Semantico

SEMANTICO: SMART WIKI SEARCH

BY PATRICK YAMIN

About Me:

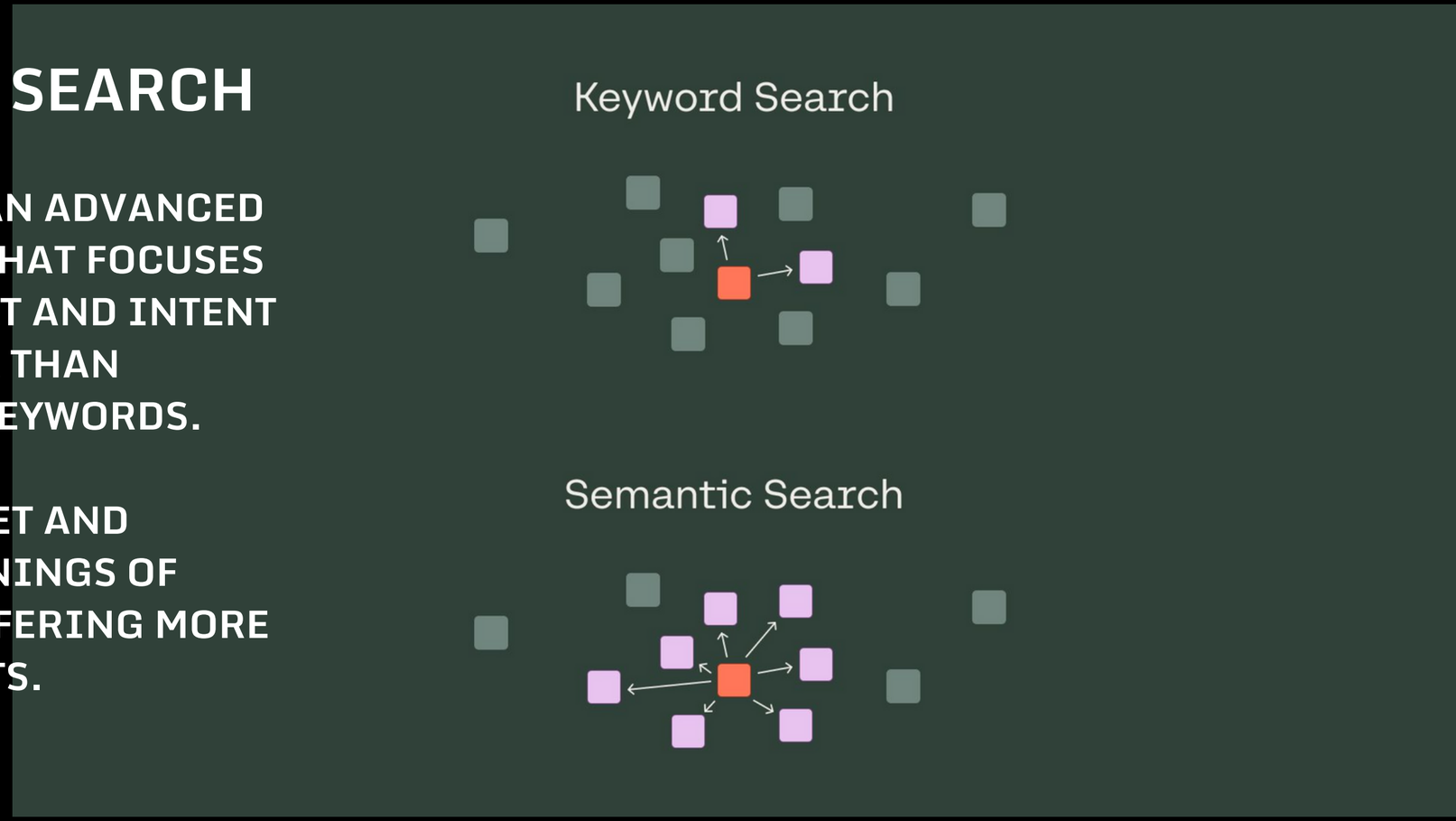
- I use Linux and in my opinion it’s the best operating system around
- Data Scientist and aspiring ML Engineer
- Fascinated with **search**, one of the most useful tools we utilize on daily basis
- Motivations for capstone were inspired by Stack Overflow search algorithms, Google, and new SOTA web search like “Perplexity”

Business Problem - Why Semantic Search?

