Natural Language Processing - Search

Recap

Relevant concepts

- Corpus
- Vocabulary
- Bag of words
- Similarity Measures
- Indexing
- TF-IDF
- Okapi BM25

Natural Language Processing - Search

How to build a frequency-based search engine ?

Natural Language Processing - Search

How to build a frequency-based search engine ?

1. Define the corpus
   a. Define what is the ‘document’
   b. Pre-process data

Natural Language Processing - Search

How to build a frequency-based search engine ?

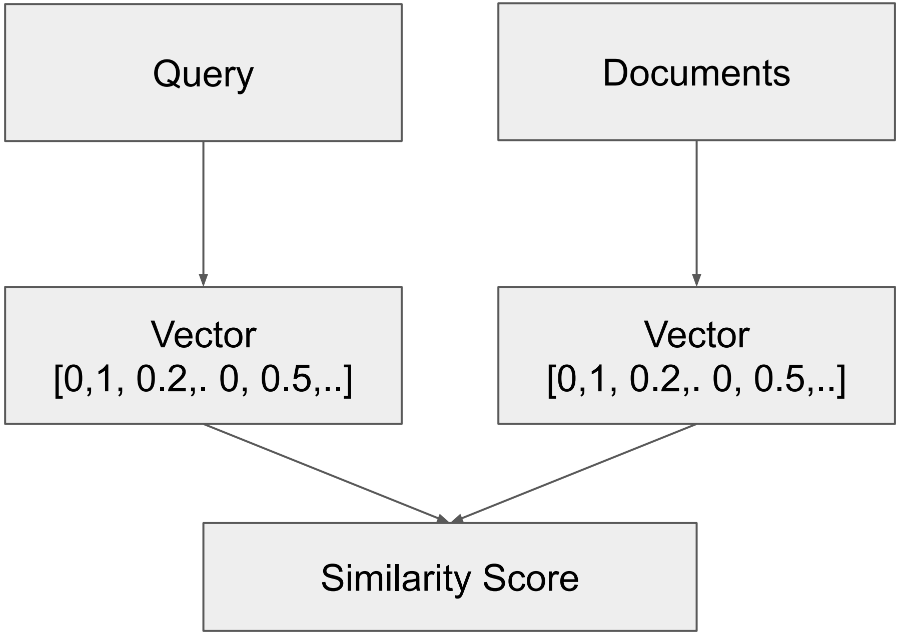
1. Define the corpus
2. Decide on retrieval algorithm
   a. Bag-of-words + Cosine Similarity
   b. TF-IDF + Cosine Similarity
   c. BM25

Computers do not understand text.
They only understand numbers.

Natural Language Processing - Search

How to build a frequency-based search engine ?

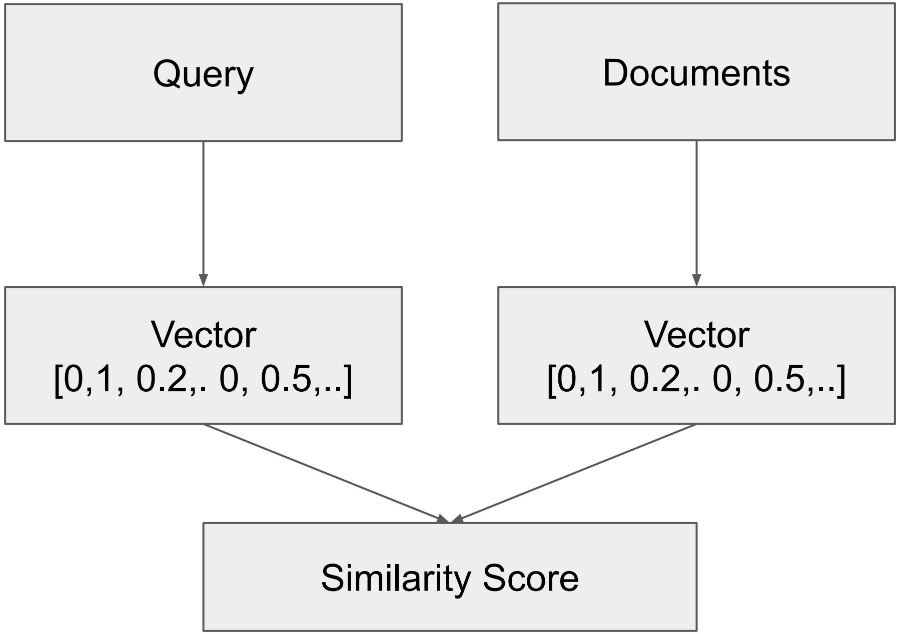
1. Define the corpus
2. Decide on retrieval algorithm
   a. **Bag-of-words + Cosine Similarity**
   b. TF-IDF + Cosine Similarity
   c. BM25



