Data-Intensive Distributed Computing

CS 451/651 (Fall 2025)

Text Processing I

(v1.02)

Week 9: October 28, 2025

Jimmy Lin
David R. Cheriton School of Computer Science
University of Waterloo

These slides are available at http://lintool.github.io/bigdata-2025f/

Key Questions

How do data products with operational requirements complicate lakehouse architectures?

Why does retrieval remain important in the era of LLMs?

How is retrieval formulated as the problem of computing vector similarity?

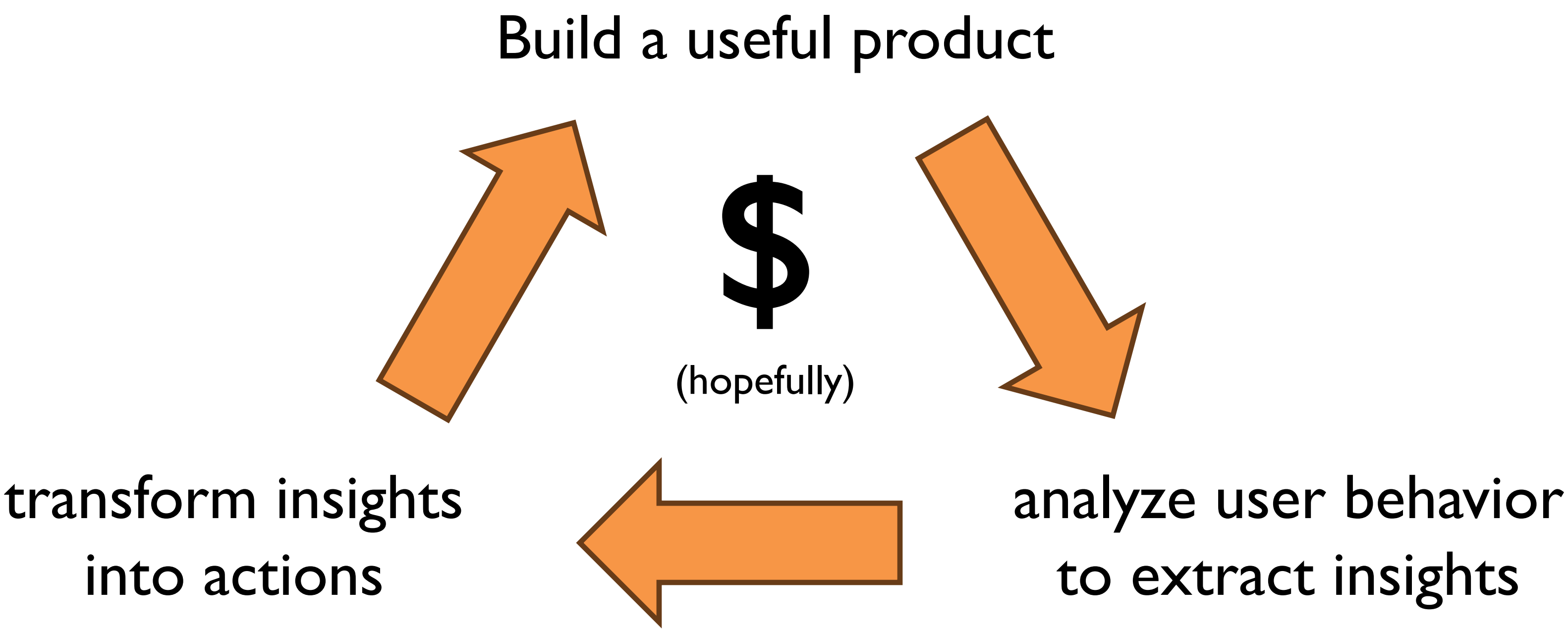
What’s the intuition behind sparse and dense representations?

Recap: What are we doing and why?

Context…

The Data Flywheel

(a virtuous cycle)



Google. Facebook. Twitter. Amazon. Uber.

Context…

What's this course about?

The *infrastructure* that supports the data flywheel.

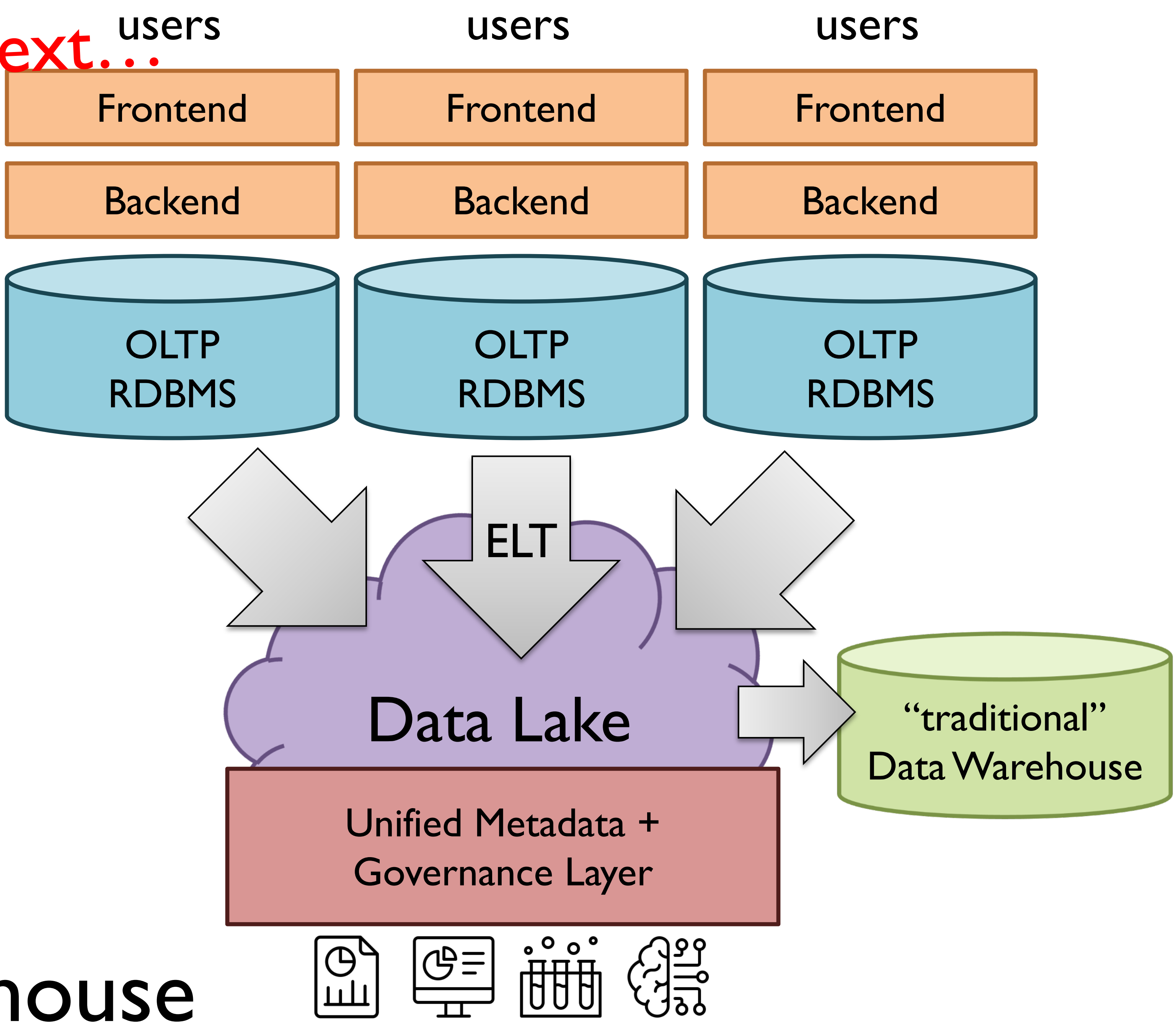
data platforms + data engineering

Context…

What problems do data platforms solve?

Ingesting, storing, manipulating, maintaining, serving…
the data that supports the data flywheel.

Context…



Lakehouse

Context…

Transform Insights into Actions

What does that really mean?

Report generation

Dashboards

Business Intelligence

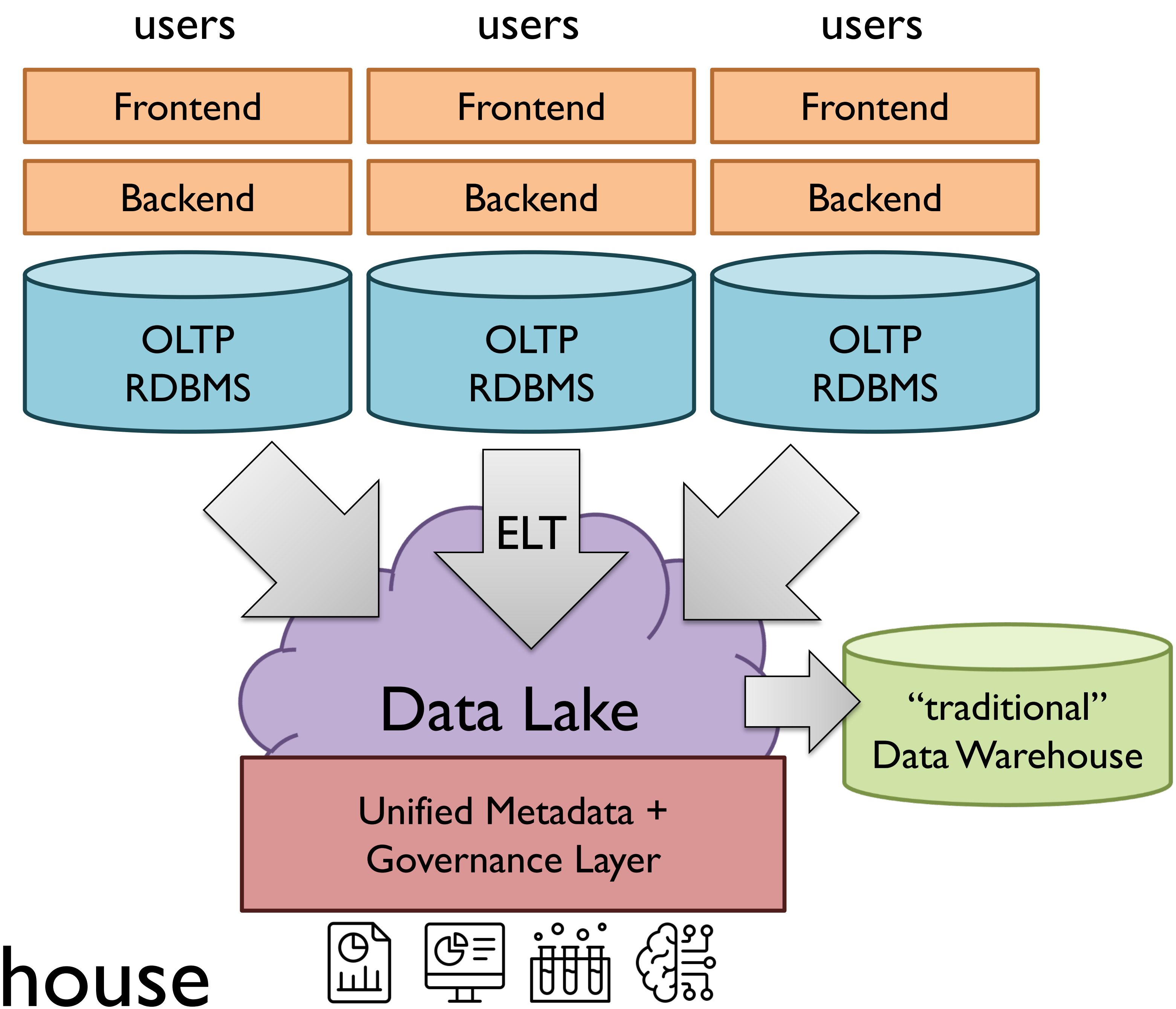
Ad hoc analyses

ML models

Data Science

Syllabus

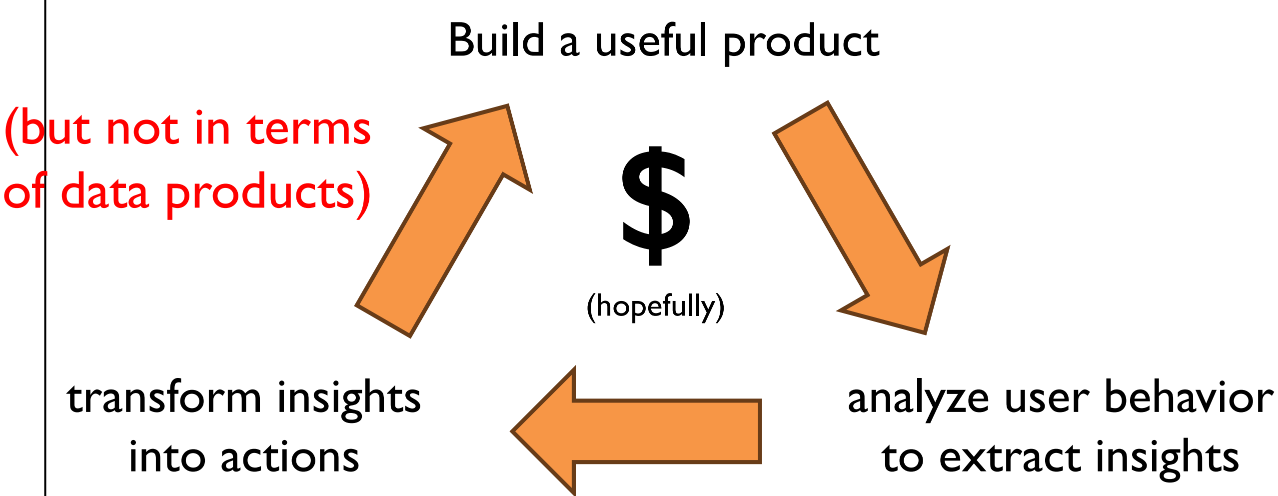
Week	Description
1	The Data Flywheel
2	Data Warehouses, Data Lakes, and Lakehouses
3	Batch Processing I
4	Batch Processing II
5	Rubber, Meet Road
6	Data Infrastructure for Machine Learning
7	**Reading Week**: No Classes!
8	Midterm Exam
9	Text Processing I
10	Text Processing II
11	Clustering
12	Graph Processing
13	Stream Processing
14	LLMs
	Final Exam


users
users
users
Frontend
Frontend
Frontend
Backend
Backend
Backend
OLTP RDBMS
OLTP RDBMS
OLTP RDBMS
ELT
Data Lake
"traditional" Data Warehouse
Unified Metadata + Governance Layer

Lakehouse

The Data Flywheel

(a virtuous cycle)



Transform Insights into Actions

What does that really mean?

Report generation
Dashboards
→ Business Intelligence

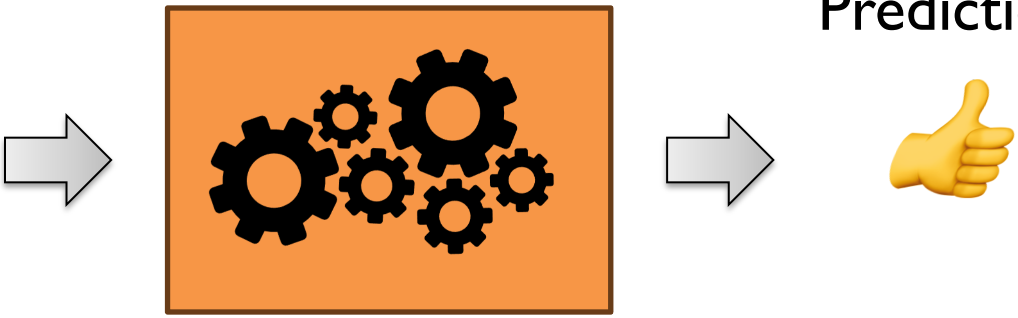
Ad hoc analyses
~~ML models~~
→ Data Science

Syllabus

Week	Description
1	The Data Flywheel
2	Data Warehouses, Data Lakes, and Lakehouses
3	Batch Processing I
4	Batch Processing II
5	Rubber, Meet Road
6	Data Infrastructure for Machine Learning
7	**Reading Week**: No Classes!
8	Midterm Exam
9	Text Processing I
10	Text Processing II
11	Clustering
12	Graph Processing
13	Stream Processing
14	LLMs
	Final Exam

Instance

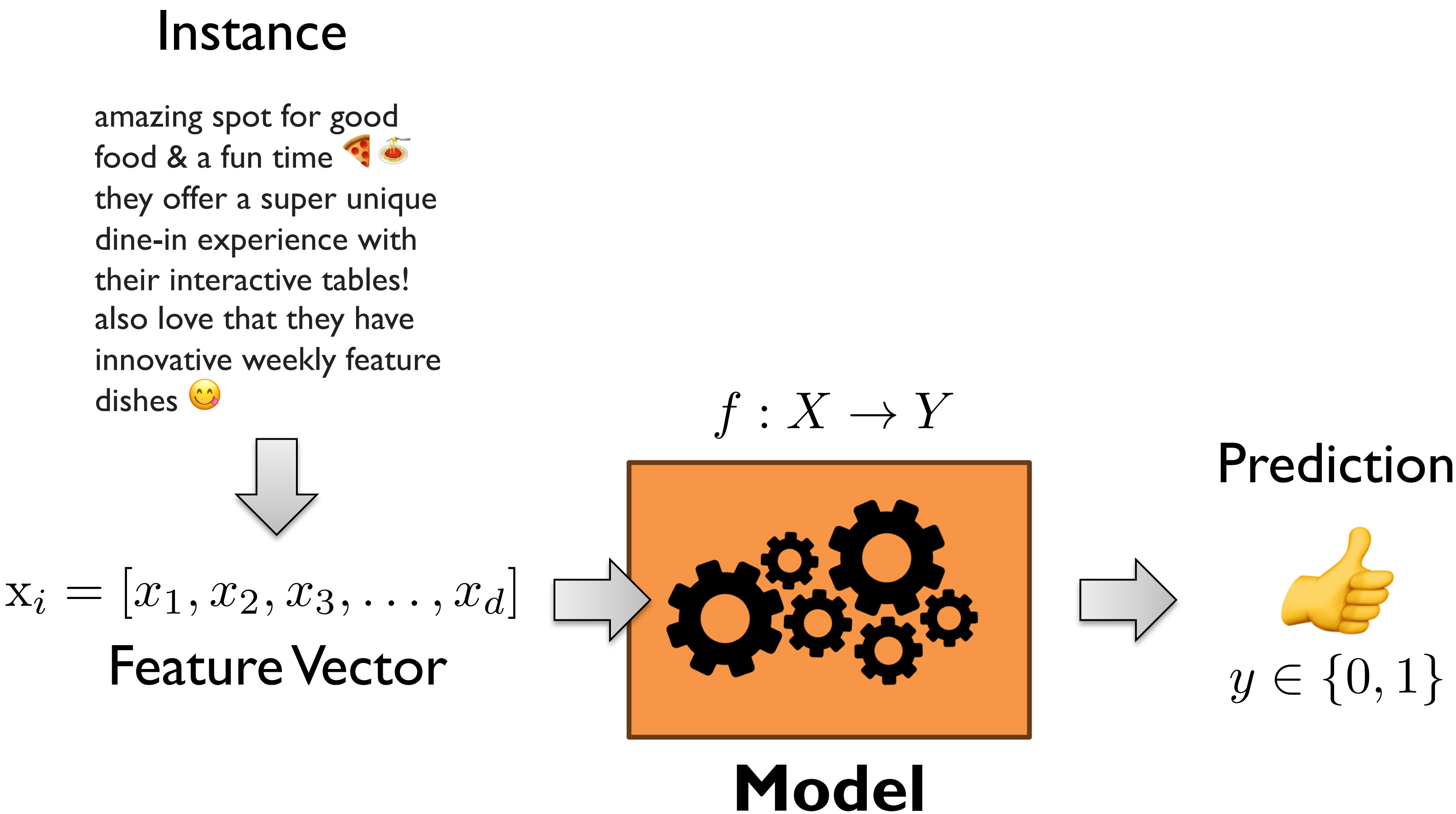
amazing spot for good food & a fun time 🍕🍝 they offer a super unique dine-in experience with their interactive tables! also love that they have innovative weekly feature dishes 😋



Model

Prediction

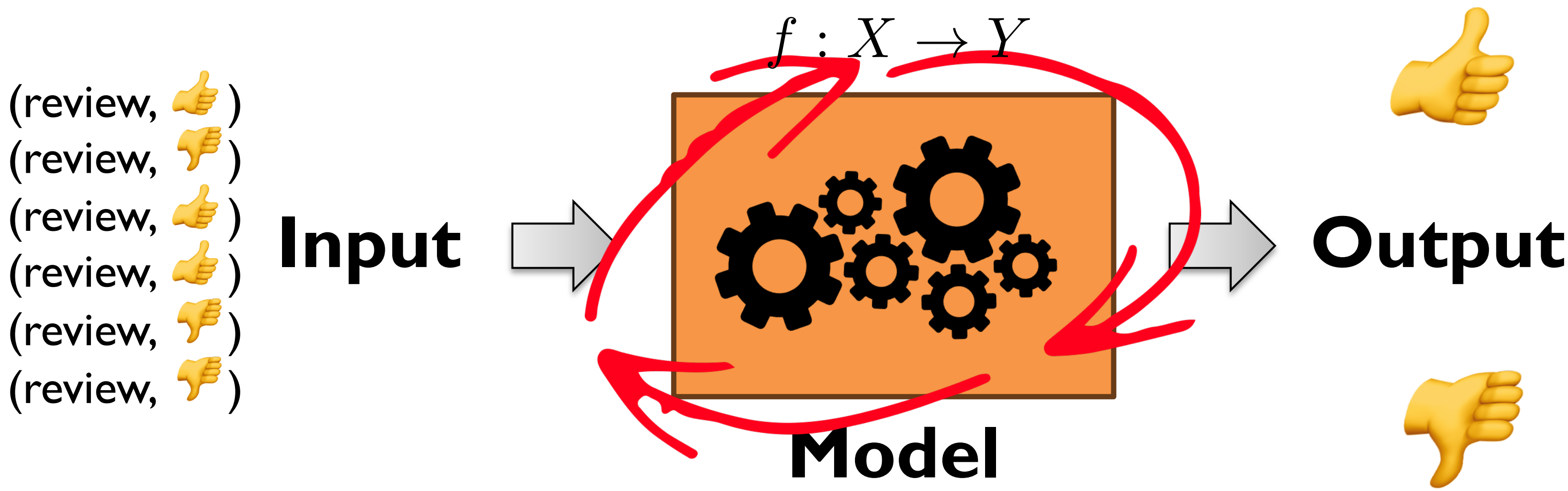
👍


Instance
amazing spot for good food & a fun time 🍕🍝 they offer a super unique dine-in experience with their interactive tables! also love that they have innovative weekly feature dishes 😋
$\mathrm{x}_i = [x_1, x_2, x_3, \ldots, x_d]$
Feature Vector
$f : X \rightarrow Y$
Model
Prediction
$y \in \{0, 1\}$

