz: What are the advantages of using T5 Encoder-Decoder instead of encoder only to produce document embeddings?

Dense Retrieval: Performances

Evaluation Tasks:

- Supervised Retrieval: MS MARCO Passage Task.
  - Retrieve answer passages for Bing questions from a corpus of ~10M passages
- Zero-Shot Retrieval: Transfer from MARCO to BEIR Benchmark.
  - A fused benchmark of 18 public tasks, with diverse domains and tasks

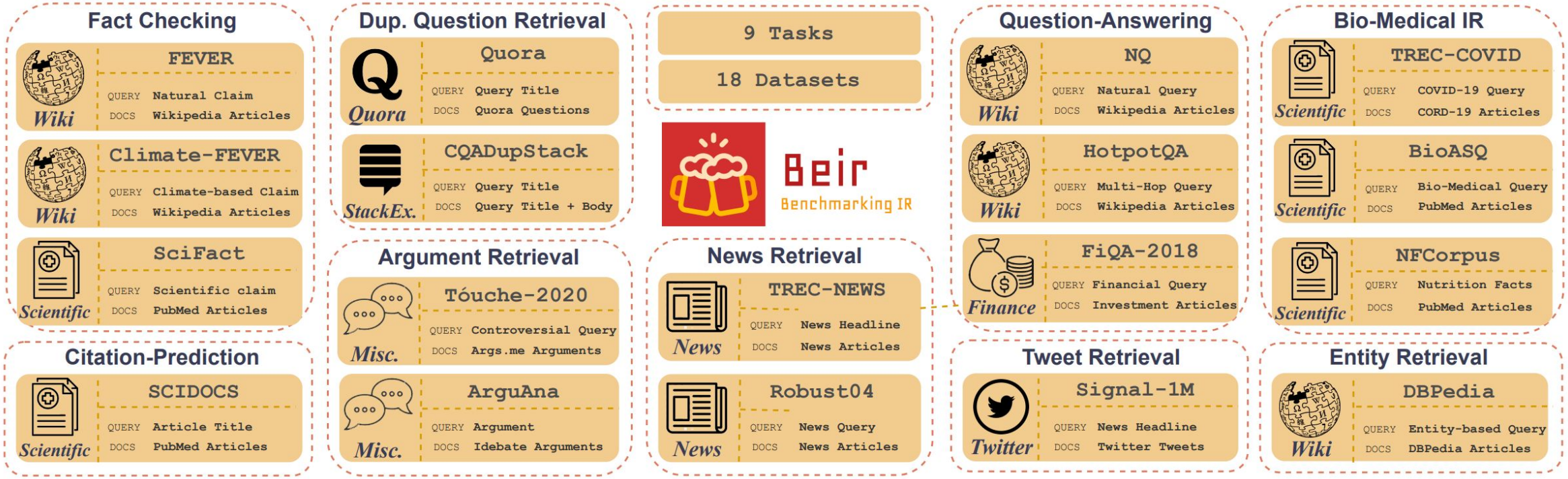


Figure 6: Tasks included in BEIR [6]