Natural Language Processing - Search

How to build a frequency-based search engine ?

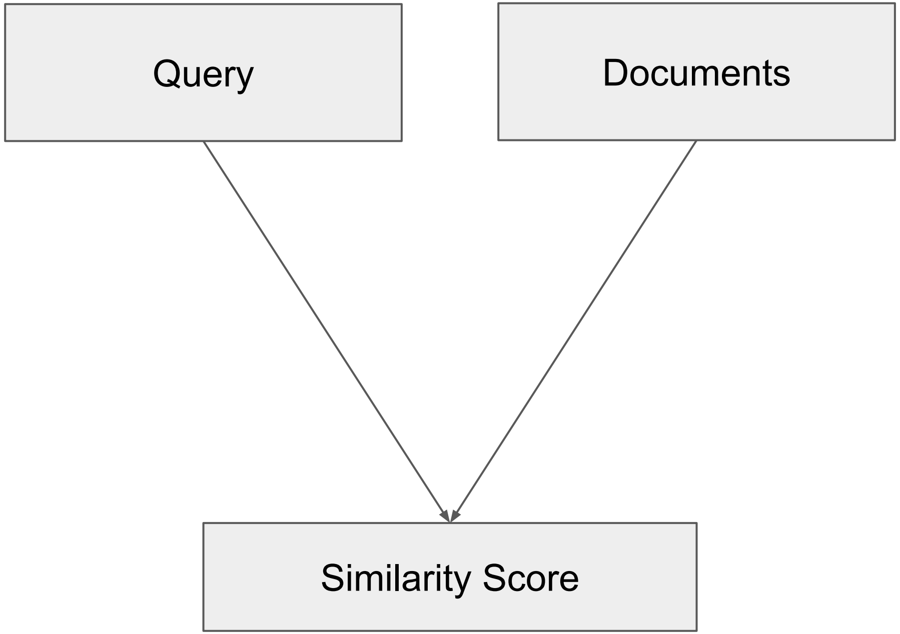
1. Define the corpus
2. Decide on retrieval algorithm
   a. Bag-of-words + Cosine Similarity
   b. TF-IDF + Cosine Similarity
   c. BM25



Natural Language Processing - Search

How to build a frequency-based search engine ?

1. Define the corpus
2. Decide on retrieval algorithm
   a. Bag-of-words + Cosine Similarity
   b. TF-IDF + Cosine Similarity
   c. BM25



Natural Language Processing - Search

How to build a frequency-based search engine ?

1. Define the corpus
2. Decide on retrieval algorithm
3. Index corpus
   a. Compute count of words, per corpus and per document
   b. Create auxiliary vectors

Natural Language Processing - Search

How to build a frequency-based search engine ?

1. Define the corpus
2. Decide on retrieval algorithm
3. Index corpus
4. That’s it

Let's compare them.

How much is pre-processing important ?

How does BoW, TF-IDF and BM25 compare against each other?

Natural Language Processing - Search

Let’s compare them.

How much is pre-processing important ?

How does BoW, TF-IDF and BM25 compare against each other?

How can we formally compare them ?