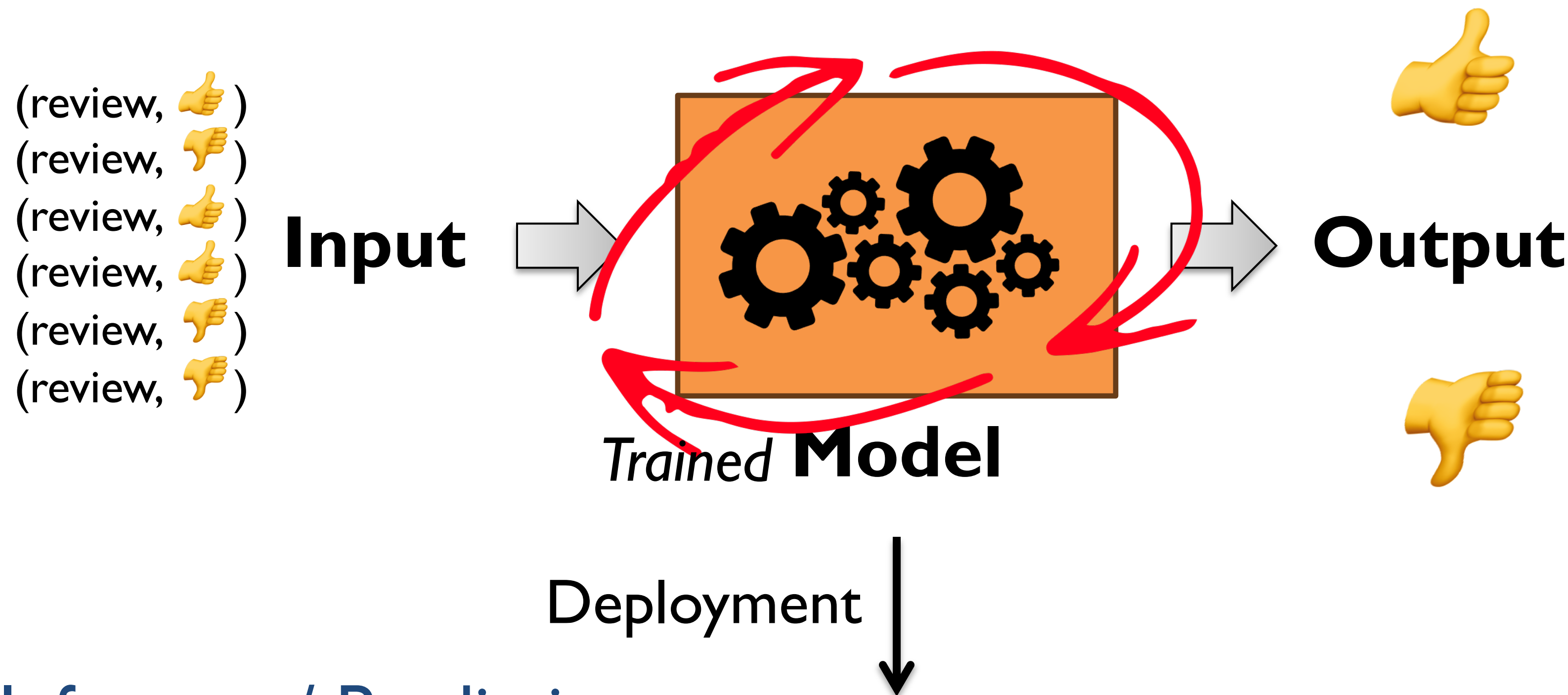
Components of an ML solution

(data, features, model, optimization)

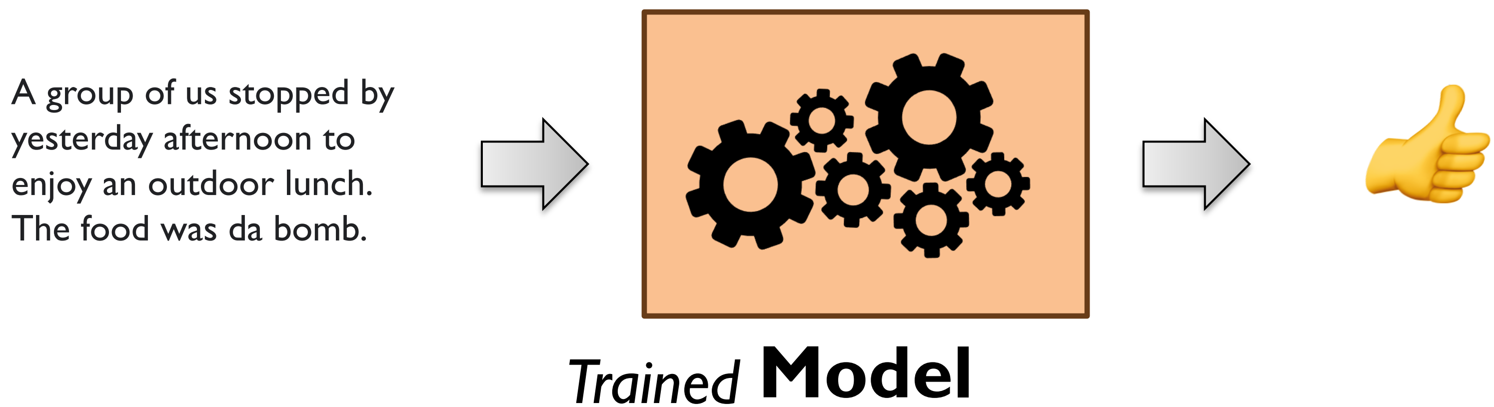
Model *learns* from the data



Machine learning algorithm adjusts the model *parameters*



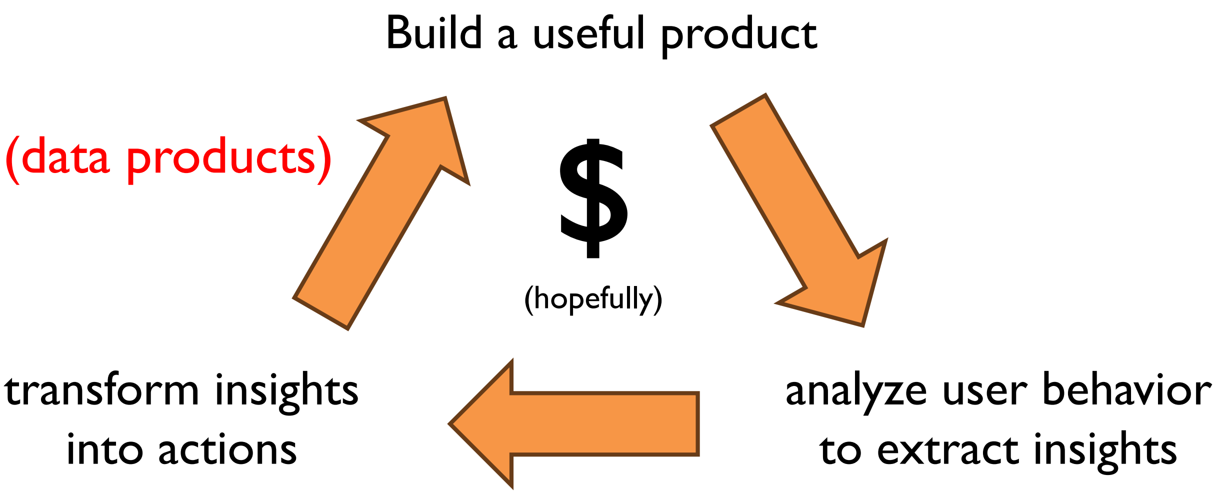
Inference / Prediction



Where might you actually deploy this?

The Data Flywheel

(a virtuous cycle)

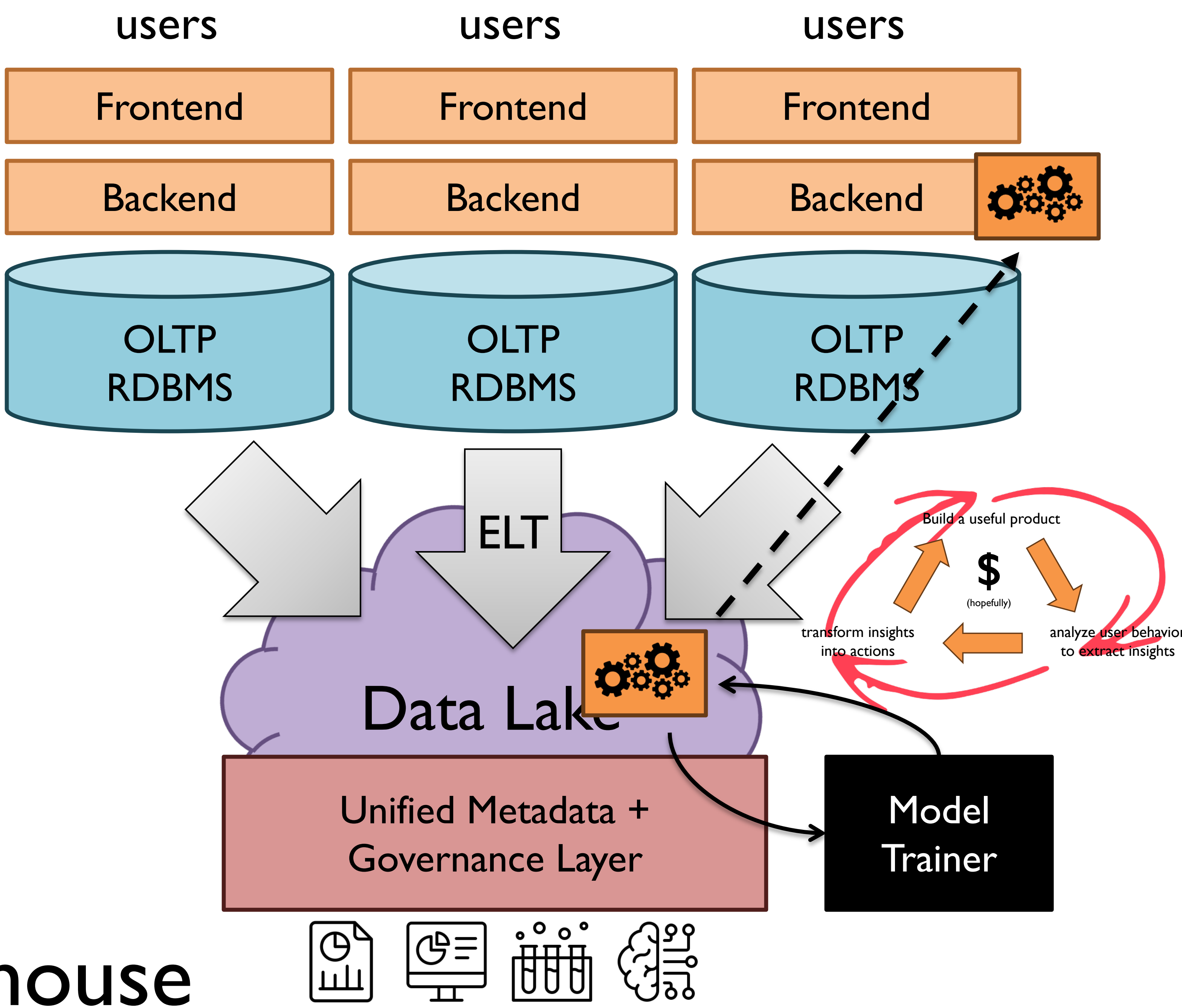


Transform Insights into Actions

What does that really mean?

Report generation
Dashboards
→ Business Intelligence

Ad hoc analyses
ML models
→ Data Science


users
users
users
Frontend
Frontend
Frontend
Backend
Backend
Backend
OLTP RDBMS
OLTP RDBMS
OLTP RDBMS
ELT
Data Lake
Unified Metadata + Governance Layer
Model Trainer
Build a useful product
$
(hopefully)
analyze user behavior to extract insights
transform insights into actions

Lakehouse

Syllabus

Week	Description
1	The Data Flywheel
2	Data Warehouses, Data Lakes, and Lakehouses
3	Batch Processing I
4	Batch Processing II
5	Rubber, Meet Road
6	Data Infrastructure for Machine Learning
7	**Reading Week**: No Classes!
8	Midterm Exam
9	Text Processing I
10	Text Processing II
11	Clustering
12	Graph Processing
13	Stream Processing
14	LLMs
	Final Exam

Remainder of the Semester

“Weird stuff” that doesn’t fit *traditional* lakehouses

Data products with operational requirements

Both!

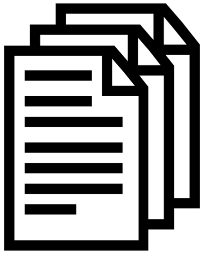
Recap: What are we doing and why?
Also: Where are we going and why?

What's the problem we're trying to solve?

How to connect users with relevant information

search (information retrieval)…
… but also question answering, summarization, etc.
"information access"

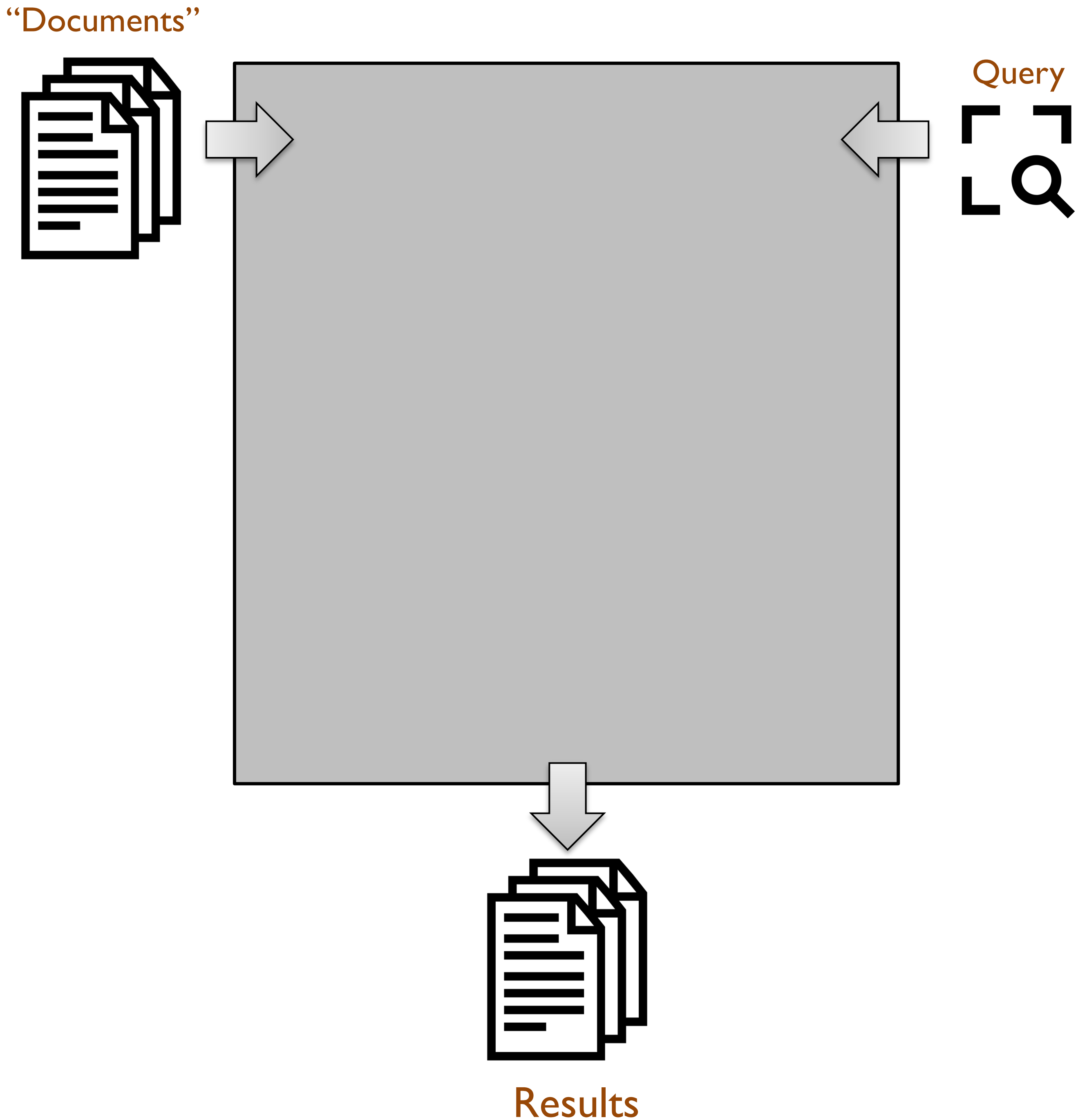
… on text, images, videos, etc.
… for "everyday" searchers, domain experts, etc.

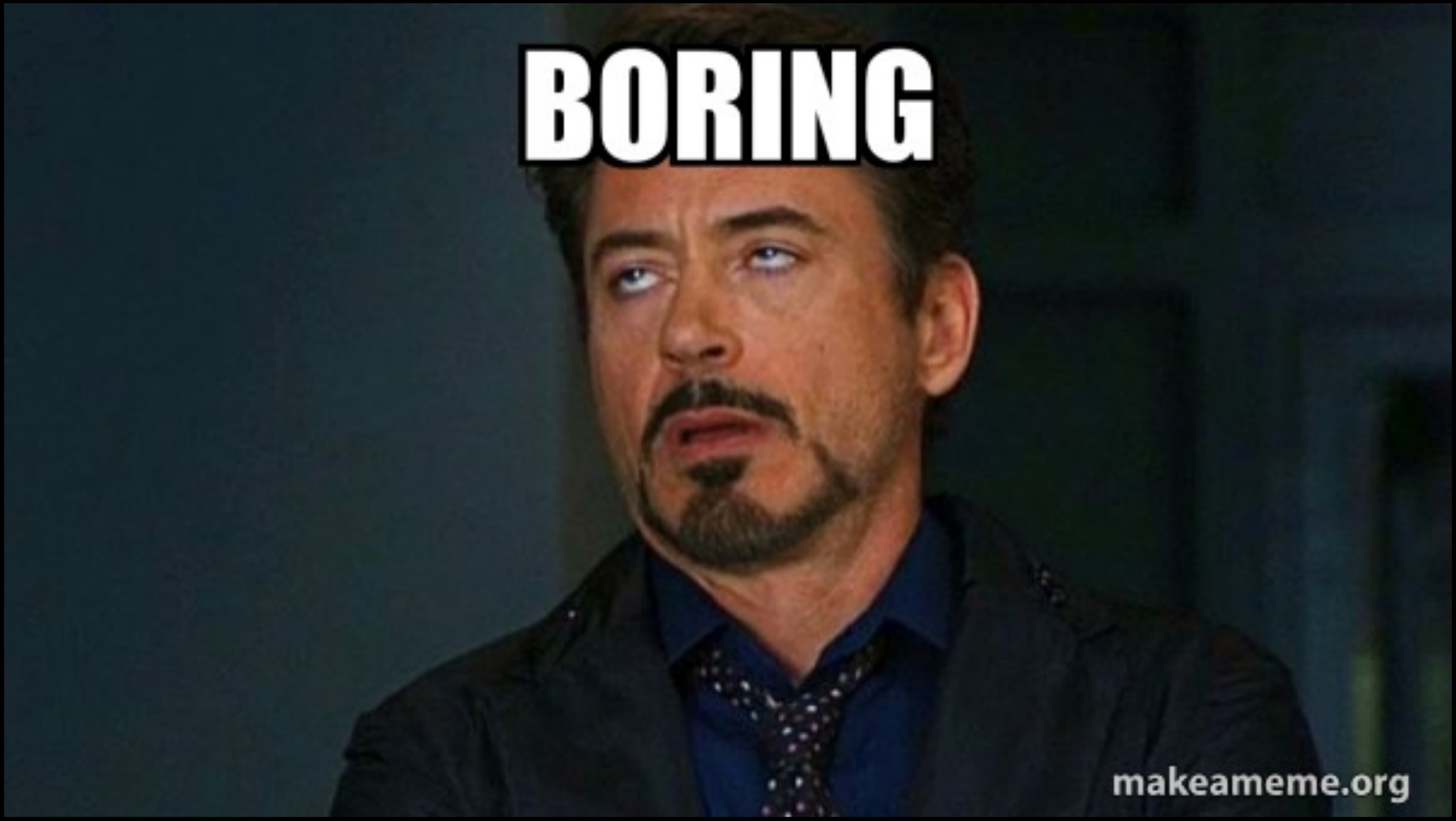


What's the problem we're trying to solve?

The "core" (text) retrieval problem

Given an information need expressed as a query q, the text retrieval task is to return a ranked list of k texts $\{d_1, d_2 \ldots d_k\}$ from an arbitrarily large but finite collection of texts $C = \{d_i\}$ that maximizes a metric of interest, for example, precision, nDCG, AP, etc.


"Documents"
Query
Results



That is so last millennium!

Source: https://www.businessinsider.com/heres-what-google-looked-like-the-first-day-it-launched-in-1998-2013-9

Google!

google | 10 results | Google Search | I'm feeling lucky

Showing results **1-10** of approximately **234,000** for **google**. Search took **0.06** seconds.