Evaluation Metrics

- Precision (at K)
- Recall (at K)
- F1 Score
- NDCG

Evaluation Metrics

- Precision (at K)

Natural Language Processing - Search

Evaluation Metrics

- Precision (at K)

Precision = TP / (TP + FP)

True Positives
i.e The system classifies something as true and they are true

False Positives
i.e The system classifies something as true and they are false

Evaluation Metrics

- Precision (at K)

Precision = TP / (TP + FP)

All items

True Positives
i.e The system classifies them as true and they are true

False Positives
i.e The system classifies them as true and they are false

Evaluation Metrics

- Precision (at K)

Precision = TP / (TP + FP)

% of times the system got the true label right

Evaluation Metrics

- Precision (at K) is the proportion of recommended items in the top-k set that are relevant

Precision at K = TP at K / (TP at K + FP at K)

Evaluation Metrics

- Precision (at K) is the proportion of recommended items in the top-k set that are relevant

Precision at K = TP at K / (TP at K + FP at K)

Result A	Relevant
Result B	Relevant
Result C	Not Relevant

Natural Language Processing - Search

Evaluation Metrics

- Precision (at K) is the proportion of recommended items in the top-k set that are relevant

↓

Precision at K = TP at K / (TP at K + FP at K)   Precision at 3 = ⅔ = 66%

Result A	Relevant
Result B	Relevant
Result C	Not Relevant

Evaluation Metrics

- Precision (at K) is the proportion of recommended items in the top-k set that are relevant

Precision at K = TP at K / (TP at K + FP at K)

Precision at 3 = ⅔ = 66%

Result A	Relevant
Result B	Not Relevant
Result C	Relevant

Evaluation Metrics

- Recall (at K)

Recall = TP/(TP+FN)

False Negatives
i.e The system classifies something as false and they are true

Evaluation Metrics

- Recall (at K) (also known as sensitivity) is the fraction of relevant instances that were retrieved.

Recall at K= TP at K / (TP at K + FN at K)
Relevant items = 5

Recall at 3 = 2/2 = 100%

Result A	Relevant
Result B	Not Relevant
Result C	Relevant

Evaluation Metrics

- Recall (at K) (also known as sensitivity) is the fraction of relevant instances that were retrieved.

Recall at K= TP at K / (All relevant items)
Relevant items = 5

Recall at 3 = ⅖ = 40%

Result A	Relevant
Result B	Not Relevant
Result C	Relevant

Evaluation Metrics

- Recall (at K) (also known as sensitivity) is the **fraction of relevant instances that were retrieved.**

Recall at K= TP at K / (All relevant items)
Relevant items = 5

Recall at 3 = ⅖ = 40%

Result A	Relevant
Result B	Not Relevant
Result C	Relevant

Evaluation Metrics

- F1 Score is a metric that takes into account both precision and recall to provide a balanced evaluation of a system's performance

Evaluation Metrics

- F1 Score is a metric that takes into account both precision and recall to provide a balanced evaluation of a system's performance

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)

Evaluation Metrics

- F1 Score is a metric that takes into account both precision and recall to provide a balanced evaluation of a system's performance

F1 Score = 2 * (Precision * Recall) / (Precision + Recall)

… gives more weight to the lower of the two values. This means that if **either precision** or **recall is low** (i.e., the weaker of the two metrics), **the harmonic mean will also be low**, *reflecting the fact that the system is not performing well in at least one of these aspects*.
It penalizes systems that have an extreme imbalance between precision and recall.

Evaluation Metrics

- F1 Score at K is a metric that takes into account both precision and recall to provide a balanced evaluation of a system's performance

F1 Score at K = 2 * (Precision at K * Recall at K) / (Precision at K + Recall at K)

… gives more weight to the lower of the two values. This means that if **either precision** or **recall is low** (i.e., the weaker of the two metrics), **the harmonic mean will also be low**, *reflecting the fact that the system is not performing well in at least one of these aspects*.
It penalizes systems that have an extreme imbalance between precision and recall.

Evaluation Metrics

- Normalized Discounted Cumulative Gain (NDCG)

Evaluation Metrics

- Normalized Discounted Cumulative Gain (NDCG) assesses how well the top-ranked items in a list align with the preferences or relevance judgments of users, i.e order matters.

Evaluation Metrics

- Normalized **Discounted Cumulative Gain** (N**DCG**).

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} =$$

Evaluation Metrics

- Normalized **Discounted Cumulative Gain** (N**DCG**).

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} =$$

DCGp is the DCG at position p.

Evaluation Metrics

- Normalized **Discounted Cumulative Gain** (N**DCG**).

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)}$$

  - DCG_p is the DCG at position p.
  - rel_i is the relevance score of the item at position i in the ranking list (typically a non-negative number, where higher values represent higher relevance).

Evaluation Metrics

- Normalized **Discounted Cumulative Gain** (N**DCG**).

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)}$$

- $\mathrm{DCG_p}$ is the DCG at position p.
- rel_i is the relevance score of the item at position i in the ranking list (typically a non-negative number, where higher values represent higher relevance).
- Log is used to produce a smooth reduction

Evaluation Metrics

- Normalized **Discounted Cumulative Gain** (N**DCG**).

  The premise of DCG is that **highly relevant documents appearing lower in a search result list should be penalized** as the graded relevance value is reduced logarithmically proportional to the position of the result.

Evaluation Metrics

- **Normalized** Discounted Cumulative Gain (**N**DCG).

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p}$$

$$\mathrm{IDCG_p} = \sum_{i=1}^{|REL_p|} \frac{rel_i}{\log_2(i+1)}$$

represents the list of relevant documents (ordered by their relevance) in the corpus up to position p.

ideal discounted cumulative gain

Let's compare them.

How much is pre-processing important ?

How does BoW, TF-IDF and BM25 compare against each other?

How can we formally compare them ?

- Precision at K
- Recall at K
- F1 Score at K
- NDCG

Let's compare them.

How much is pre-processing important ?

How does BoW, TF-IDF and BM25 compare against each other?

How can we formally compare them ?

- Precision at K
- Recall at K
- F1 Score at K
- NDCG

Semantic Search

… is a search technique that focuses on understanding the meaning and context of user queries to provide more relevant search results.

Natural Language Processing - Search

Semantic Search

… is a search technique that focuses on understanding the meaning and context of user queries to provide more relevant search results.

- Goes beyond keyword matching.
- Utilizes Natural Language Understanding (NLU).
- Aims for contextual accuracy.

Semantic Search

… is a search technique that focuses on understanding the meaning and context of user queries to provide more relevant search results.

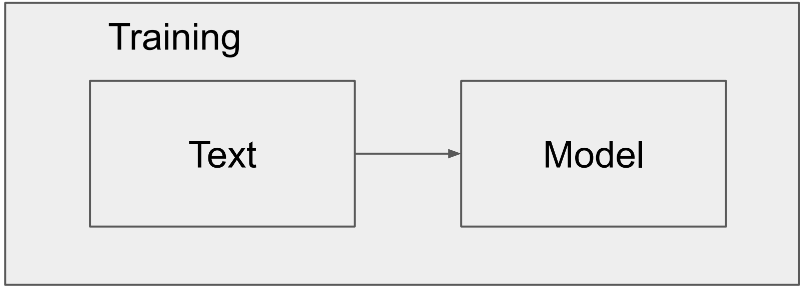
- **Aims for contextual accuracy.**

Semantic Search

Word Embeddings (dense vectors)