Take me back to 2013

Google (Google[RN])
...the web using **Google** Try our special searches: Uncle Sam Search millions...
...web's Linux resources ©1998 **Google** Inc....
www.google.com/ - Cached: 2k - GoogleScout

Google Search: <Unclesam>
...terms. Search the entire web from the **Google** home page! Copyright...
www.google.com/unclesam - Cached: 2k - GoogleScout

Google Search: <Linux>
...terms. Search the entire web from the **Google** home page! Copyright...
www.google.com/linux - Cached: 2k - GoogleScout

www.google.com/search
GoogleScout

Why Use Google?
...Why Use **Google**? Because **Google** delivers the most relevant search...
...search results--first and fast! **Google** uses sophisticated next-generation...
www.google.com/why_use.html - Cached: 8k -GoogleScout

www.google.com/netscape
GoogleScout

Google Help
...Basic Search To enter a query into **Google**, just type in a few descriptive...
...descriptive keywords and click on the **Google** Search button for your list...
www.google.com/help.html - Cached: 13k -GoogleScout

google.stanford.edu/
GoogleScout

Constructing and refining searches in Google: Detailed Searching Instructions
...refining searches in **Google**: Detailed Searching Instructions Teaching...
...University of California, Berkeley **Google** is a fairly new Web searching...
www.lib.berkeley.edu/TeachingLib/Guides/Internet/Google.html - Cached: 12k - GoogleScout

Source: https://www.businessinsider.com/heres-what-google-looked-like-the-first-day-it-launched-in-1998-2013-9

A MESSAGE FROM OUR CEO

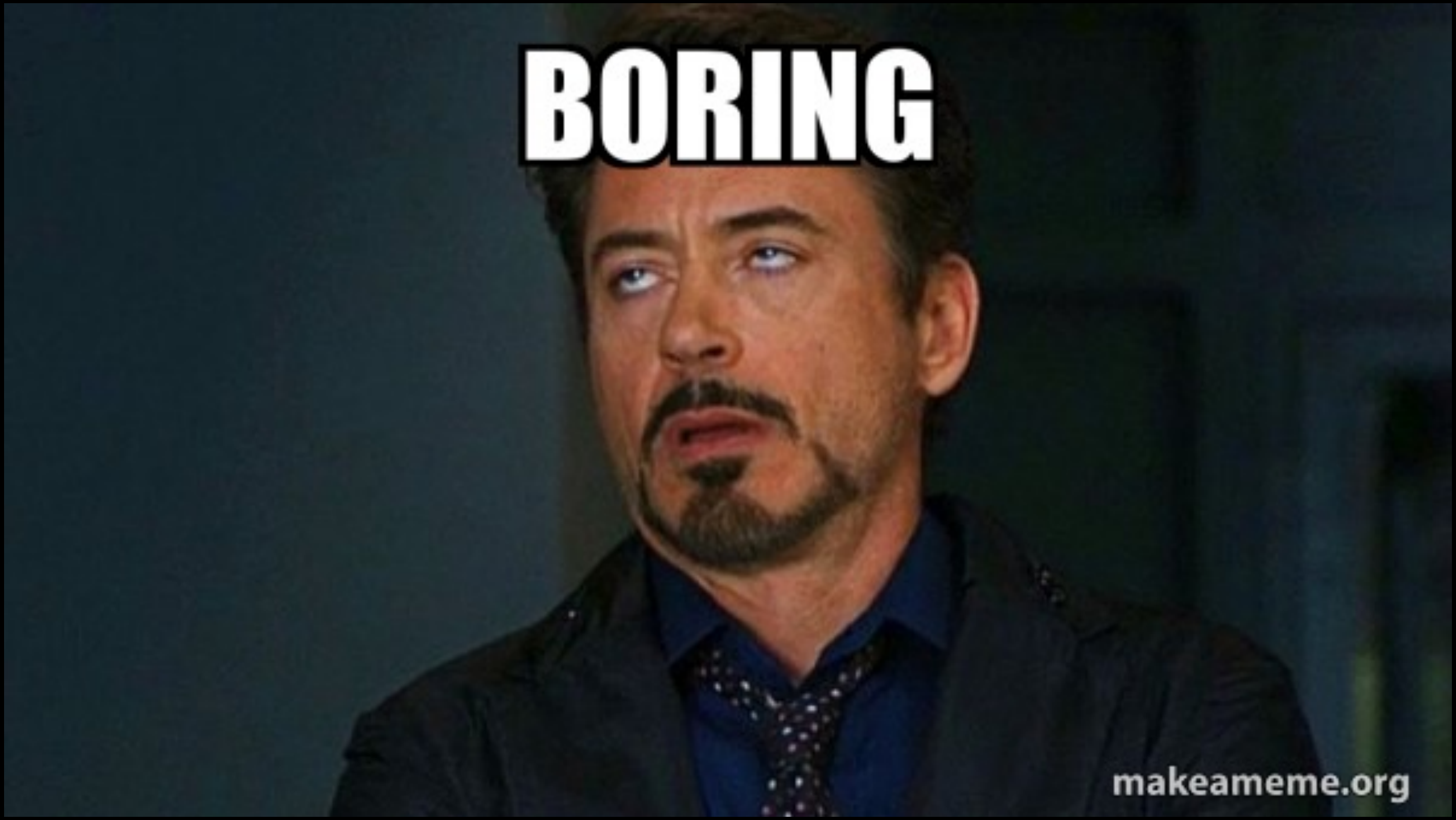
Questions, shrugs and what comes next: A quarter century of change

Sep 05, 2023 · 9 min read

Sundar Pichai
CEO of Google and Alphabet

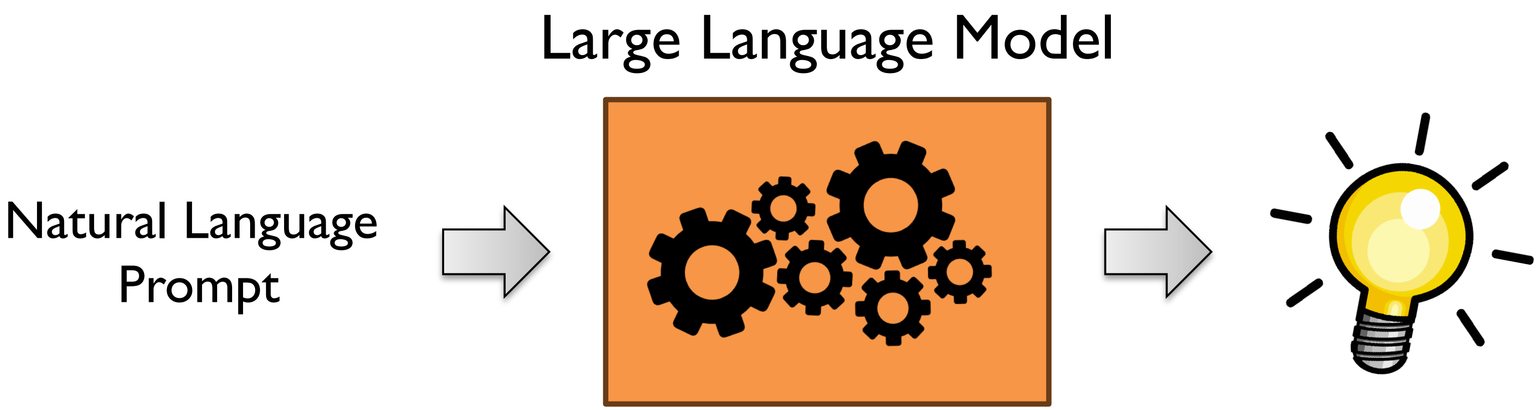
Share

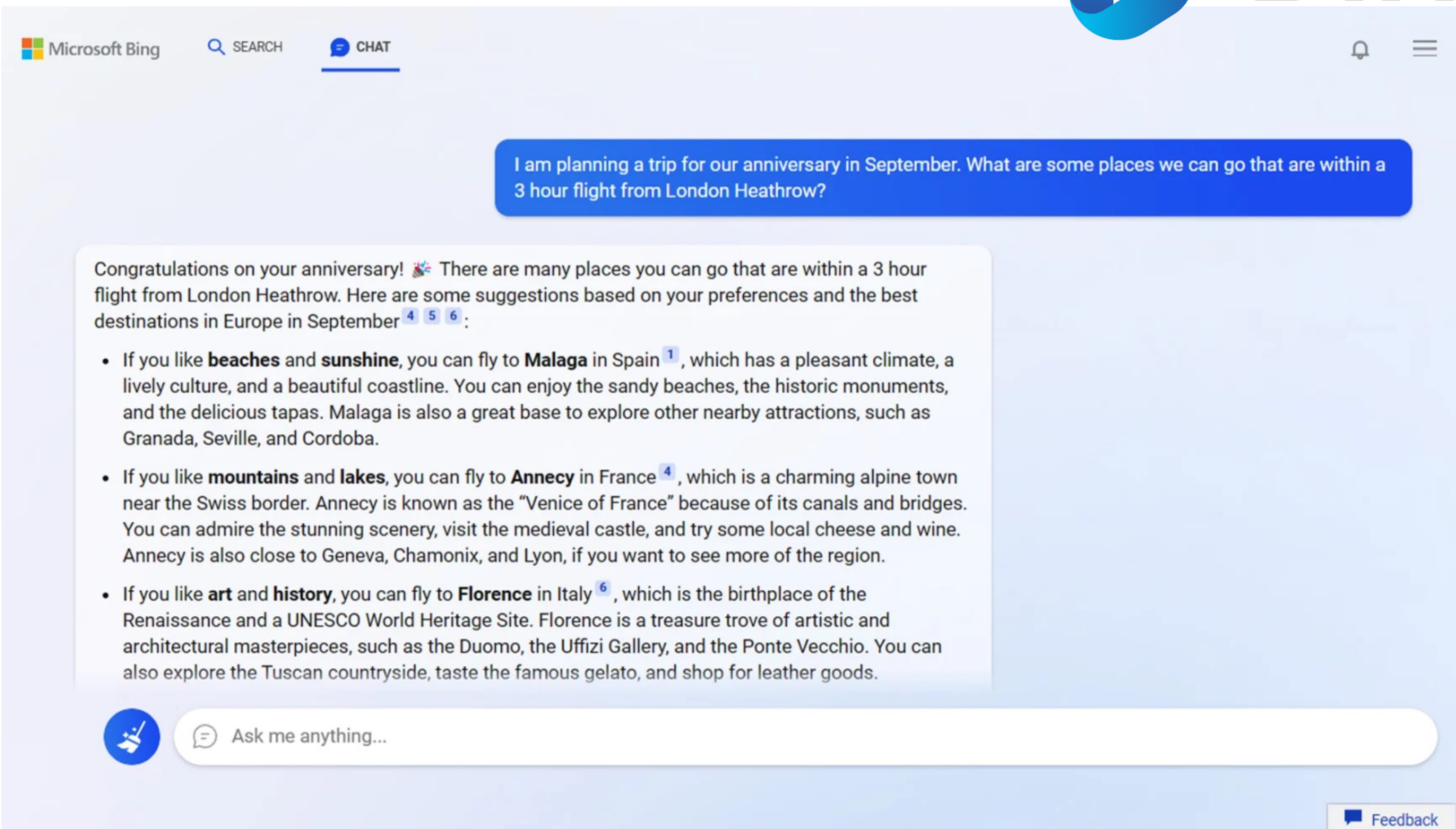
Source: https://blog.google/inside-google/message-ceo/google-25th-birthday-sundar-pichai/



But you know what's *not* boring?

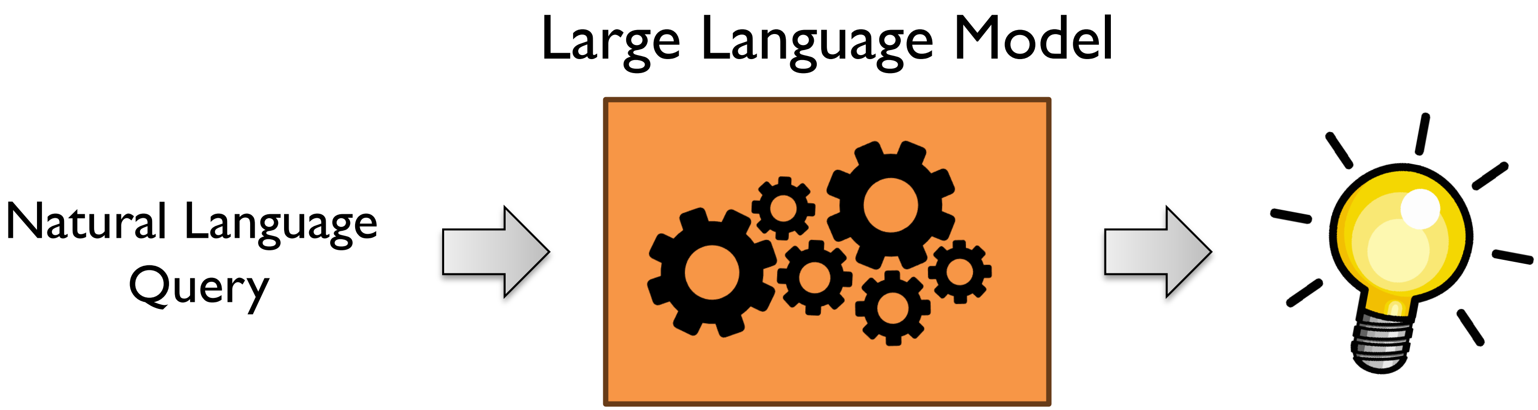
What's *not* boring?



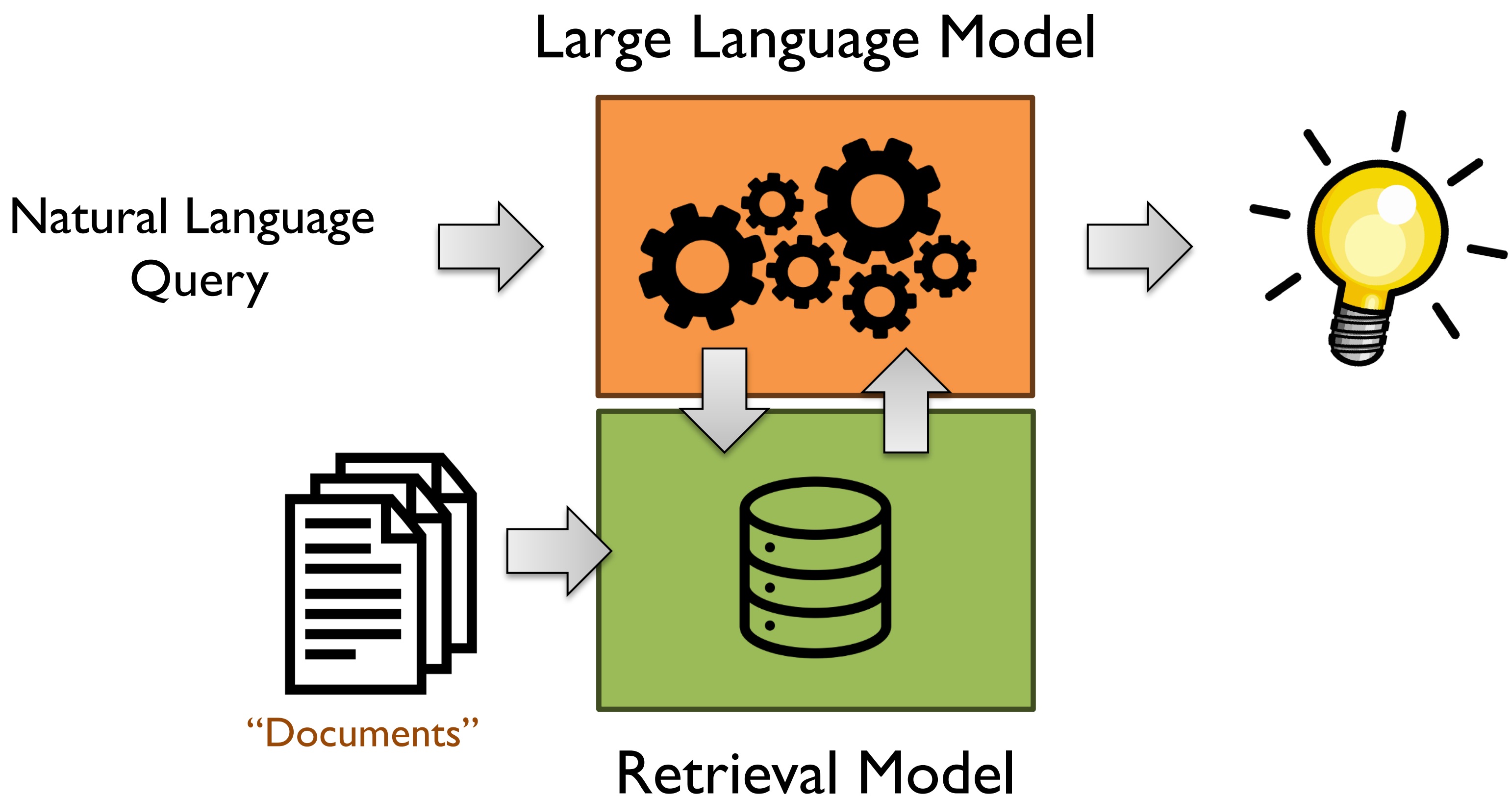


Source: https://www.engadget.com/microsofts-next-gen-bing-more-powerful-language-model-than-chatgpt-182647588.html

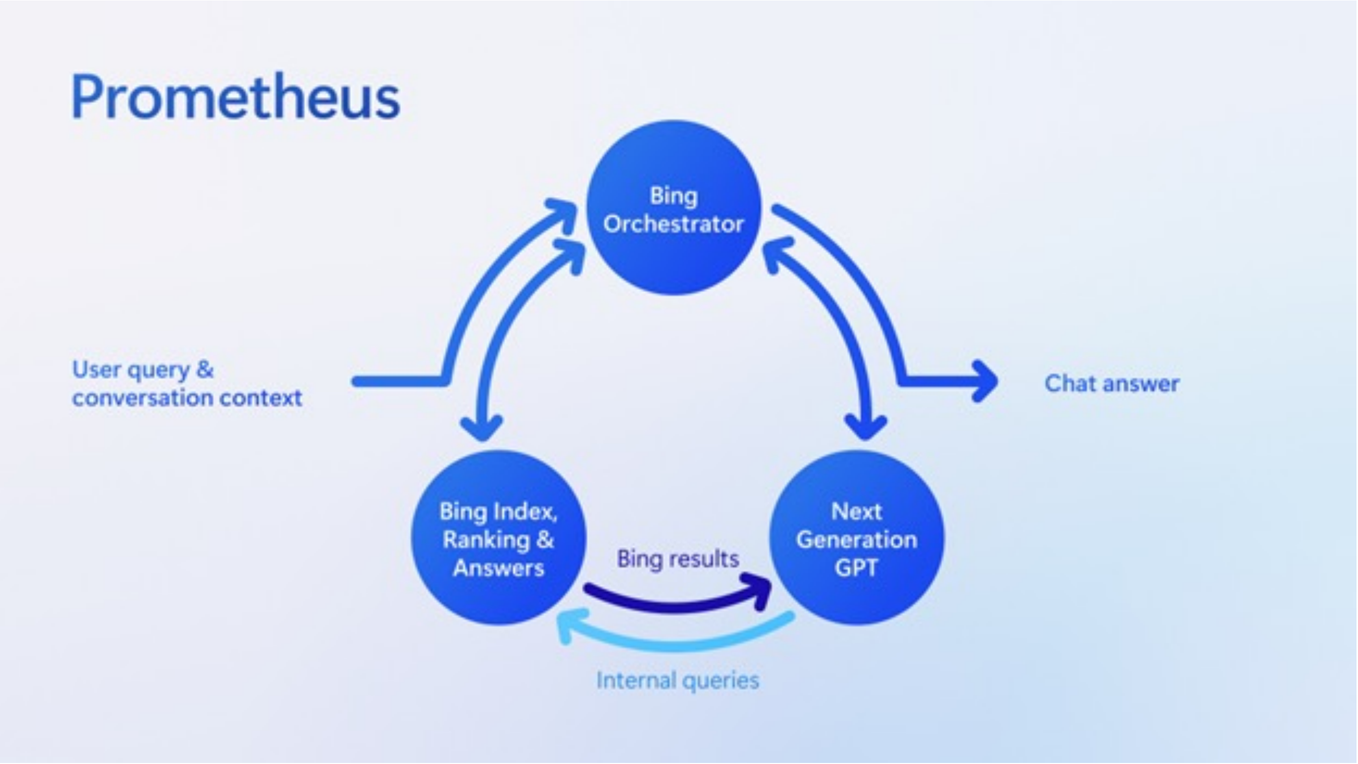
How does Sydney work?



How does Sydney work?

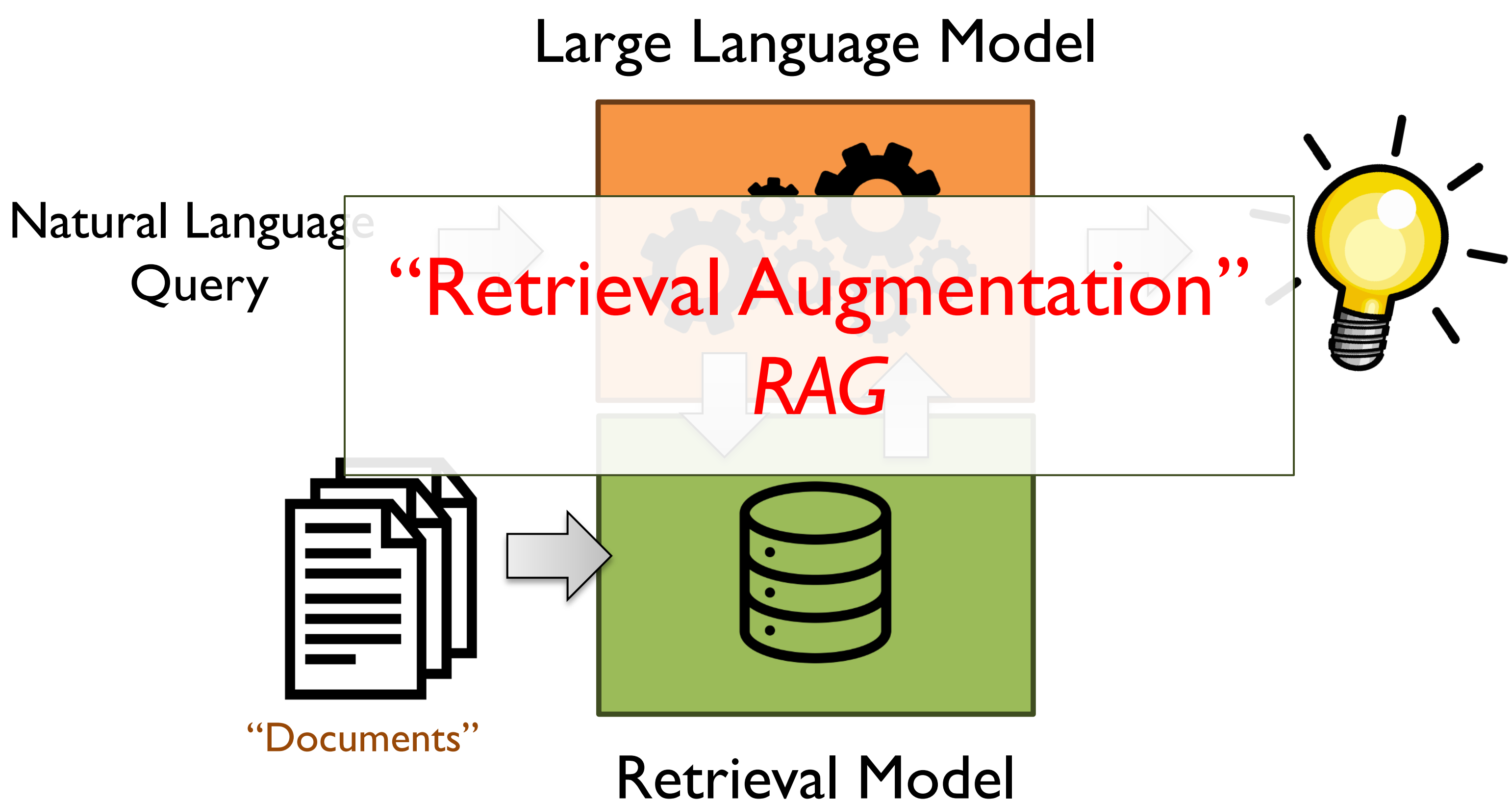


How does Sydney work?



Source: https://www.engadget.com/microsofts-next-gen-bing-more-powerful-language-model-than-chatgpt-182647588.html

How does Sydney work?



Why Retrieval Augmentation?

(combat) Hallucinations

(incorporate) Up-to-date information

(exploit) Private data

Why Retrieval Augmentation?

Generate a short biography of Jimmy Lin.

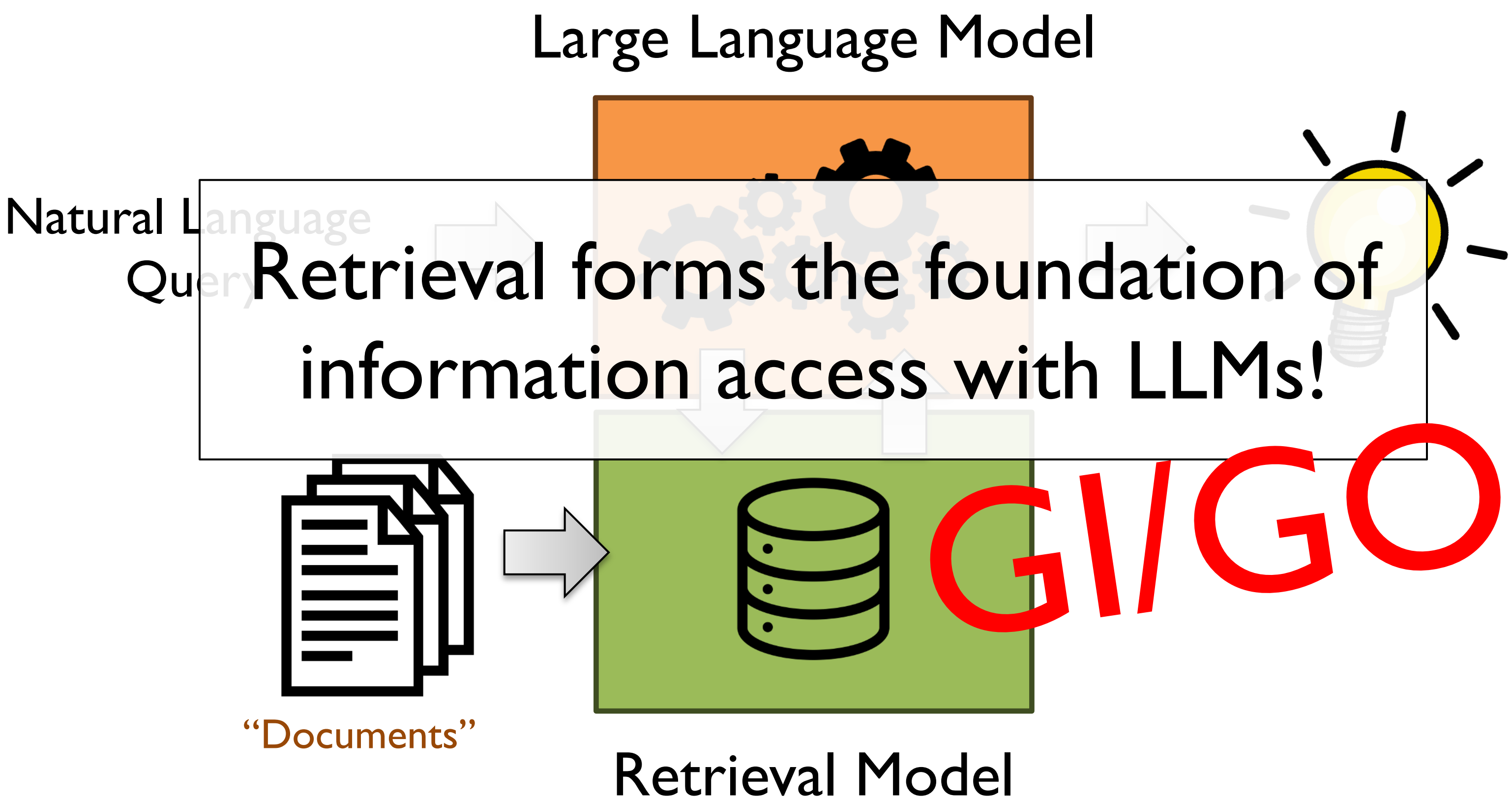
Given the following facts, generate a short biography of Jimmy Lin.

- Jimmy Lin is a professor at the University of Waterloo.
- Lin holds the David R. Cheriton Chair in the David R. Cheriton School of Computer Science.
- Jimmy Lin received his Ph.D. from MIT in 2004
- …

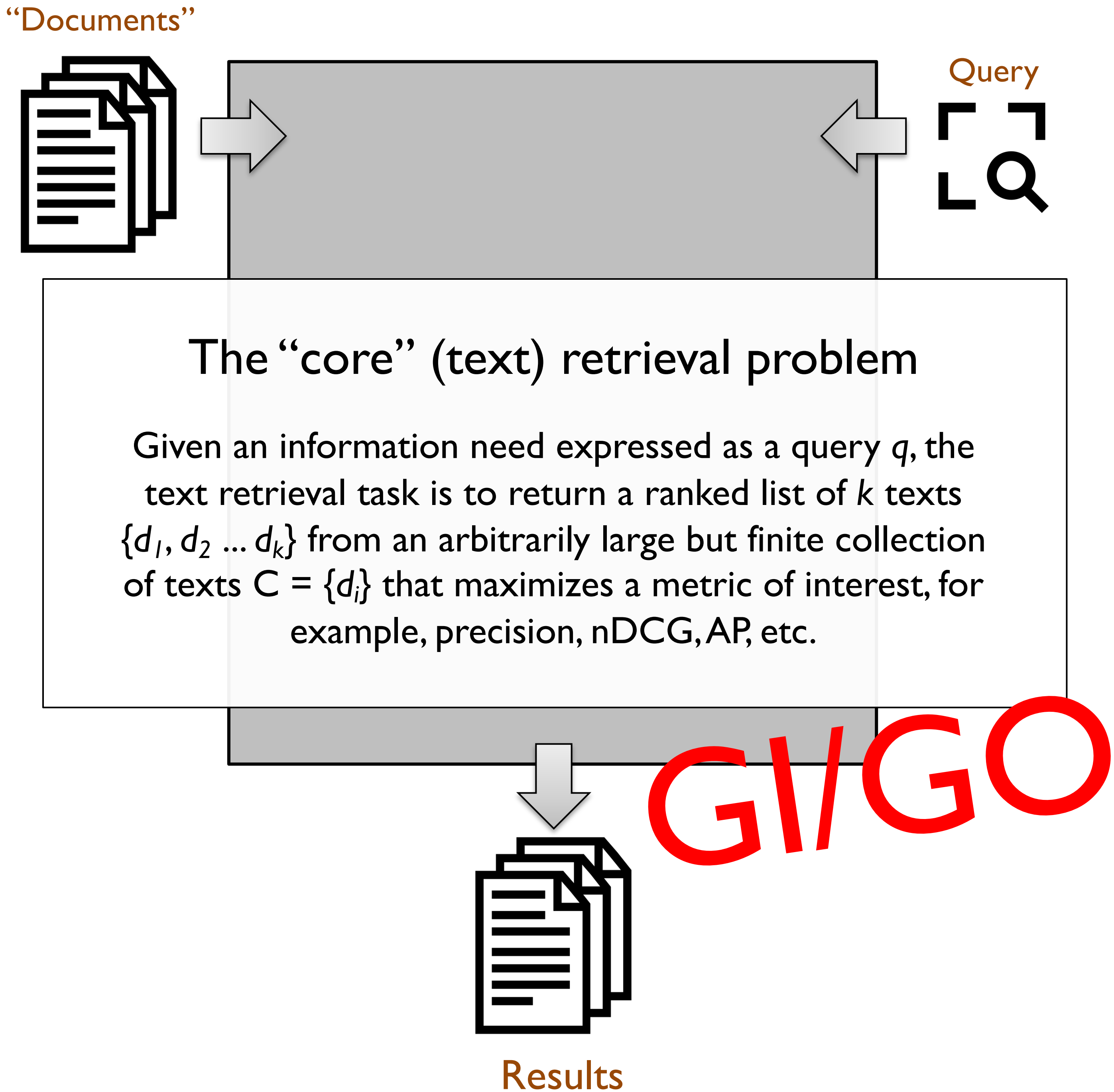
- He recently joined Yupp as Chief Scientist…

- His cell phone number is (519) 721-xxxx…

Why Retrieval Augmentation?

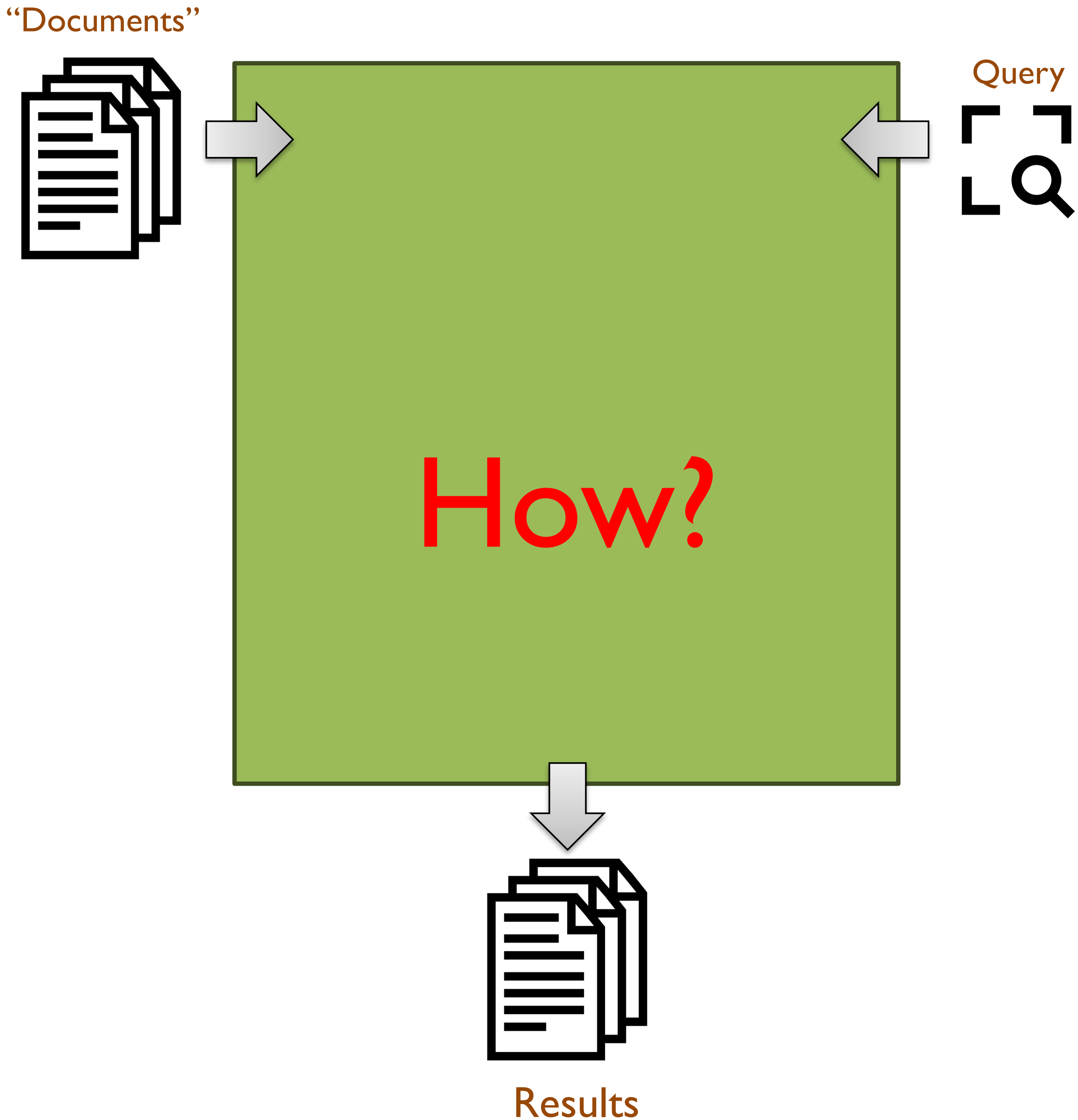


Retrieval forms the foundation of information access with LLMs!



The "core" (text) retrieval problem

Given an information need expressed as a query q, the text retrieval task is to return a ranked list of k texts $\{d_1, d_2 \ldots d_k\}$ from an arbitrarily large but finite collection of texts $C = \{d_i\}$ that maximizes a metric of interest, for example, precision, nDCG, AP, etc.



“Documents”
Query
How?
Results

A Vector Space Model for Automatic Indexing

G. Salton, A. Wong
and C. S. Yang
Cornell University

In a document retrieval, or other pattern matching environment where stored entities (documents) are compared with each other or with incoming patterns (search requests), it appears that the best indexing (property) space is one where each entity lies as far away from the others as possible; in these circumstances the value of an indexing system may be expressible as a function of the density of the object space; in particular, retrieval performance may correlate inversely with space density. An approach based on space density computations is used to choose an optimum indexing vocabulary for a collection of documents. Typical evaluation results are shown, demonstating the usefulness of the model.

Key Words and Phrases: automatic information retrieval, automatic indexing, content analysis, document space

CR Categories: 3.71, 3.73, 3.74, 3.75

This study was supported in part by the National Science Foundation under grant GN 43505. Authors' addresses: G. Salton and A. Wong, Department of Computer Science, Cornell University, Ithaca, NY 14850; C. S. Yang, Department of Computer Science, The University of Iowa, Iowa City, IA, 52240.

[1] Although we speak of documents and index terms, the present development applies to any set of entities identified by weighted property vectors.

[2] Retrieval performance is often measured by parameters such as *recall* and *precision*, reflecting the ratio of relevant items actually retrieved and of retrieved items actually relevant. The question concerning optimum space configurations may then be more conventionally expressed in terms of the relationship between document indexing, on the one hand, and retrieval performance, on the other.

1. Document Space Configurations

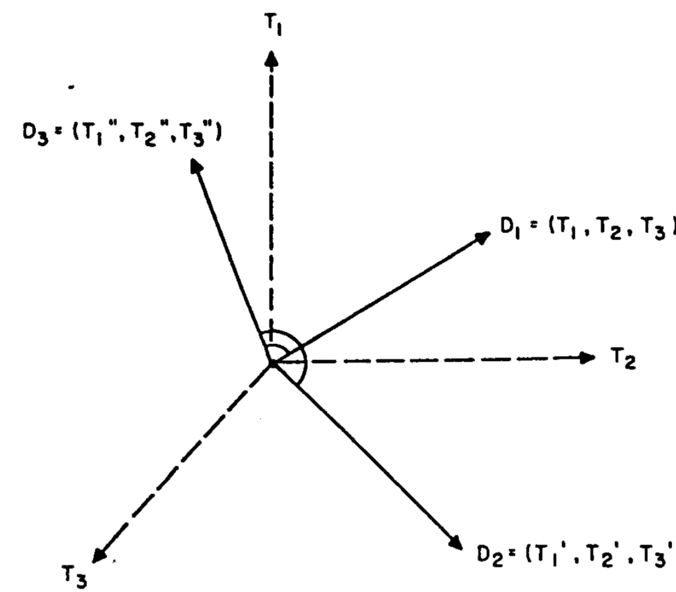
Consider a document space consisting of documents D_i, each identified by one or more index terms T_j; the terms may be weighted according to their importance, or unweighted with weights restricted to 0 and 1.[1] A typical three-dimensional index space is shown in Figure 1, where each item is identified by up to three distinct terms. The three-dimensional example may be extended to t dimensions when t different index terms are present. In that case, each document D_i is represented by a t-dimensional vector

$$D_i = (d_{i1}, d_{i2}, \ldots, d_{it}),$$

d_{ij} representing the weight of the jth term.

Given the index vectors for two documents, it is possible to compute a similarity coefficient between them, $s(D_i, D_j)$, which reflects the degree of similarity in the corresponding terms and term weights. Such a similarity measure might be the inner product of the two vectors, or alternatively an inverse function of the angle between
term assignm
will be zero, p
Instead of
vector origina
tem, the rela
served by no
considering th
velope of the
that case, eac
point whose p
correspondin
of the space.
are then repr
gether in the
tween two do
correlated wi
ing vectors.
Since the
function of th
are assigned
one may ask
configuration
optimum retr
If nothing
under consid
ideal docume
jointly releva
together, thu
jointly in res
trariwise, do

Fig. 1. Vector representation of document space.



Salton et al. (1975) A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 18(11):613-620.

A Vector Space Model for Automatic Indexing

1. Document Space Configurations

Consider a document space consisting of documents

index terms are present. In that case, each document D_i is represented by a t-dimensional vector

$$D_i = (d_{i1}, d_{i2}, \ldots, d_{it}),$$

d_{ij} representing the weight of the jth term.

Given the index vectors for two documents, it is possible to compute a similarity coefficient between them, $s(D_i, D_j)$, which reflects the degree of similarity in the corresponding terms and term weights. Such a similarity measure might be the inner product of the two vectors, or alternatively an inverse function of the angle between the corresponding vector pairs; when the

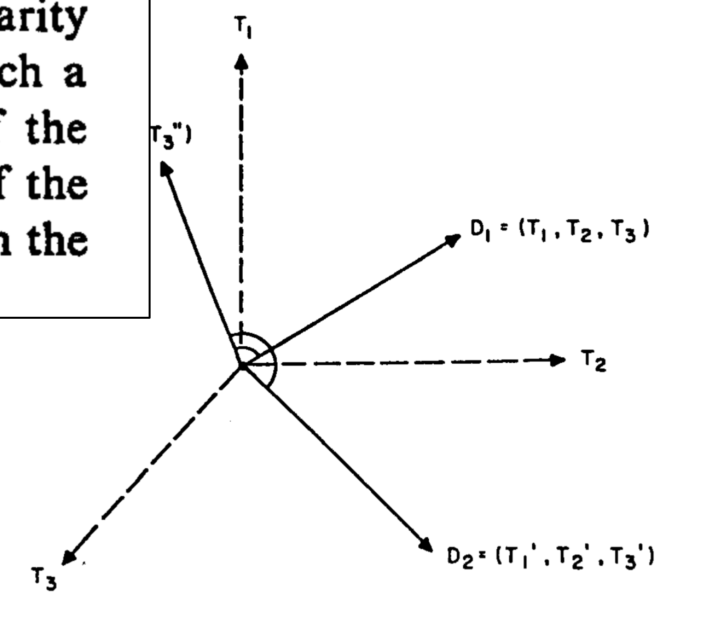
of this material is granted provided that ACM's copyright notice is given and that reference is made to the publication, to its date of issue, and to the fact that reprinting privileges were granted by permission of the Association for Computing Machinery.

This study was supported in part by the National Science Foundation under grant GN 43505. Authors' addresses: G. Salton and A. Wong, Department of Computer Science, Cornell University, Ithaca, NY 14850; C. S. Yang, Department of Computer Science, The University of Iowa, Iowa City, IA, 52240.

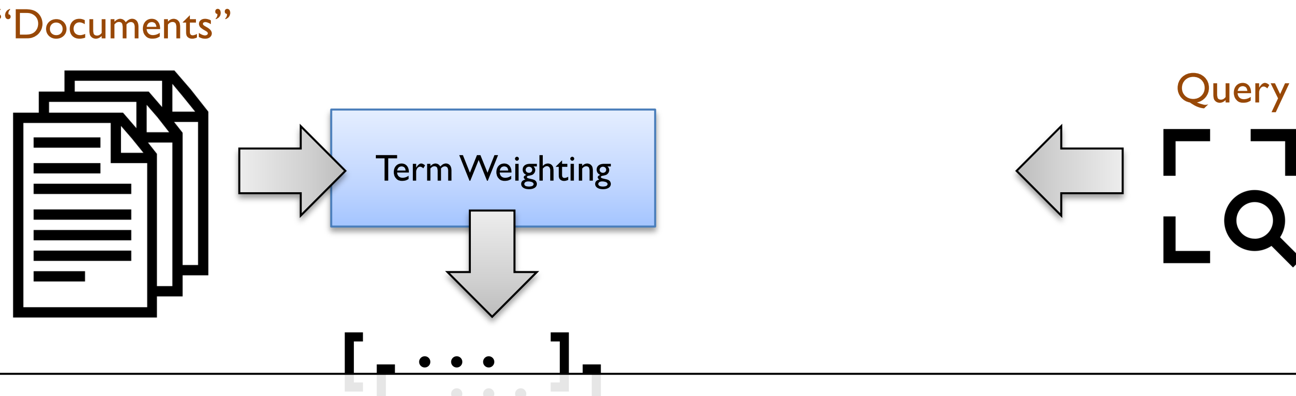
[1] Although we speak of documents and index terms, the present development applies to any set of entities identified by weighted property vectors.

[2] Retrieval performance is often measured by parameters such as *recall* and *precision*, reflecting the ratio of relevant items actually retrieved and of retrieved items actually relevant. The question concerning optimum space configurations may then be more conventionally expressed in terms of the relationship between document indexing, on the one hand, and retrieval performance, on the other.

presentation of document space.

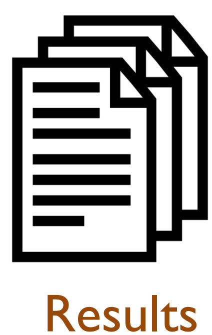


Salton et al. (1975) A Vector Space Model for Automatic Indexing.
Communications of the ACM, 18(11):613-620.



The Manhattan Project and its atomic bomb helped bring an end to World War II. Its legacy of peaceful uses of atomic energy continues to have an impact on history and science.

```
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu':
2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help':
1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it':
2.0473, 'legaci': 4.1335, 'manhattan': 4.1345, 'peac': 3.5205,
'project': 2.6442, 'scienc': 2.8700, 'us': 0.9967, 'war':
2.6454, 'world': 1.9974}
```

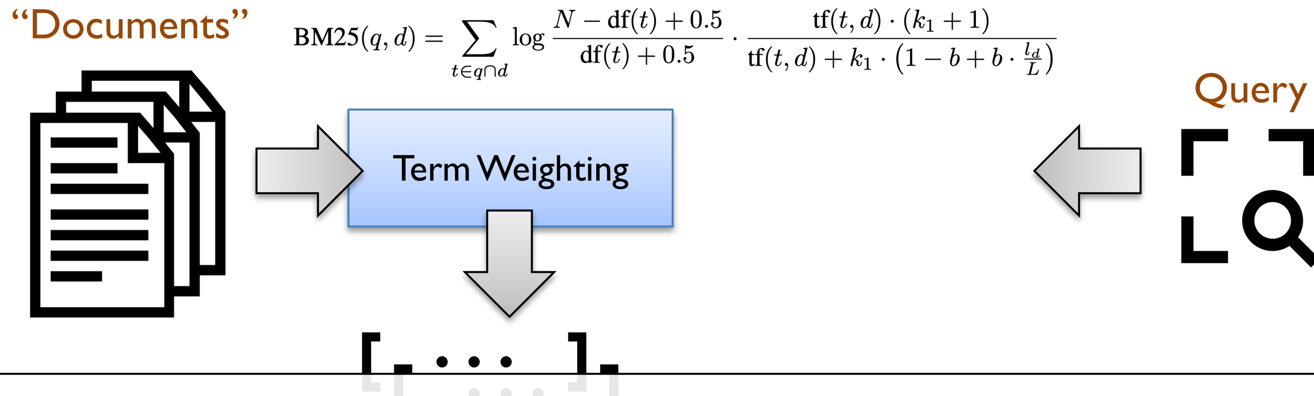


"bag of words"

"lexical"

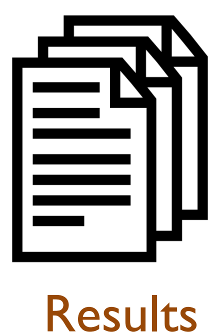
sparse vector

$$\mathrm{BM25}(q,d) = \sum_{t \in q \cap d} \log \frac{N - \mathrm{df}(t) + 0.5}{\mathrm{df}(t) + 0.5} \cdot \frac{\mathrm{tf}(t,d) \cdot (k_1 + 1)}{\mathrm{tf}(t,d) + k_1 \cdot \left(1 - b + b \cdot \frac{l_d}{L}\right)}$$



The Manhattan Project and its atomic bomb helped bring an end to World War II. Its legacy of peaceful uses of atomic energy continues to have an impact on history and science.