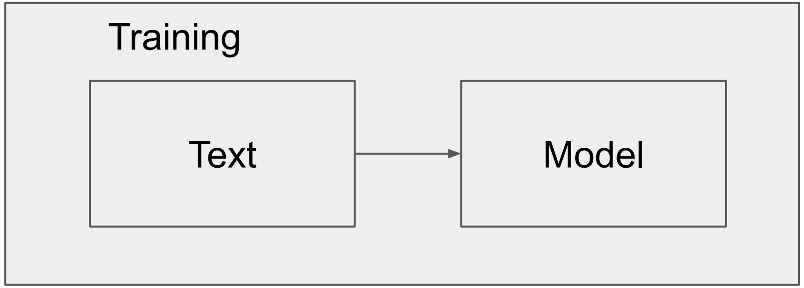
Semantic Search

Word Embeddings



Semantic Search

Word Embeddings

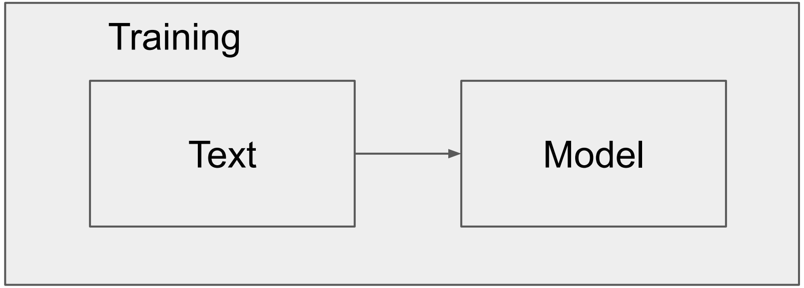


- Word2Vec
- GloVe
- CBOW
- Skipgram
- …
- Transformers

Natural Language Processing - Search

Semantic Search

Word Embeddings



- Predict context i.e words around another
- Predict next word
  (next token prediction)
- Predict missing word
  (masked token token prediction)