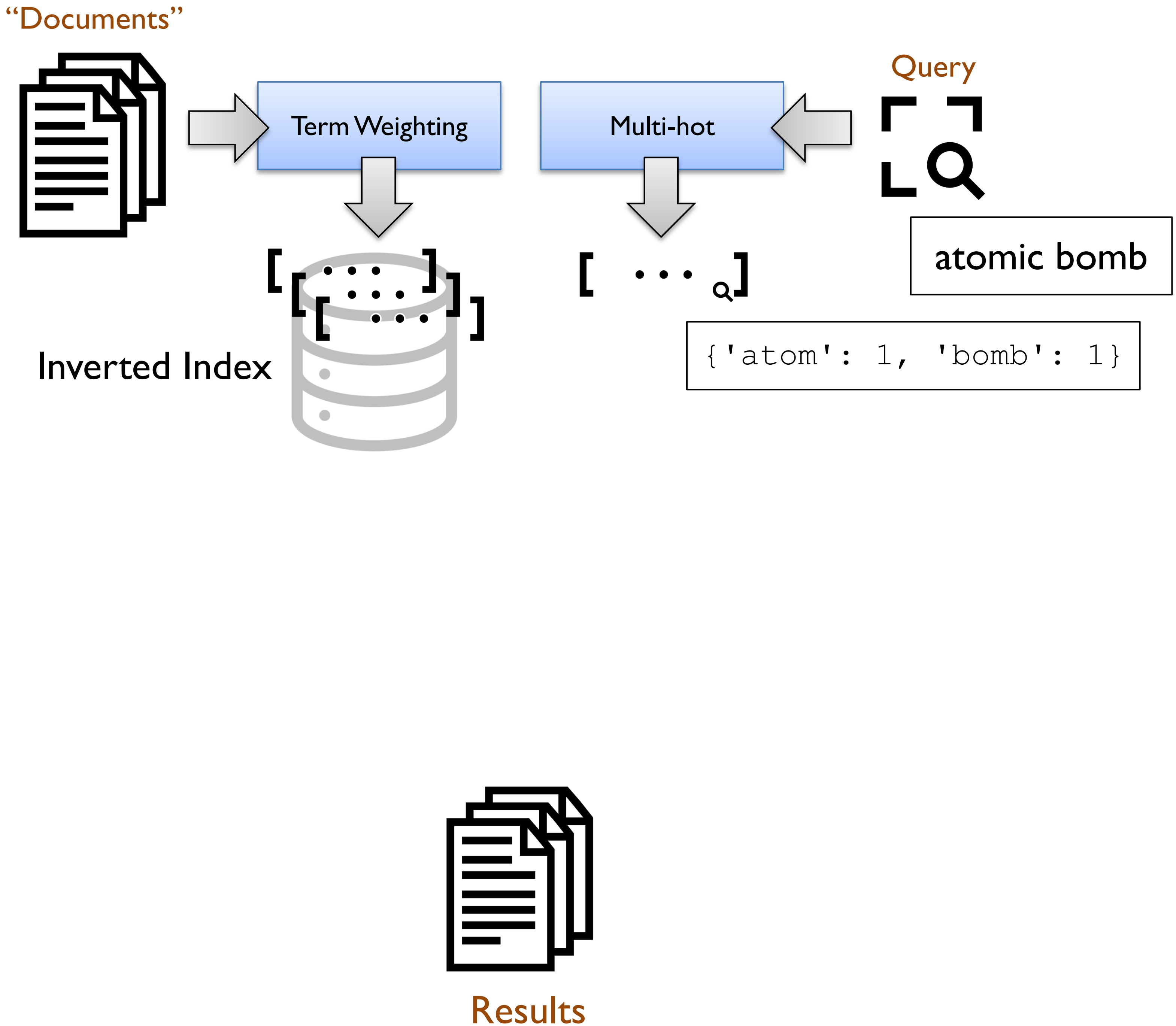
```
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu':
2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help':
1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it':
2.0473, 'legaci': 4.1335, 'manhattan': 4.1345, 'peac': 3.5205,
'project': 2.6442, 'scienc': 2.8700, 'us': 0.9967, 'war':
2.6454, 'world': 1.9974}
```

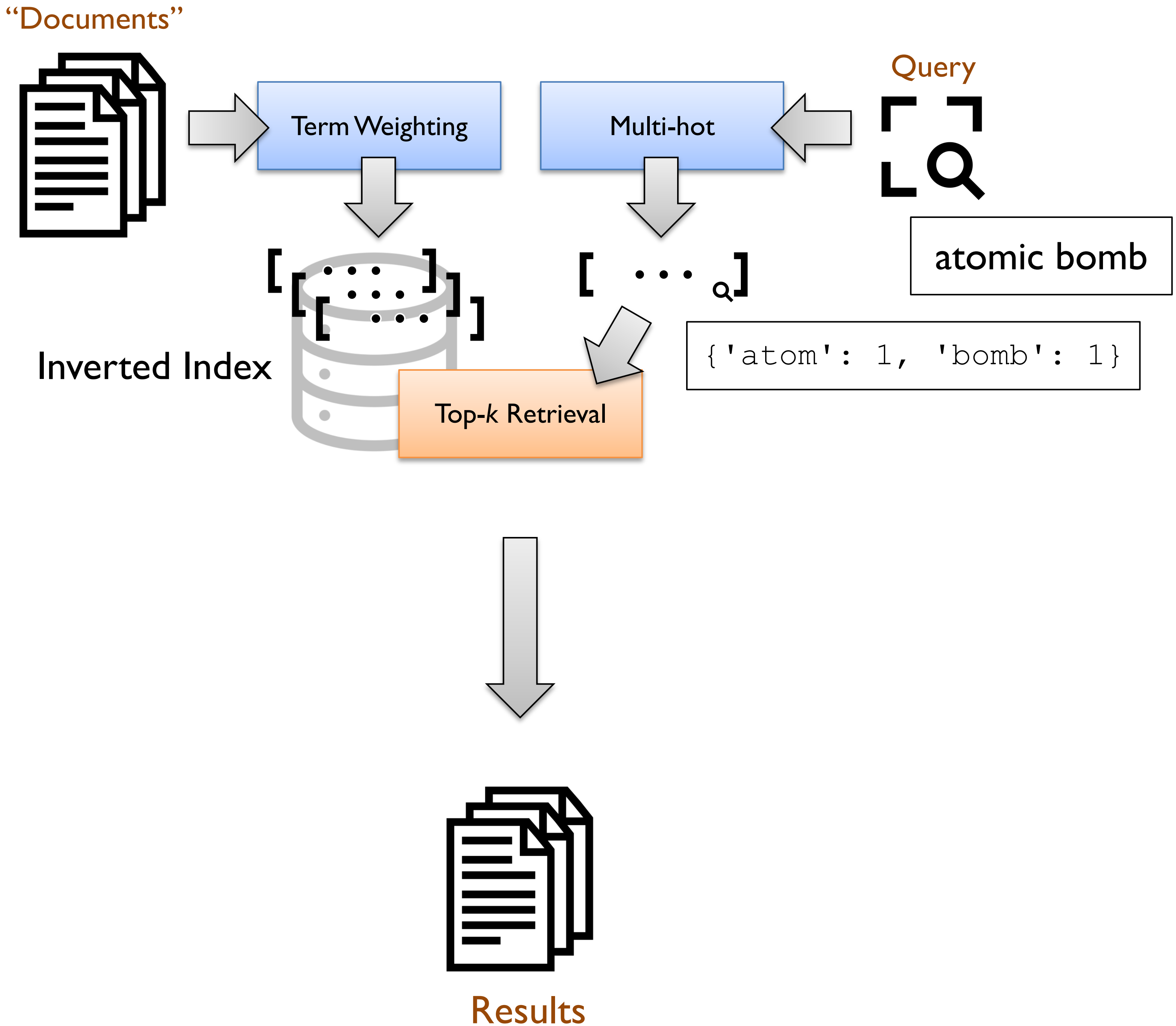


"bag of words"

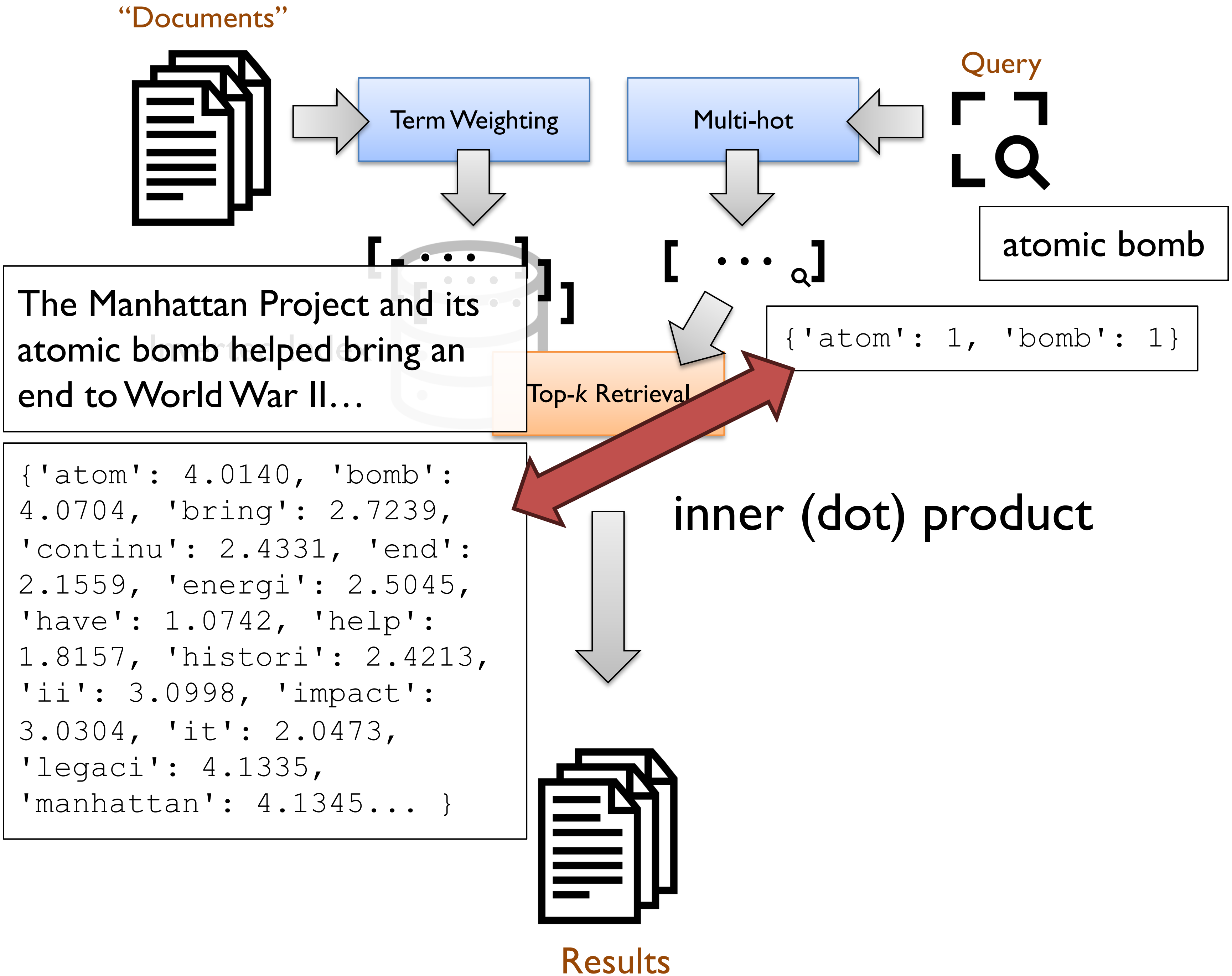
"lexical"

sparse vector

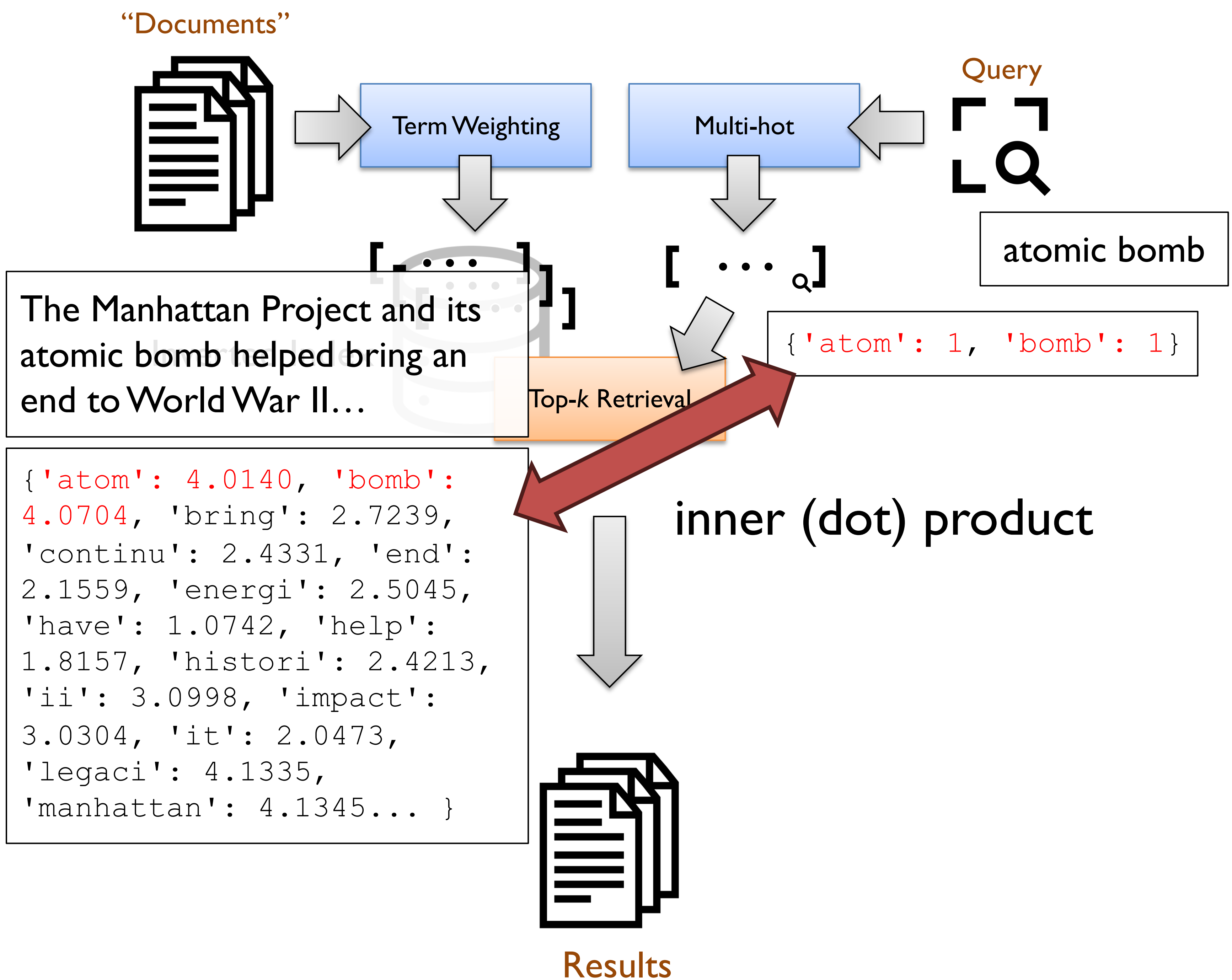

"Documents"
Term Weighting
Multi-hot
Query
atomic bomb
Inverted Index
{'atom': 1, 'bomb': 1}
Results

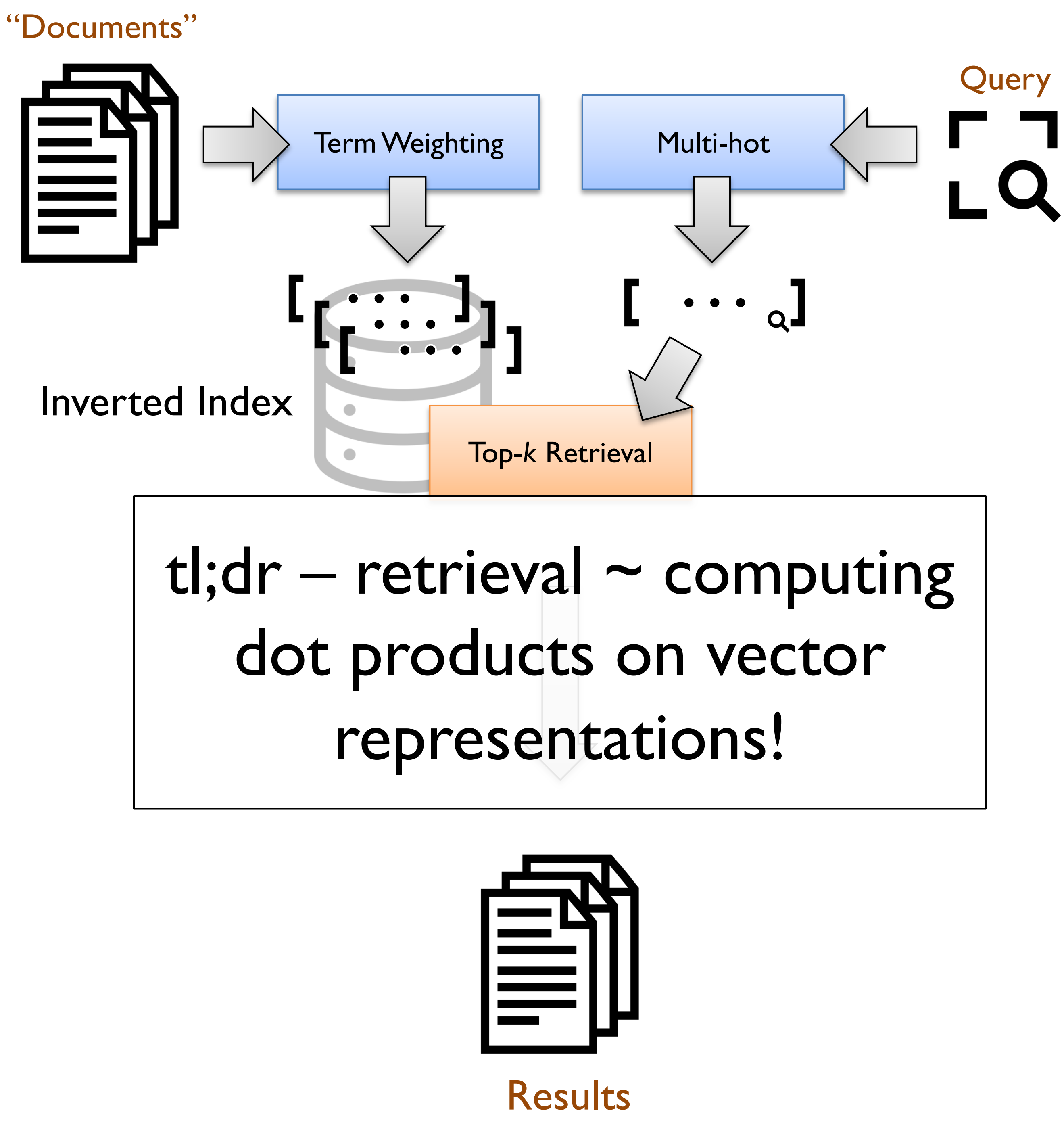


“Documents”
Term Weighting
Multi-hot
Query
atomic bomb
{'atom': 1, 'bomb': 1}
Inverted Index
Top-k Retrieval
Results

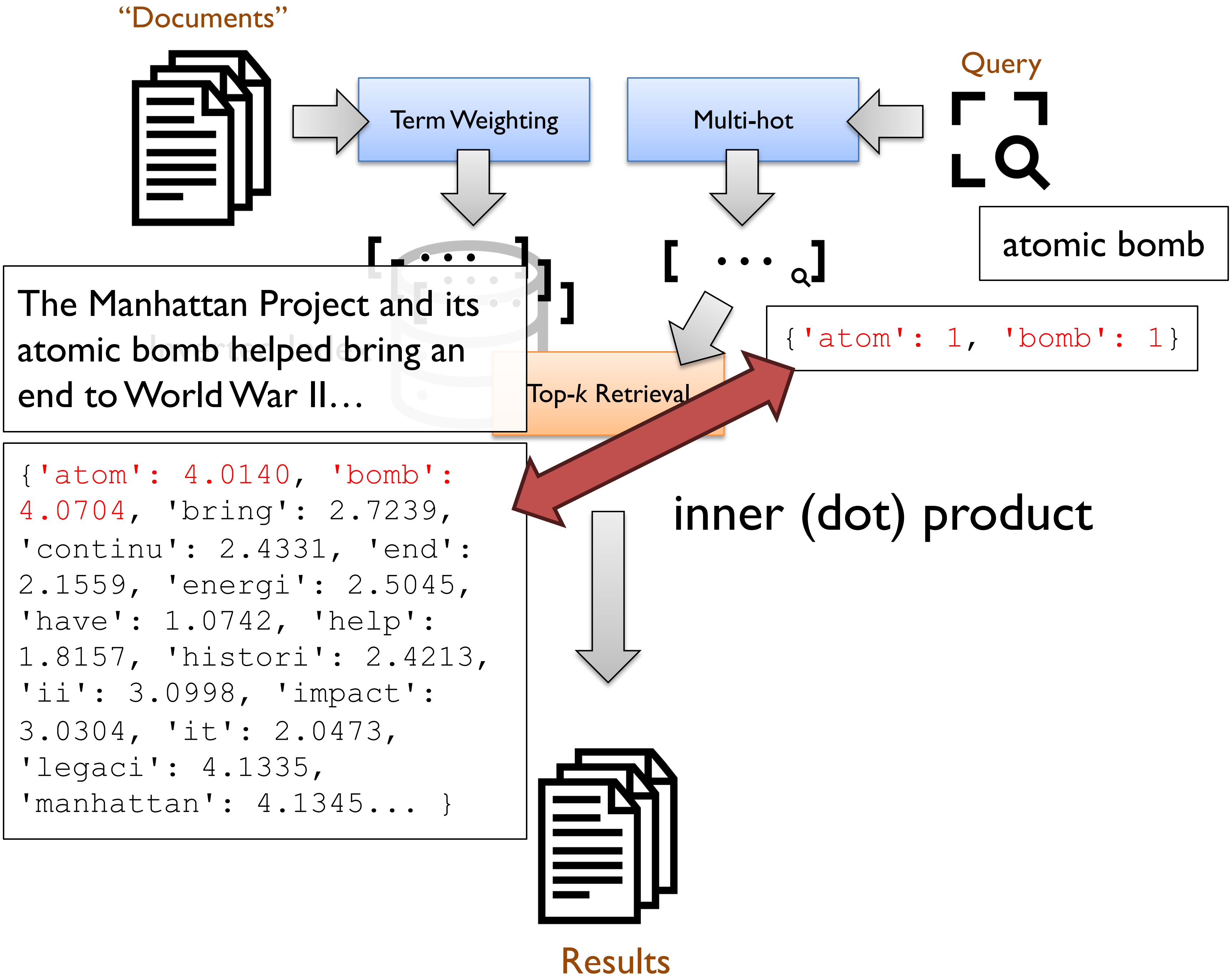


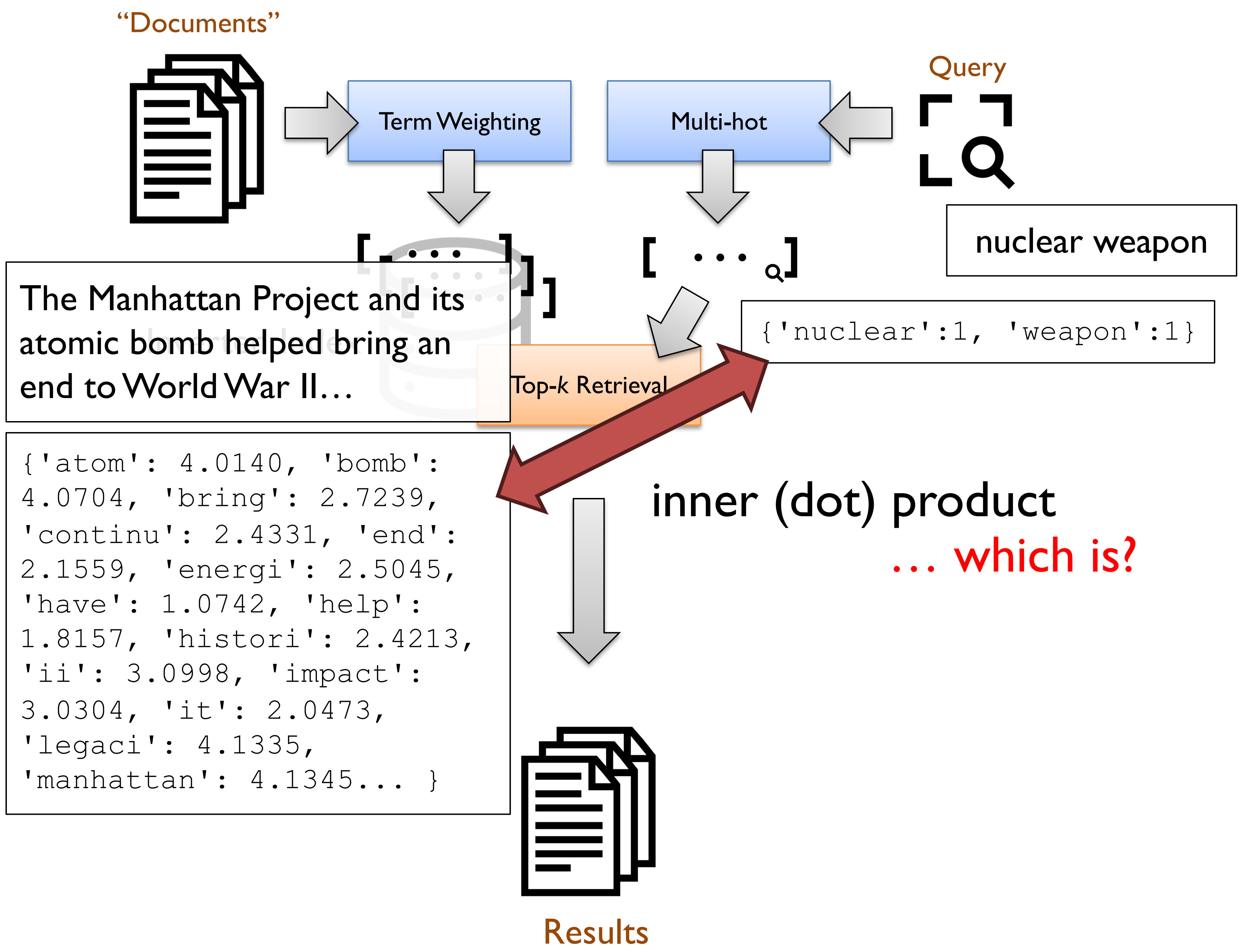
"Documents"
Term Weighting
Multi-hot
Query
atomic bomb
The Manhattan Project and its atomic bomb helped bring an end to World War II…
{'atom': 1, 'bomb': 1}
Top-k Retrieval
inner (dot) product
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu': 2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help': 1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it': 2.0473, 'legaci': 4.1335, 'manhattan': 4.1345... }
Results


"Documents"
Term Weighting
Multi-hot
Query
atomic bomb
The Manhattan Project and its atomic bomb helped bring an end to World War II…
{'atom': 1, 'bomb': 1}
Top-k Retrieval
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu': 2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help': 1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it': 2.0473, 'legaci': 4.1335, 'manhattan': 4.1345... }
inner (dot) product
Results



tl;dr – retrieval ~ computing dot products on vector representations!