Natural Language Processing - Search

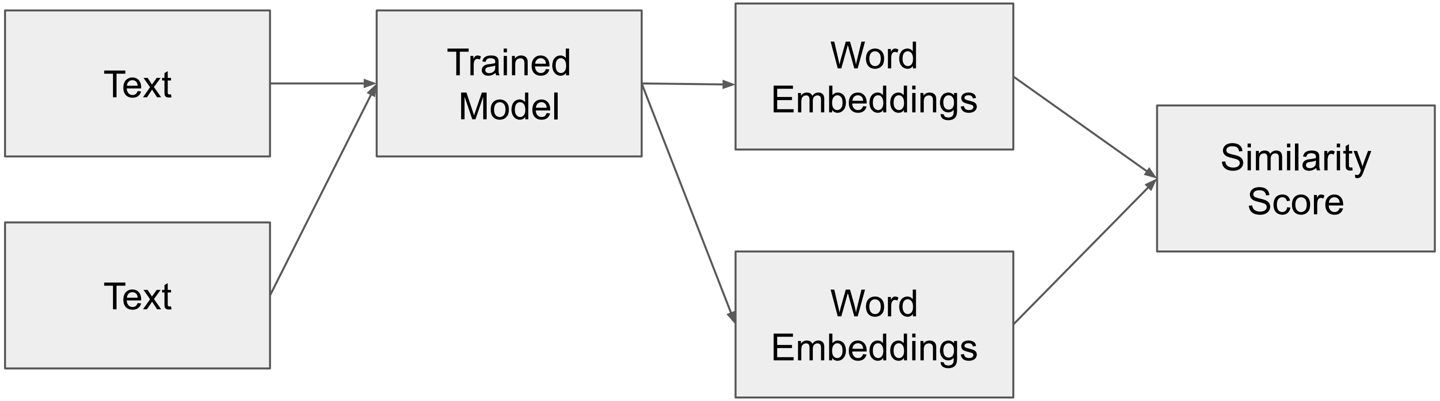
Semantic Search

Word Embeddings

Natural Language Processing - Search

Semantic Search

Word Embeddings



Natural Language Processing - Search

Semantic Search

Key players

spaCy Hugging Face

OpenAI

… and much more

Natural Language Processing - Search

Semantic Search

Key players

spaCy

Hugging Face

OpenAI

… and much more

Natural Language Processing - Search

Semantic Search

+

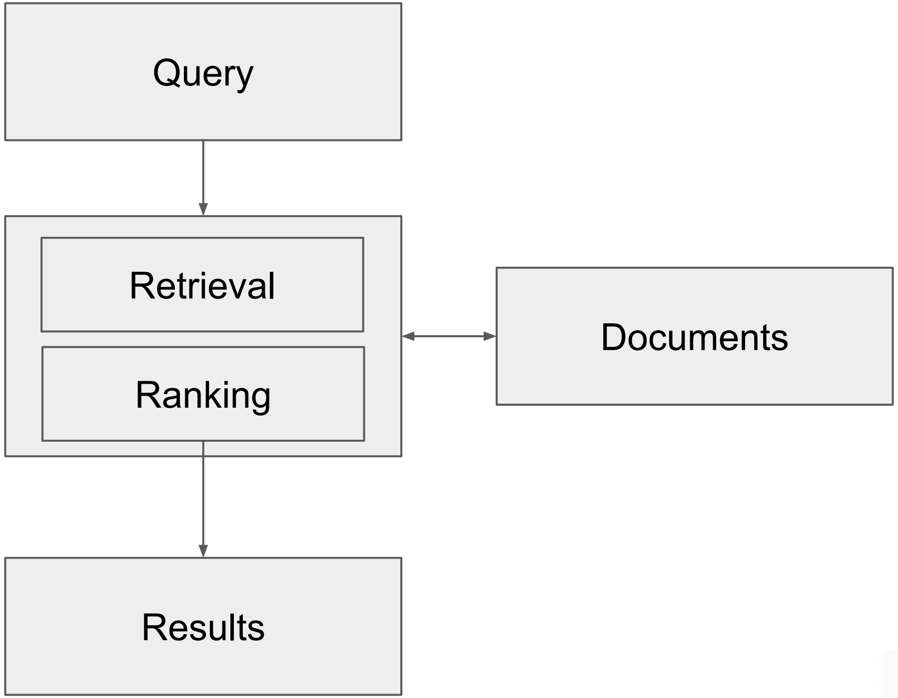
Generative AI

=

Retrieval Augmented Generation (RAG)

Natural Language Processing - Search

Semantic Search
+
Generative AI
=
Retrieval Augmented Generation (RAG)



Natural Language Processing - Search

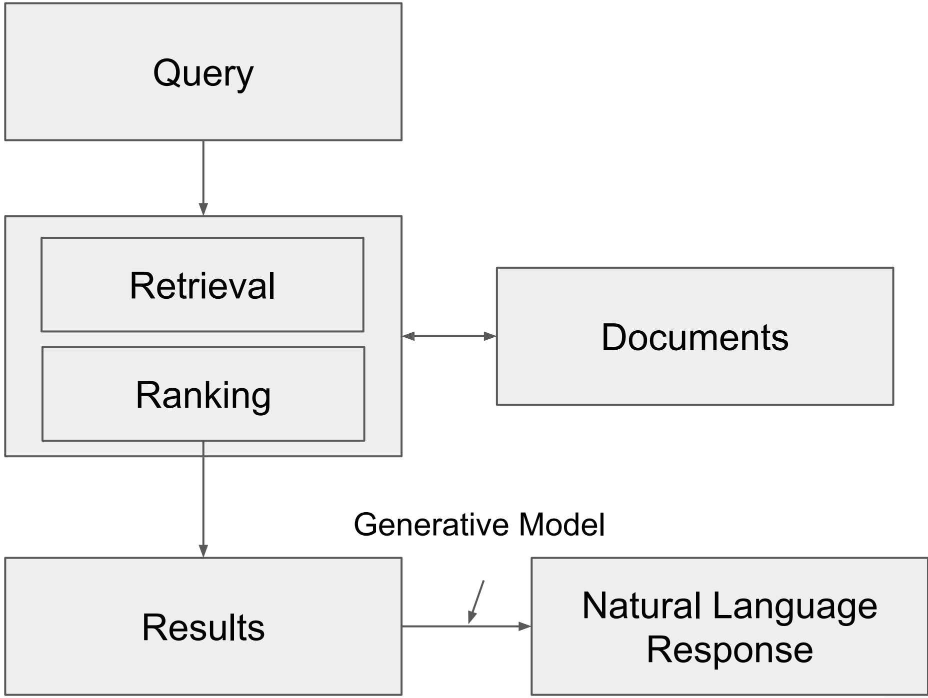
Semantic Search

+

Generative AI

=

Retrieval Augmented Generation (RAG)