"Documents"
Term Weighting
Multi-hot
Query
atomic bomb
The Manhattan Project and its atomic bomb helped bring an end to World War II…
{'atom': 1, 'bomb': 1}
Top-k Retrieval
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu': 2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help': 1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it': 2.0473, 'legaci': 4.1335, 'manhattan': 4.1345... }
inner (dot) product
Results

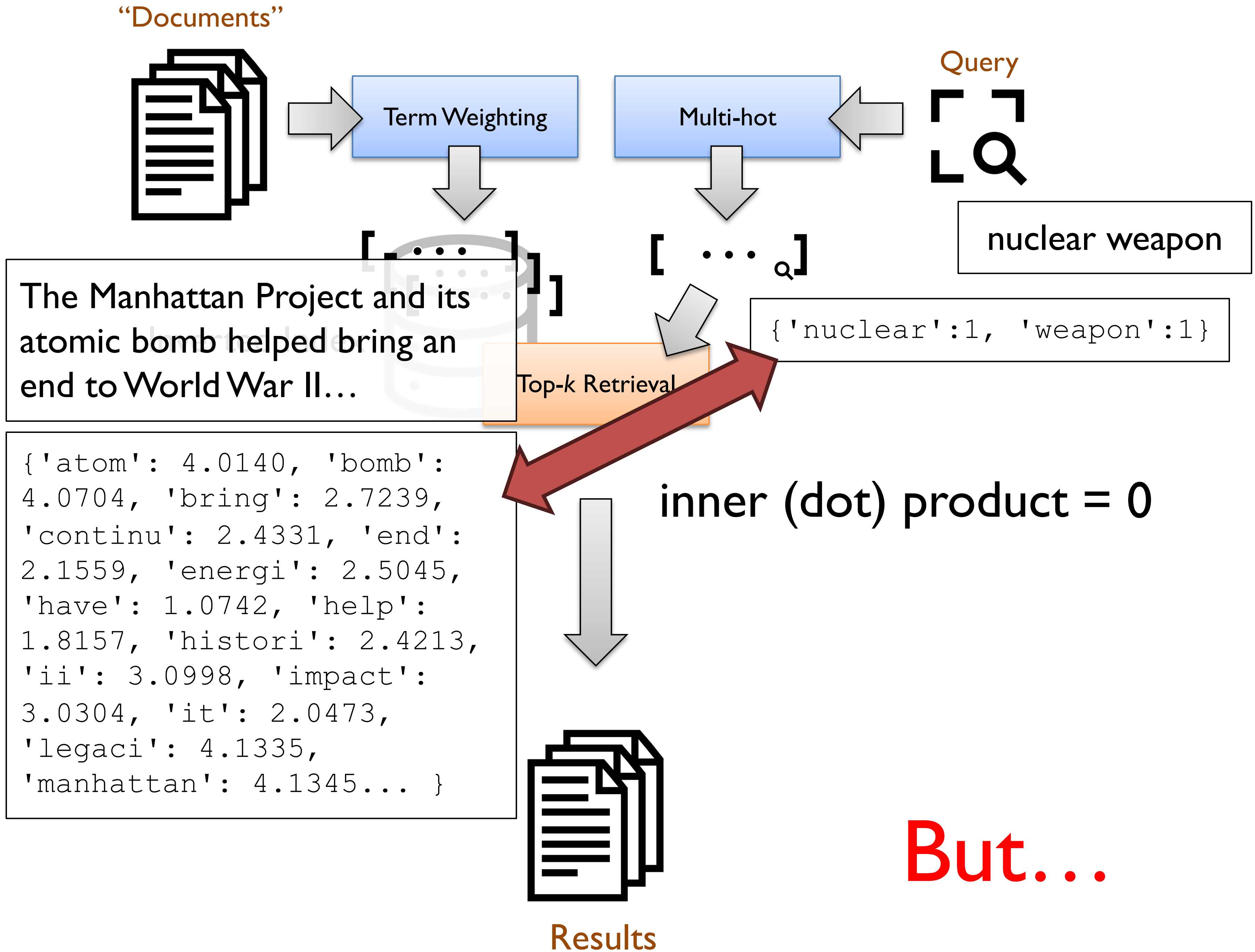


"Documents"
Term Weighting
Multi-hot
Query
nuclear weapon
The Manhattan Project and its atomic bomb helped bring an end to World War II…
{'nuclear':1, 'weapon':1}
Top-k Retrieval
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu': 2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help': 1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it': 2.0473, 'legaci': 4.1335, 'manhattan': 4.1345... }
inner (dot) product
… which is?
Results

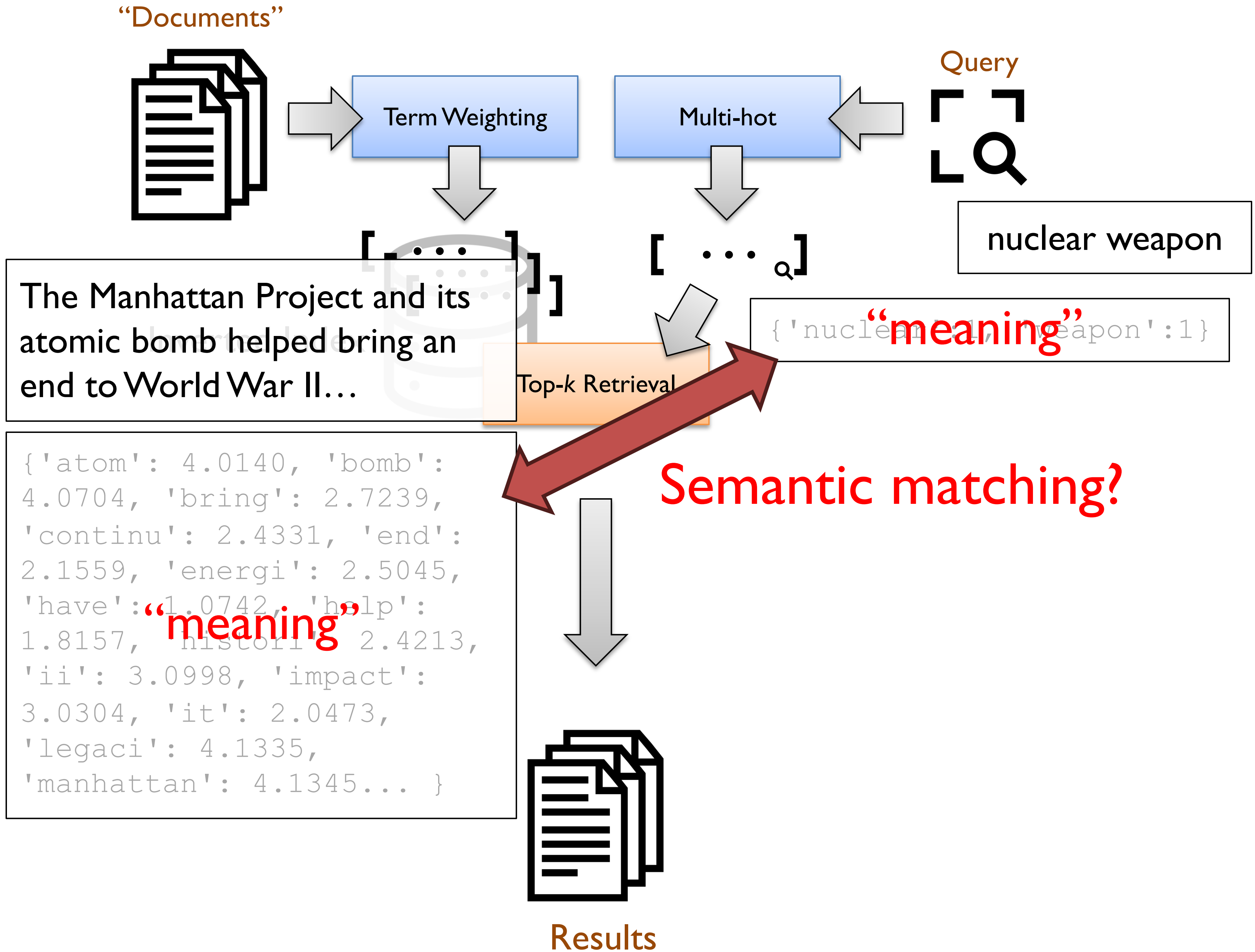
0

Semantic Mismatch

bag-of-words representations rely on lexical overlap



"Documents"
Term Weighting
Multi-hot
Query
nuclear weapon
The Manhattan Project and its atomic bomb helped bring an end to World War II…
{'nuclear':1, 'weapon':1}
Top-k Retrieval
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu': 2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help': 1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it': 2.0473, 'legaci': 4.1335, 'manhattan': 4.1345... }
inner (dot) product = 0
Results
But…

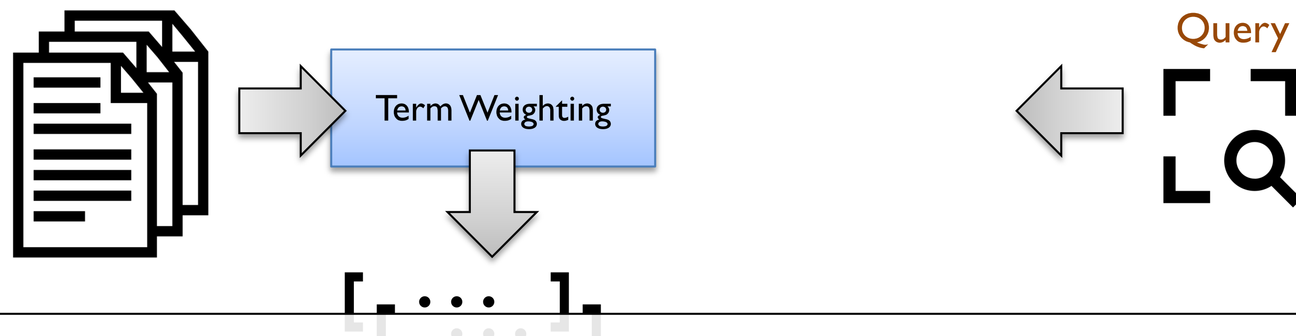


"Documents"
Term Weighting
Multi-hot
Query
nuclear weapon
The Manhattan Project and its atomic bomb helped bring an end to World War II…
Top-k Retrieval
"meaning"
Semantic matching?
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu': 2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help': 1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it': 2.0473, 'legaci': 4.1335, 'manhattan': 4.1345... }
"meaning"
Results

What if representations can capture “meaning”?

What type of representations?

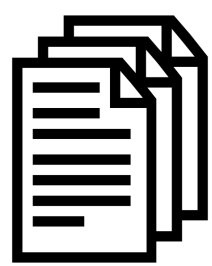
What does “meaning” mean?



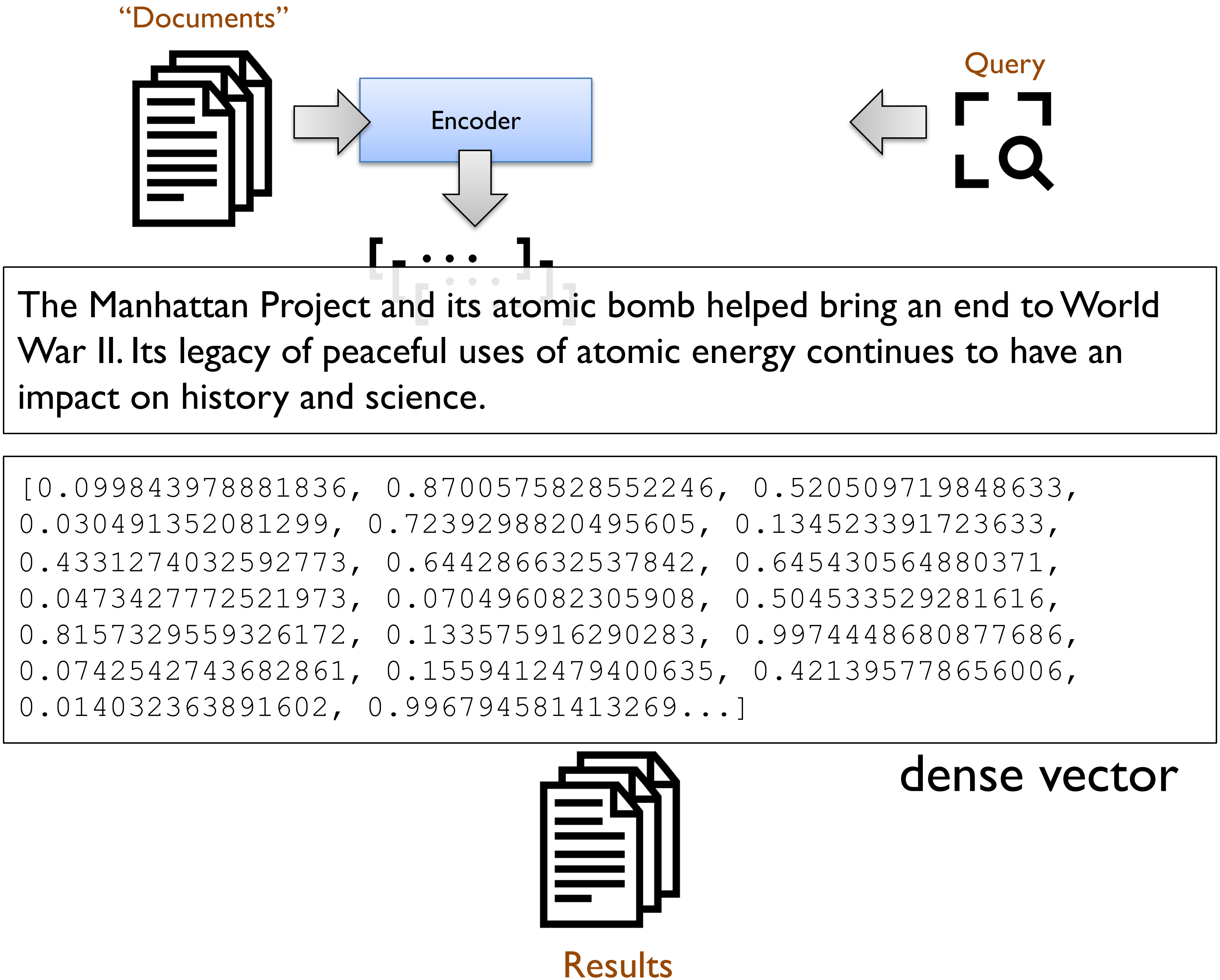
The Manhattan Project and its atomic bomb helped bring an end to World War II. Its legacy of peaceful uses of atomic energy continues to have an impact on history and science.

```
{'atom': 4.0140, 'bomb': 4.0704, 'bring': 2.7239, 'continu':
2.4331, 'end': 2.1559, 'energi': 2.5045, 'have': 1.0742, 'help':
1.8157, 'histori': 2.4213, 'ii': 3.0998, 'impact': 3.0304, 'it':
2.0473, 'legaci': 4.1335, 'manhattan': 4.1345, 'peac': 3.5205,
'project': 2.6442, 'scienc': 2.8700, 'us': 0.9967, 'war':
2.6454, 'world': 1.9974}
```

sparse vector



Results


"Documents"
Encoder
Query
The Manhattan Project and its atomic bomb helped bring an end to World War II. Its legacy of peaceful uses of atomic energy continues to have an impact on history and science.
[0.0998439788 81836, 0.8700575828552246, 0.520509719848633,
0.030491352081299, 0.7239298820495605, 0.134523391723633,
0.433127403259 2773, 0.644286632537842, 0.645430564880371,
0.047342777252 1973, 0.070496082305908, 0.504533529281616,
0.815732955932 6172, 0.133575916290283, 0.9974448680877686,
0.074254274368 2861, 0.1559412479400635, 0.421395778656006,
0.014032363891602, 0.996794581413269...]
dense vector
Results

What if representations can capture "meaning"?

What type of representations?

What does "meaning" mean?

What if…

vectors of queries and relevant passages get high inner products

vectors of queries and non-relevant passages get low inner products

And you have lots of examples?

This sounds an awful lot like…

A machine learning problem!

What if representations can capture "meaning"?

What type of representations?

What does "meaning" mean?

What if…

vectors of queries and relevant passages get high inner products

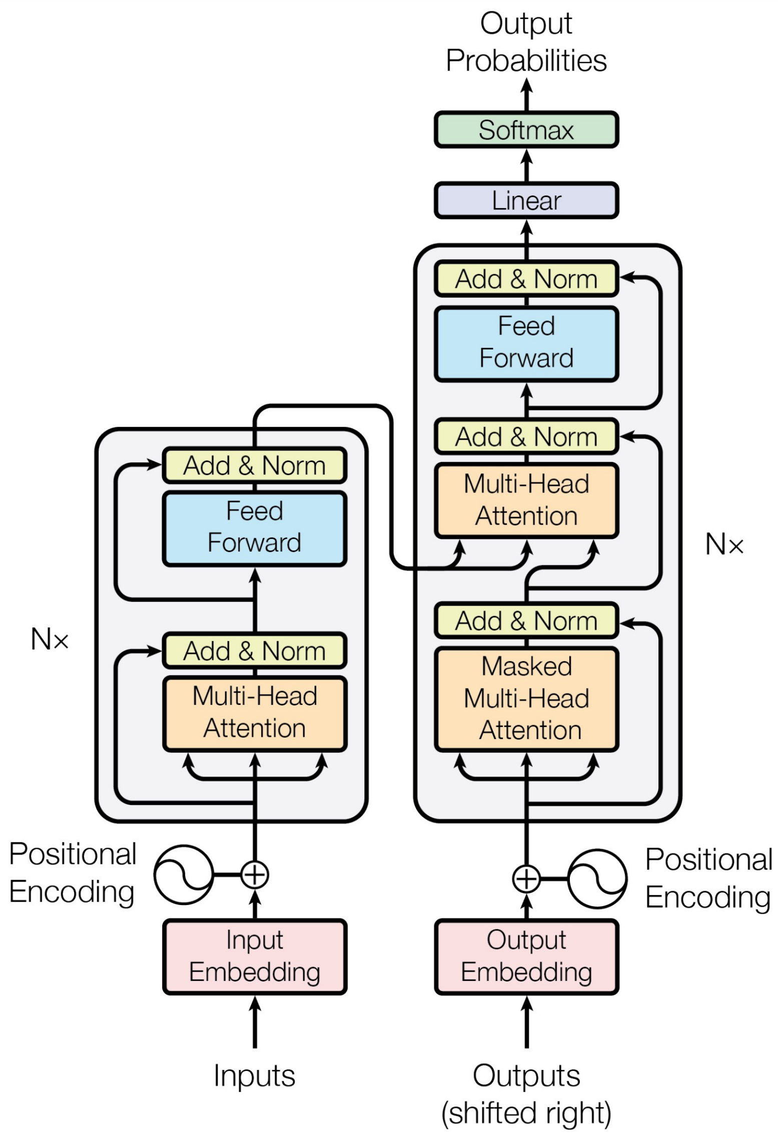
vectors of queries and non-relevant passages get low inner products

And you have lots of examples?

learned representations

= *embeddings*

But how?



Transformers!

Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

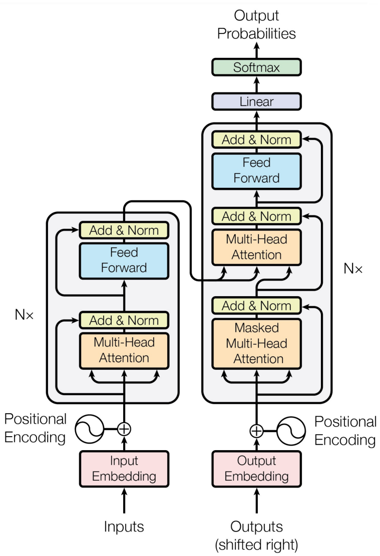
Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser*
Google Brain
lukaszkaiser@google.com

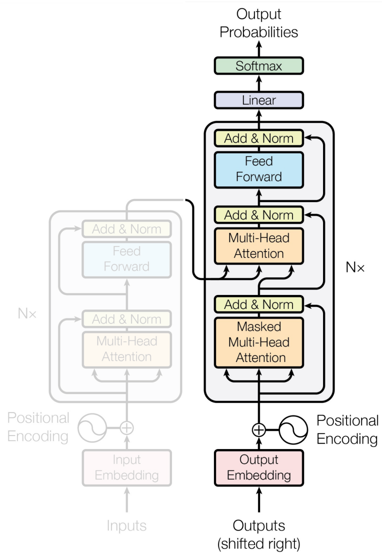
Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task,

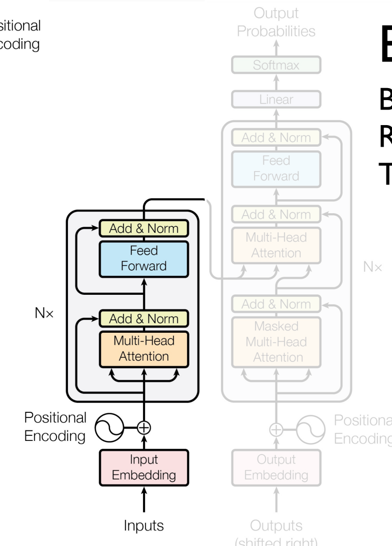


Transformer (2017)



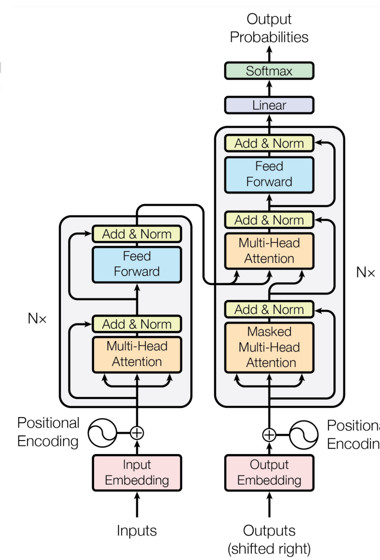
GPT (2018)

Generative Pretrained Transformer



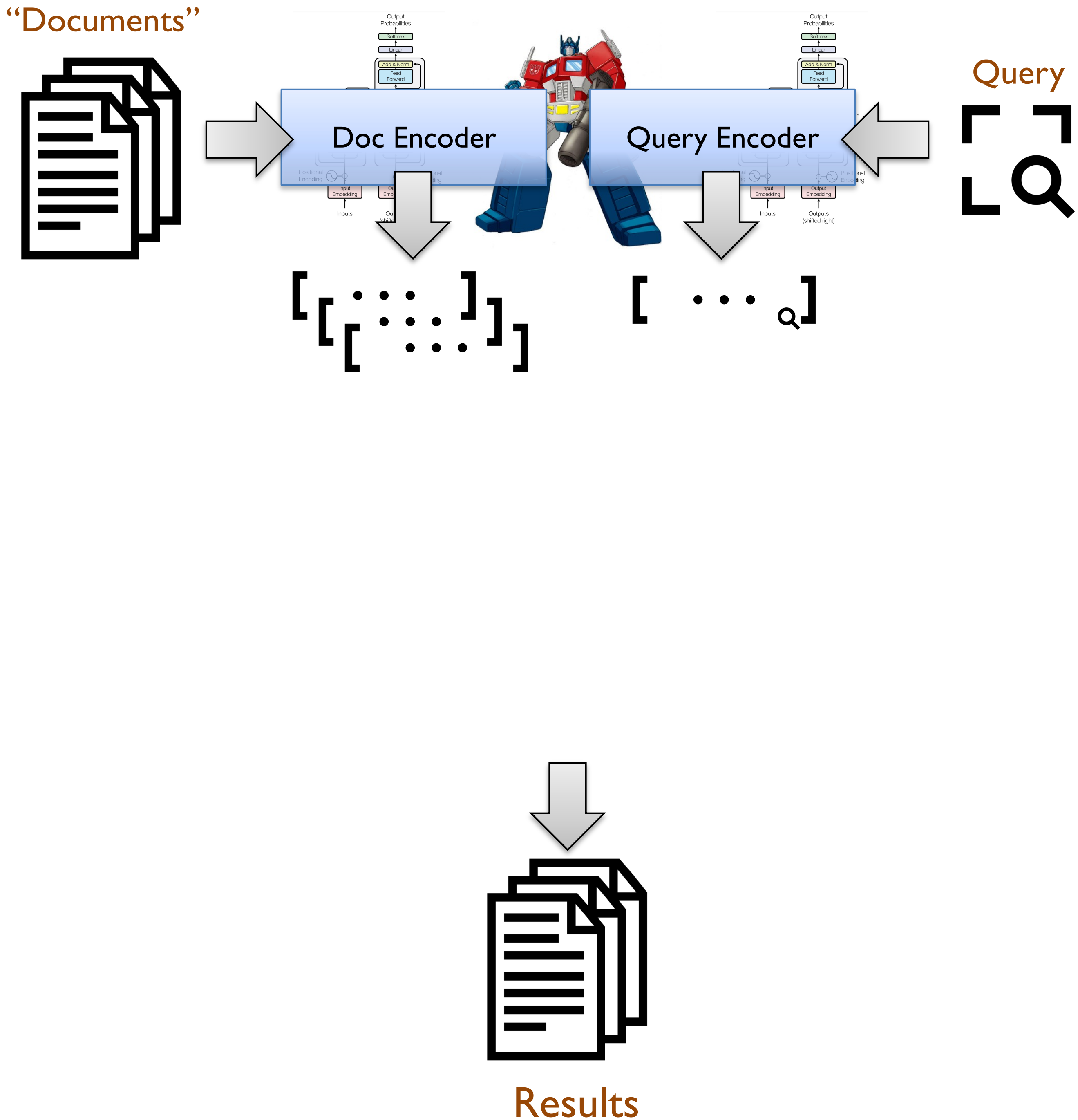
BERT (2018)

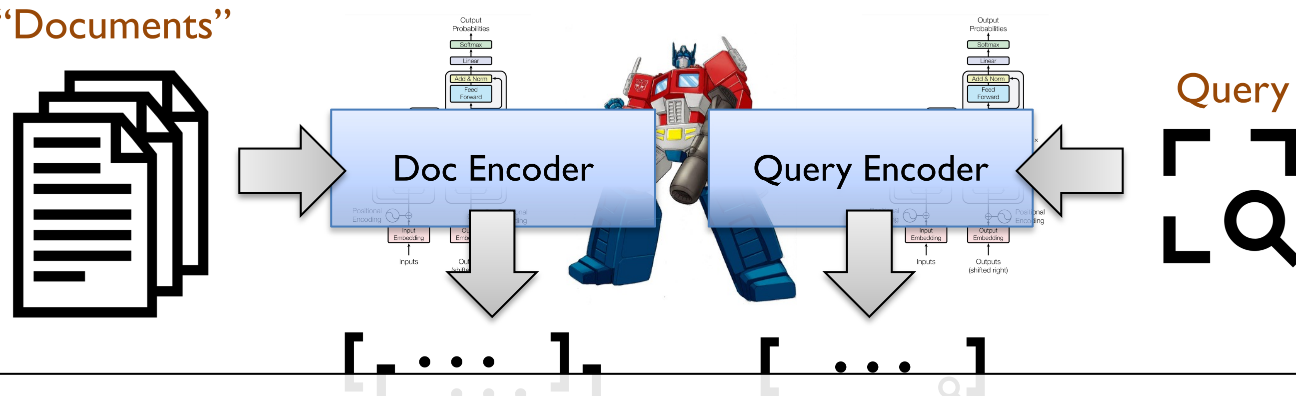
Bidirectional Encoder Representations from Transformers



T5 (2019)

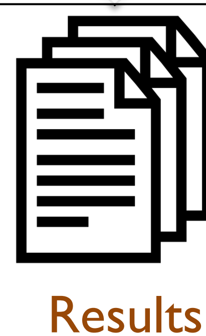
Text-To-Text Transfer Transformer

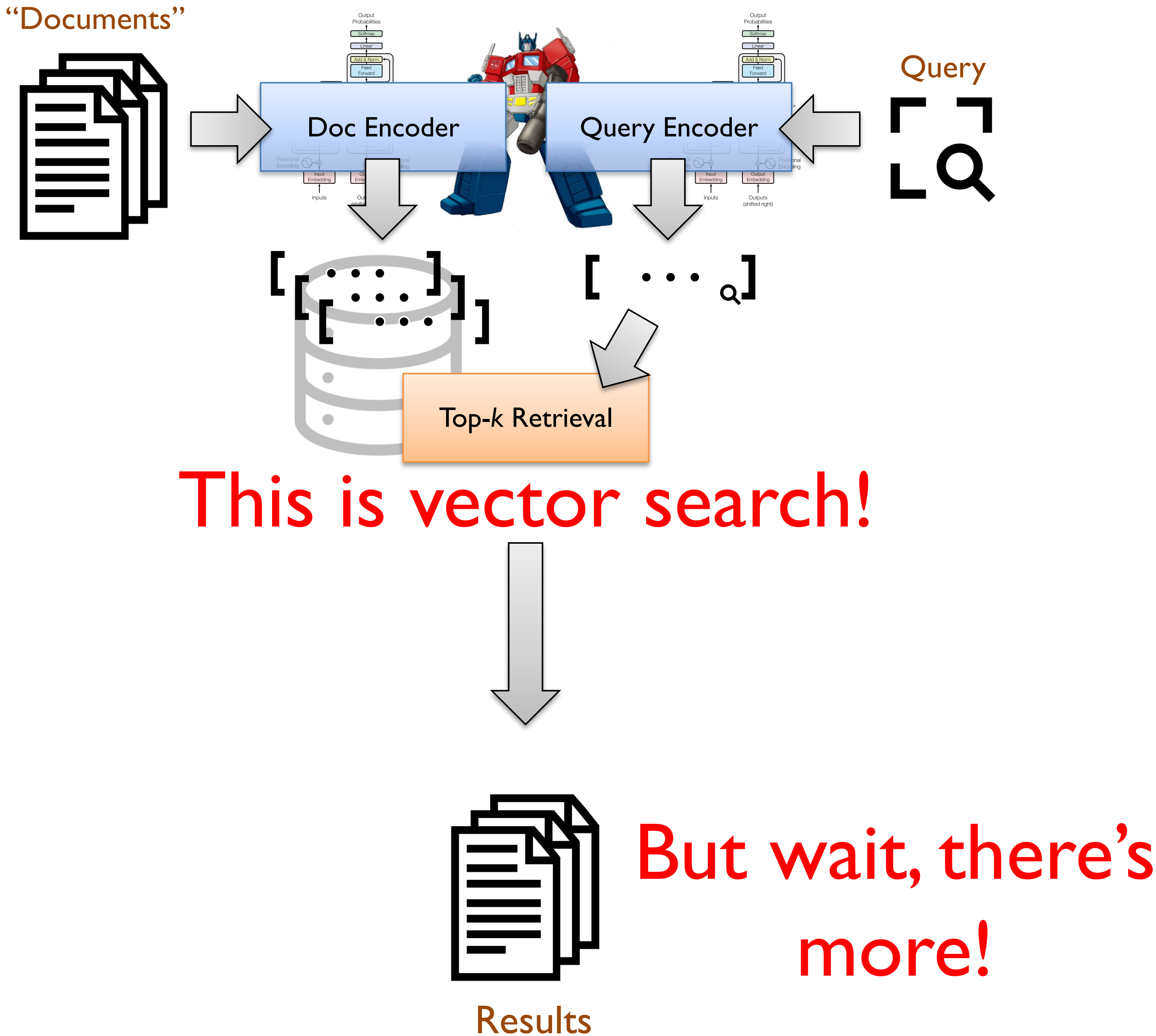

“Documents”
Doc Encoder
Query Encoder
Query
Results

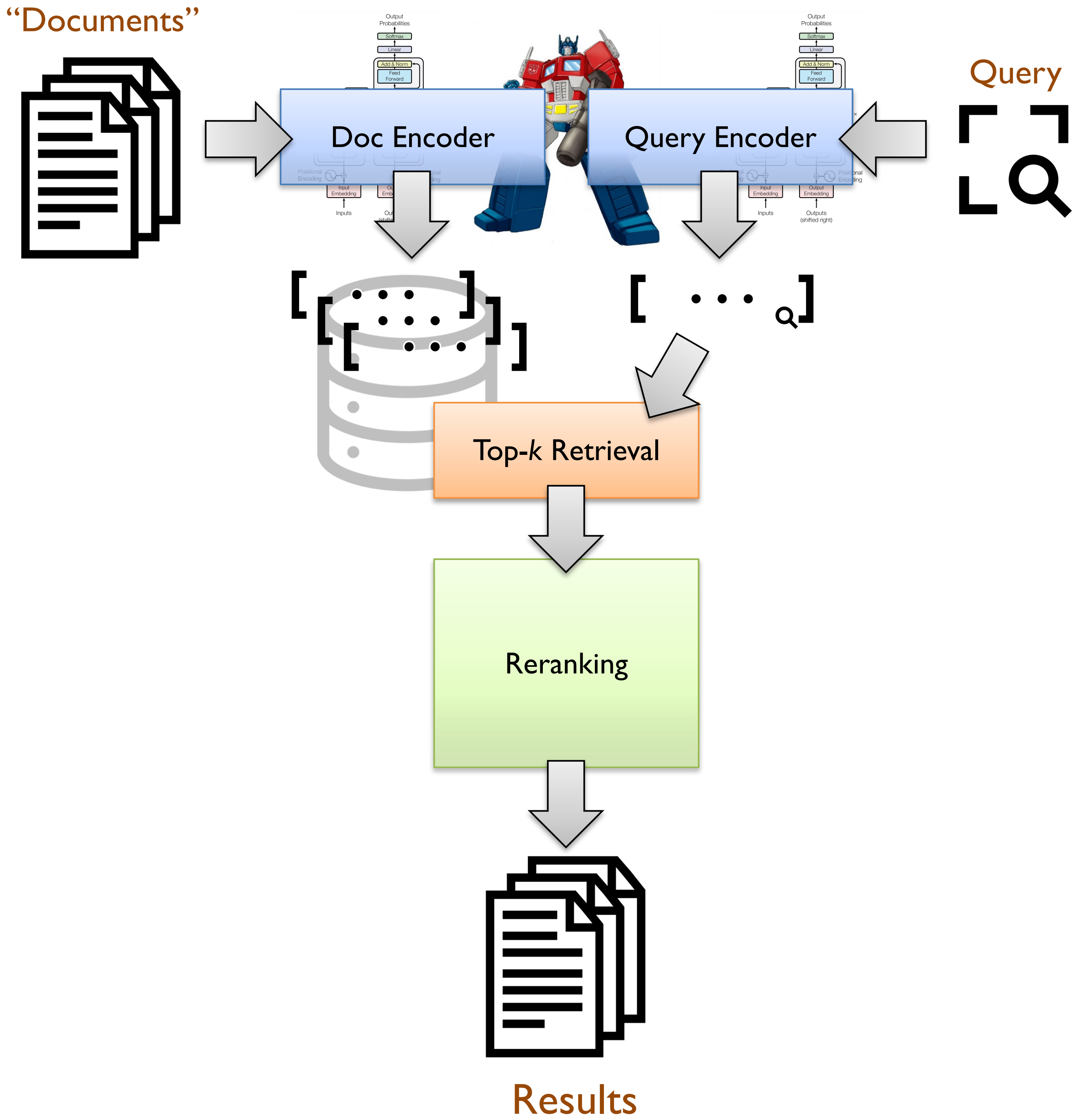


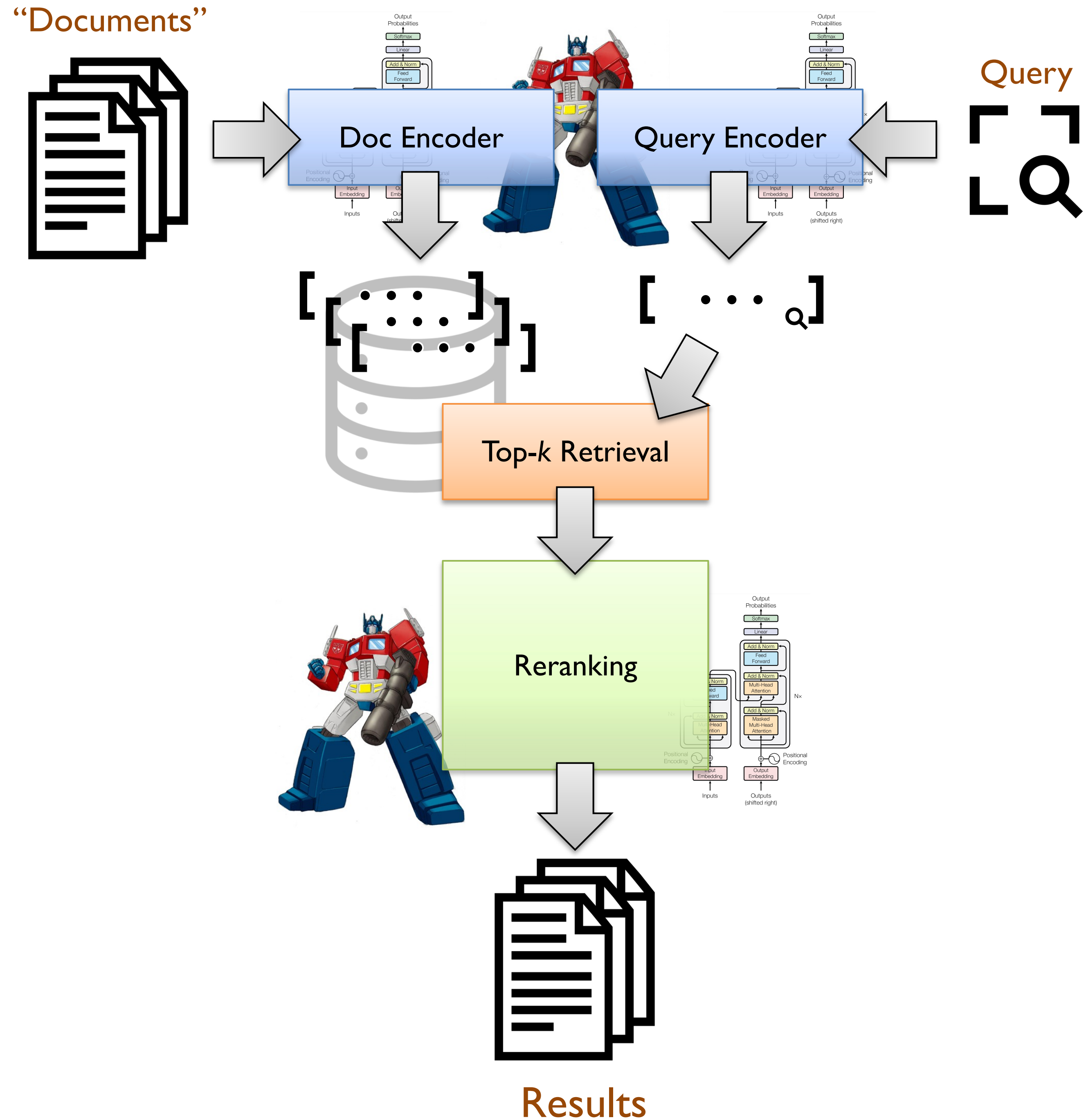
The Manhattan Project and its atomic bomb helped bring an end to World War II. Its legacy of peaceful uses of atomic energy continues to have an impact on history and science.

```
[0.0998439788881836, 0.8700575828552246, 0.5205097198486633,
0.030491352081299, 0.7239298820495605, 0.1345233917236633,
0.4331274032592773, 0.644286632537842, 0.645430564880371,
0.0473427772521973, 0.070496082305908, 0.5045335292816162,
0.8157329559326172, 0.133575916290283, 0.9974448680877686,
0.0742542743682861, 0.1559412479400635, 0.4213957786560006,
0.014032363891602, 0.996794581413269...]
```

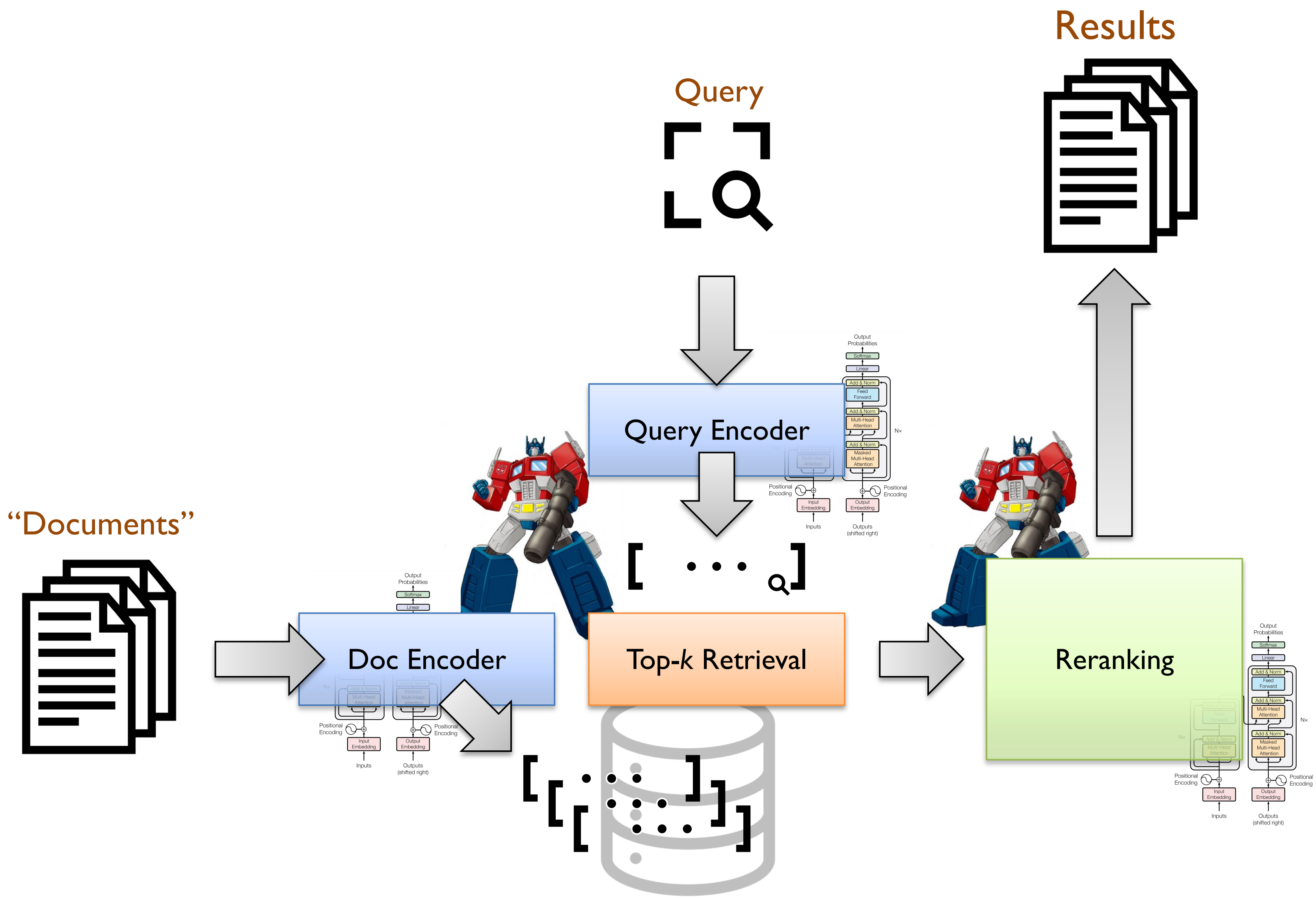



"Documents"
Doc Encoder
Query Encoder
Query
Top-k Retrieval
This is vector search!
Results
But wait, there's more!