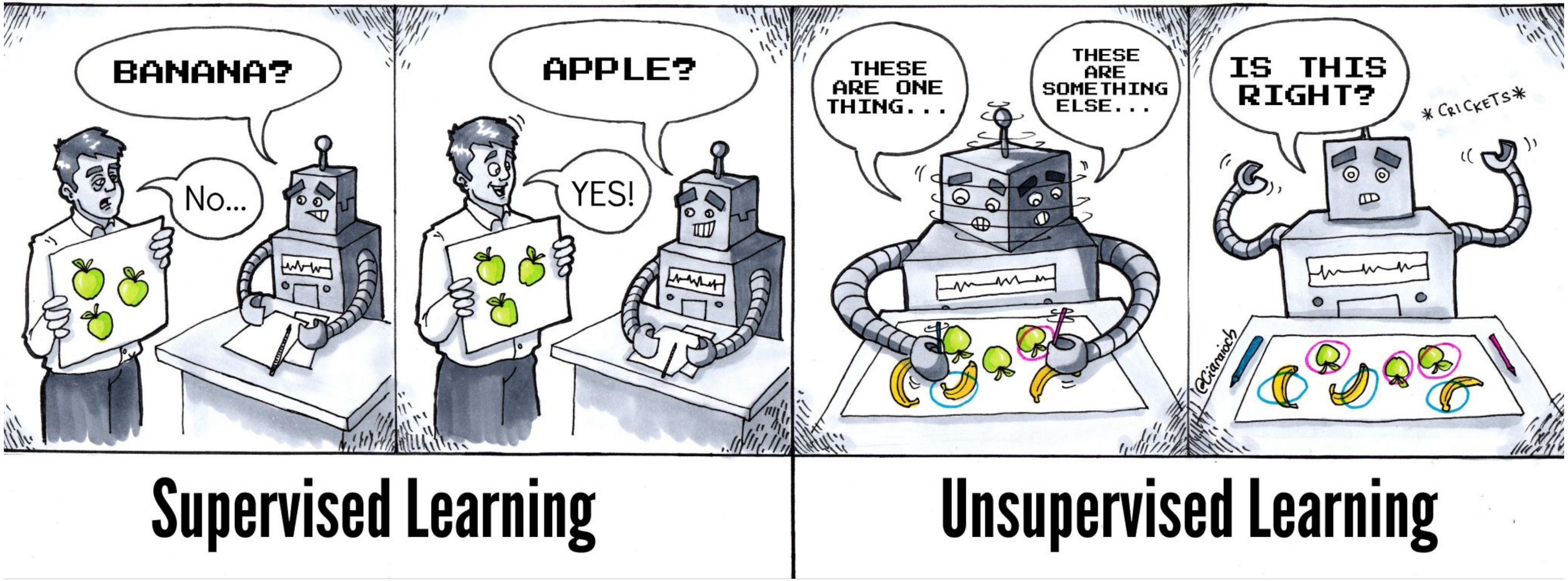


Natural Language Processing - Search

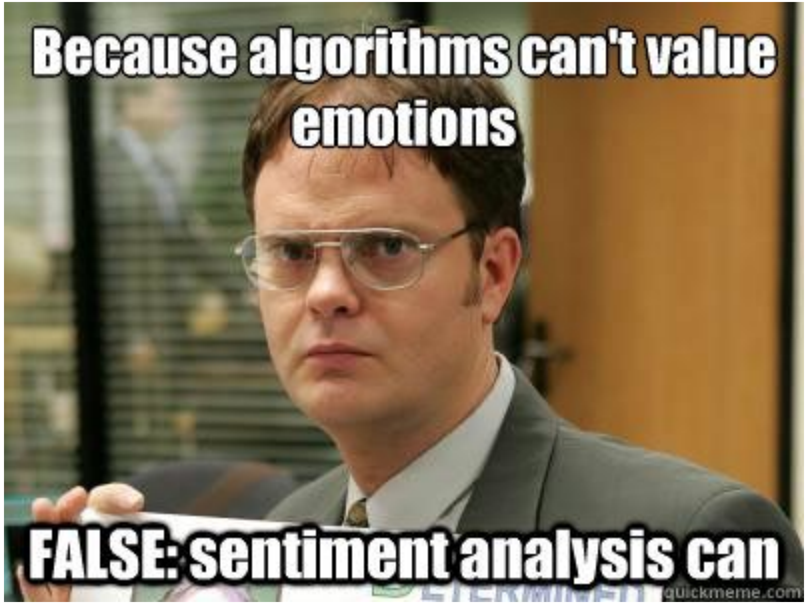
Semantic Search
+
Generative AI
=
Retrieval Augmented Generation (RAG)

Natural Language Processing

Next up:



Natural Language Processing



Natural Language Processing