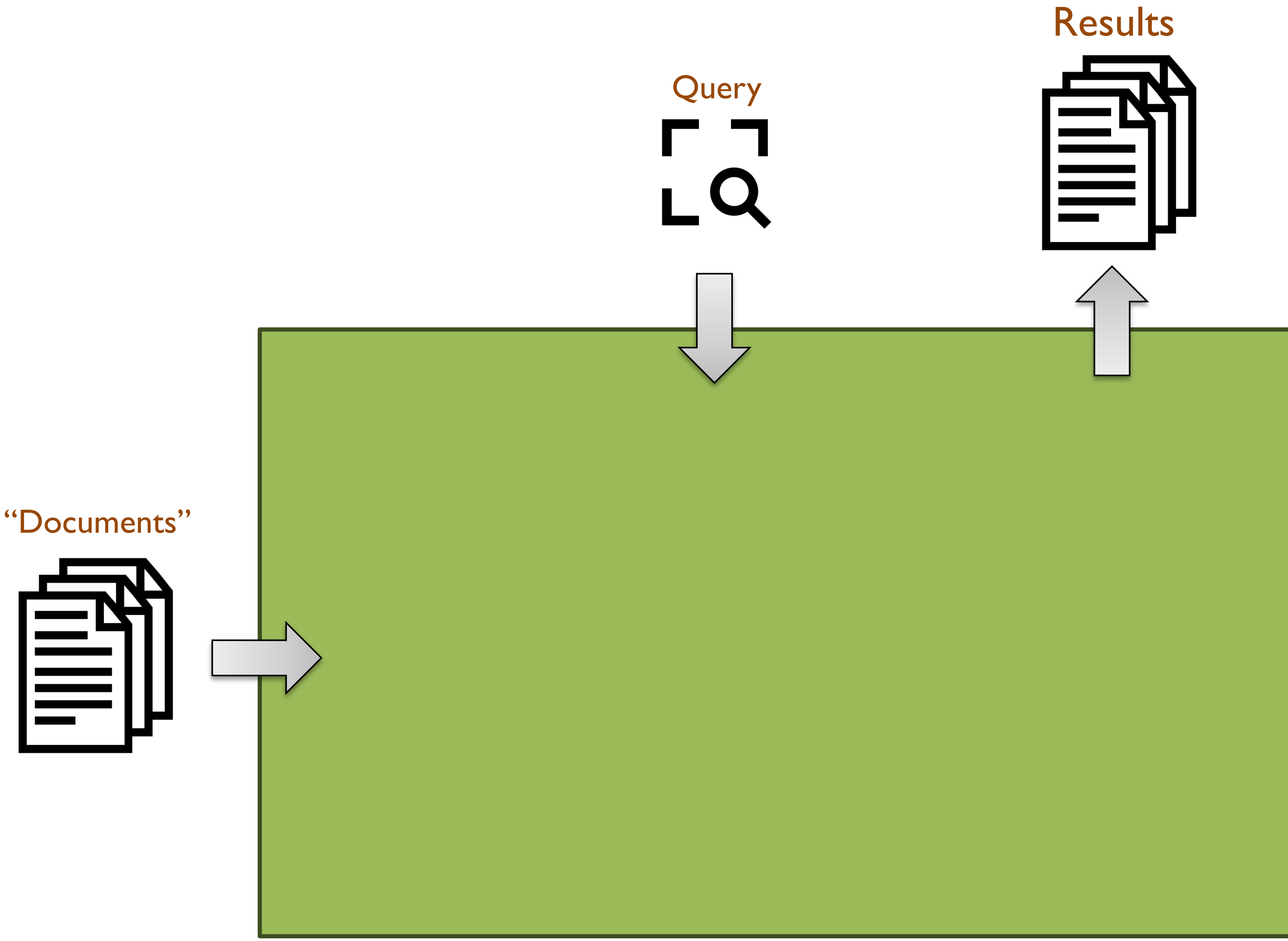

"Documents"
Doc Encoder
Query Encoder
Query
Top-k Retrieval
Reranking
Results



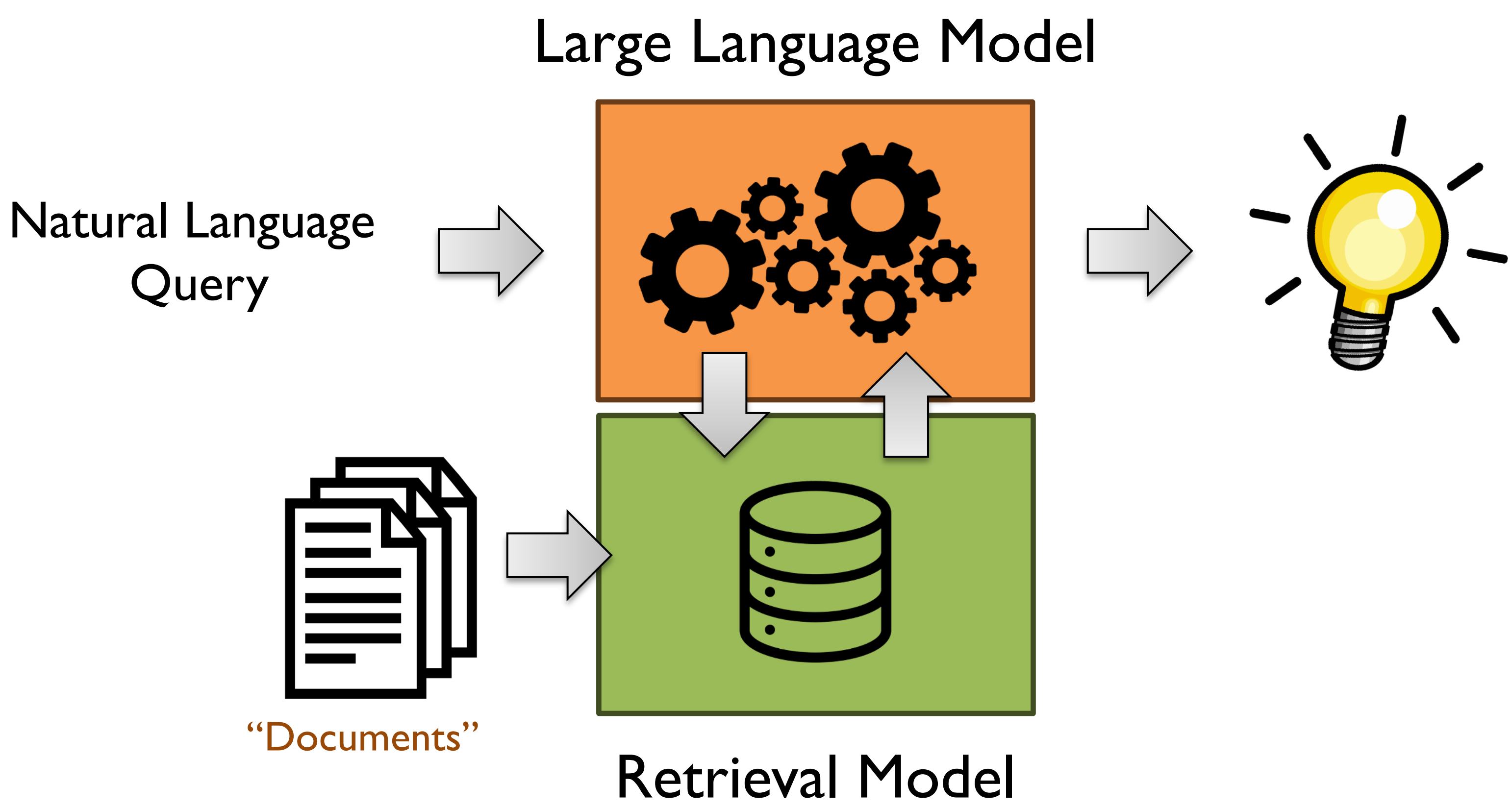
"Documents"
Query
Doc Encoder
Query Encoder
Top-k Retrieval
Reranking
Results



Results
Query
Query Encoder
"Documents"
Doc Encoder
Top-k Retrieval
Reranking

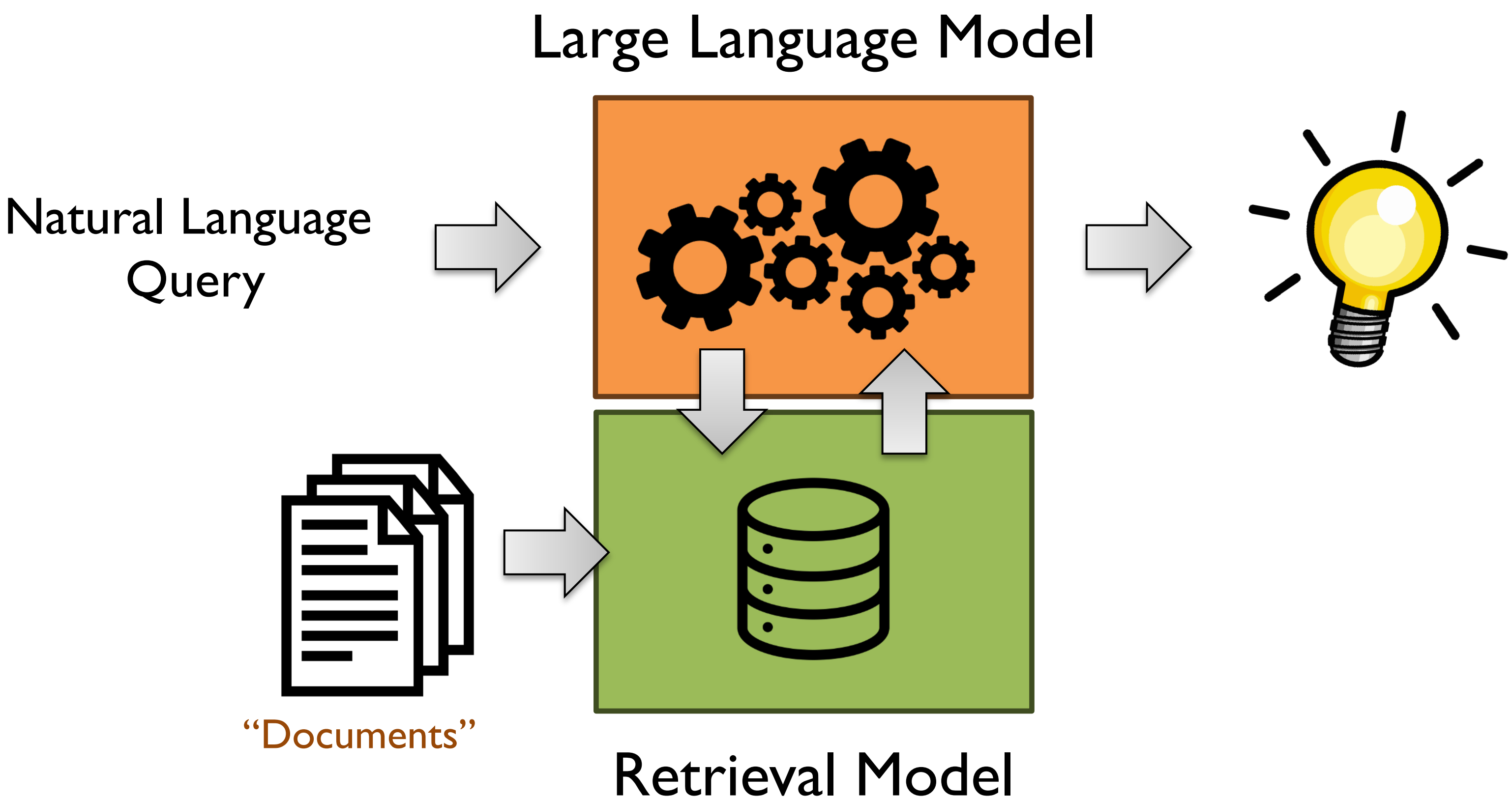

Results
Query
"Documents"

The Big Picture

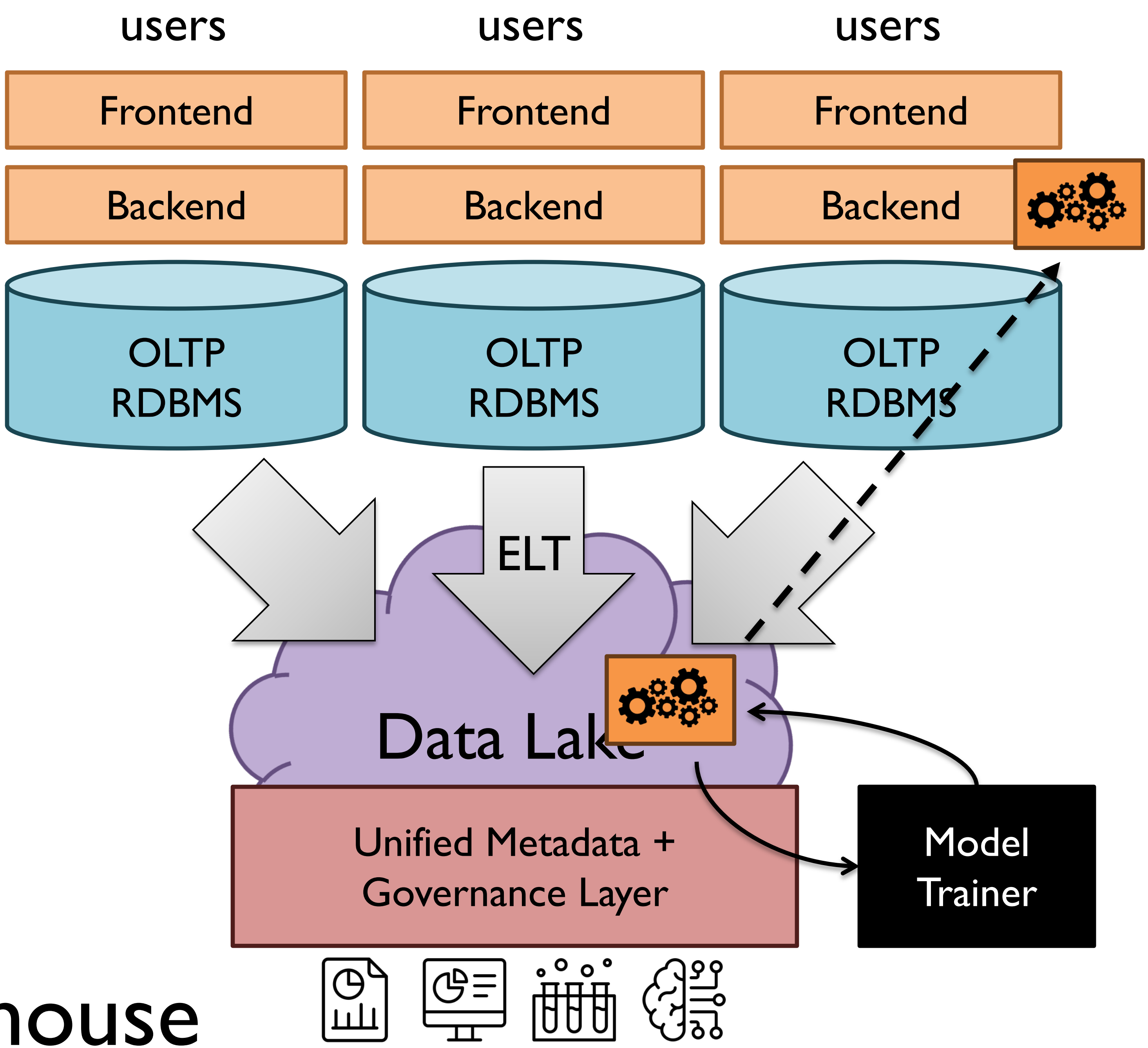


The Big Picture **Why?**

(combat) Hallucinations
(incorporate) Up-to-date information
(exploit) Private data



RAG / Lakehouse Integration: Why?


users
users
users
Frontend
Frontend
Frontend
Backend
Backend
Backend
OLTP
RDBMS
OLTP
RDBMS
OLTP
RDBMS
ELT
Data Lake
Unified Metadata +
Governance Layer
Model
Trainer

Lakehouse

RAG / Lakehouse Integration: Why?

Ingesting multiple sources of data

User-generated content + behavioral data

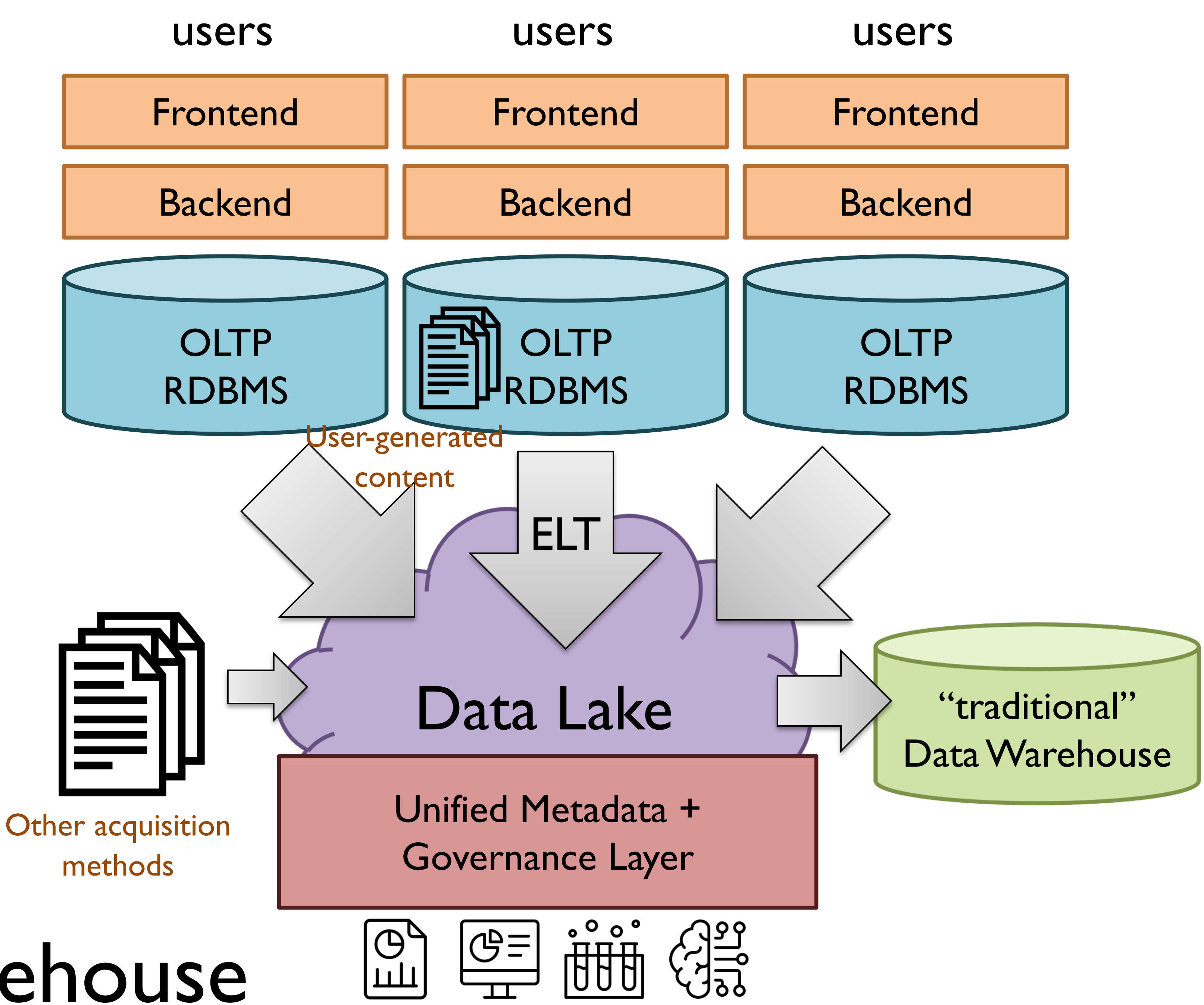
Other sources…

Performing "standard" lakehouse tasks

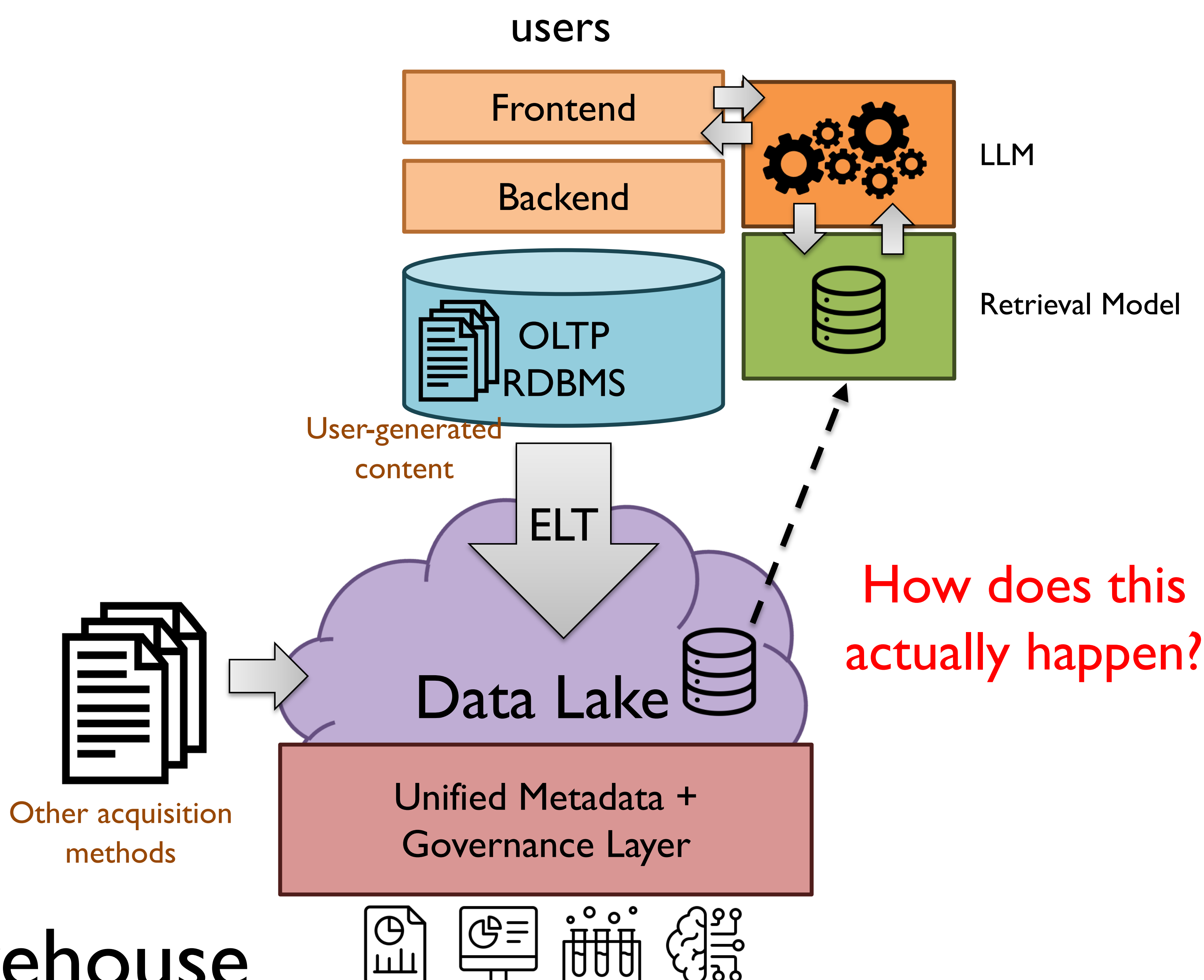
Joining, filtering, projecting, etc. heterogenous data

Data cleaning, aggregation, etc.

Training ML models


users
users
users
Frontend
Frontend
Frontend
Backend
Backend
Backend
OLTP RDBMS
OLTP RDBMS
OLTP RDBMS
User-generated content
ELT
Data Lake
Other acquisition methods
"traditional" Data Warehouse
Unified Metadata + Governance Layer

Lakehouse


users
Frontend
Backend
LLM
Retrieval Model
OLTP
RDBMS
User-generated
content
ELT
Data Lake
How does this
actually happen?
Other acquisition
methods
Unified Metadata +
Governance Layer

Lakehouse


冨嶽三十六景
神奈川沖
浪裏