SEMANTIC VS KEYWORD SEARCH

- SEMANTIC SEARCH REPRESENTS AN ADVANCED APPROACH TO SEARCH QUERIES THAT FOCUSES ON UNDERSTANDING THE CONTEXT AND INTENT BEHIND A USER'S QUERY, RATHER THAN RELYING SOLELY ON MATCHING KEYWORDS.

- SEMANTIC SEARCH CAN INTERPRET AND ANALYZE THE NUANCES AND MEANINGS OF WORDS WITHIN THE CONTEXT, OFFERING MORE ACCURATE AND RELEVANT RESULTS.



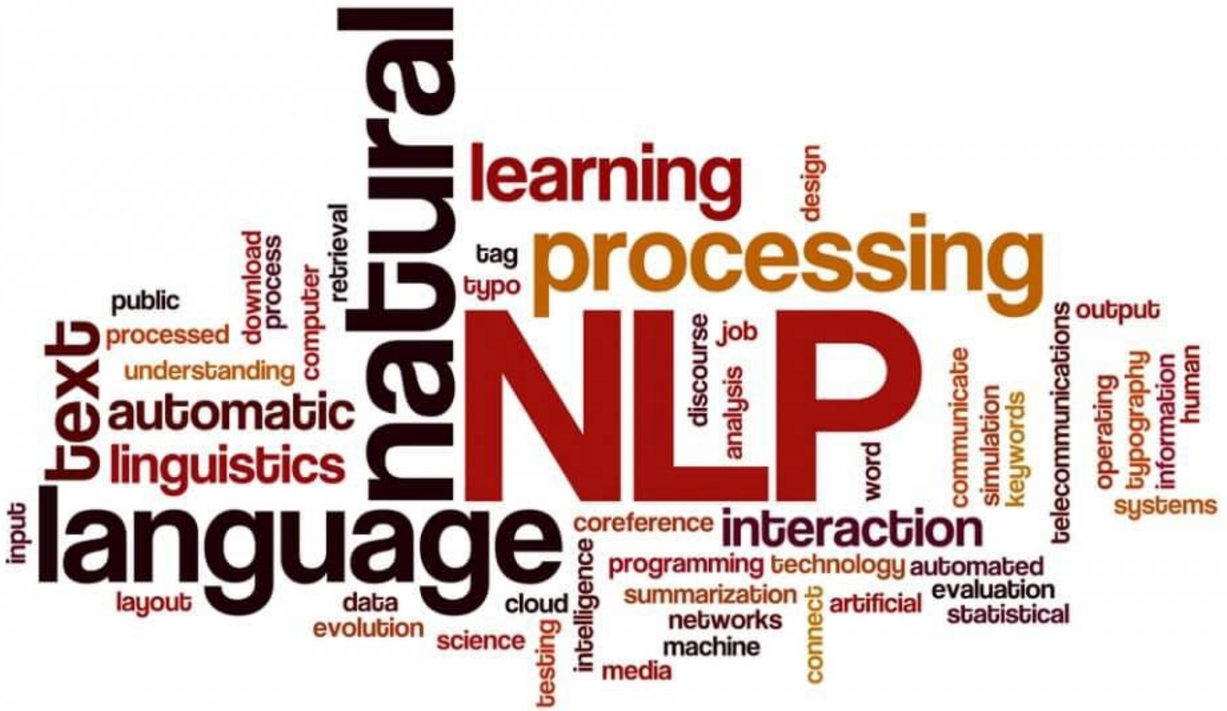
DATA MINING & UNDERSTANDING

WIKIPEDIA

- Collected through a series of API calls through the wikipedia-api

- Articles that were extracted were “Featured articles” - 6,428 articles

- Considered to be some of the best articles Wikipedia has to offer, as determined by wiki editors



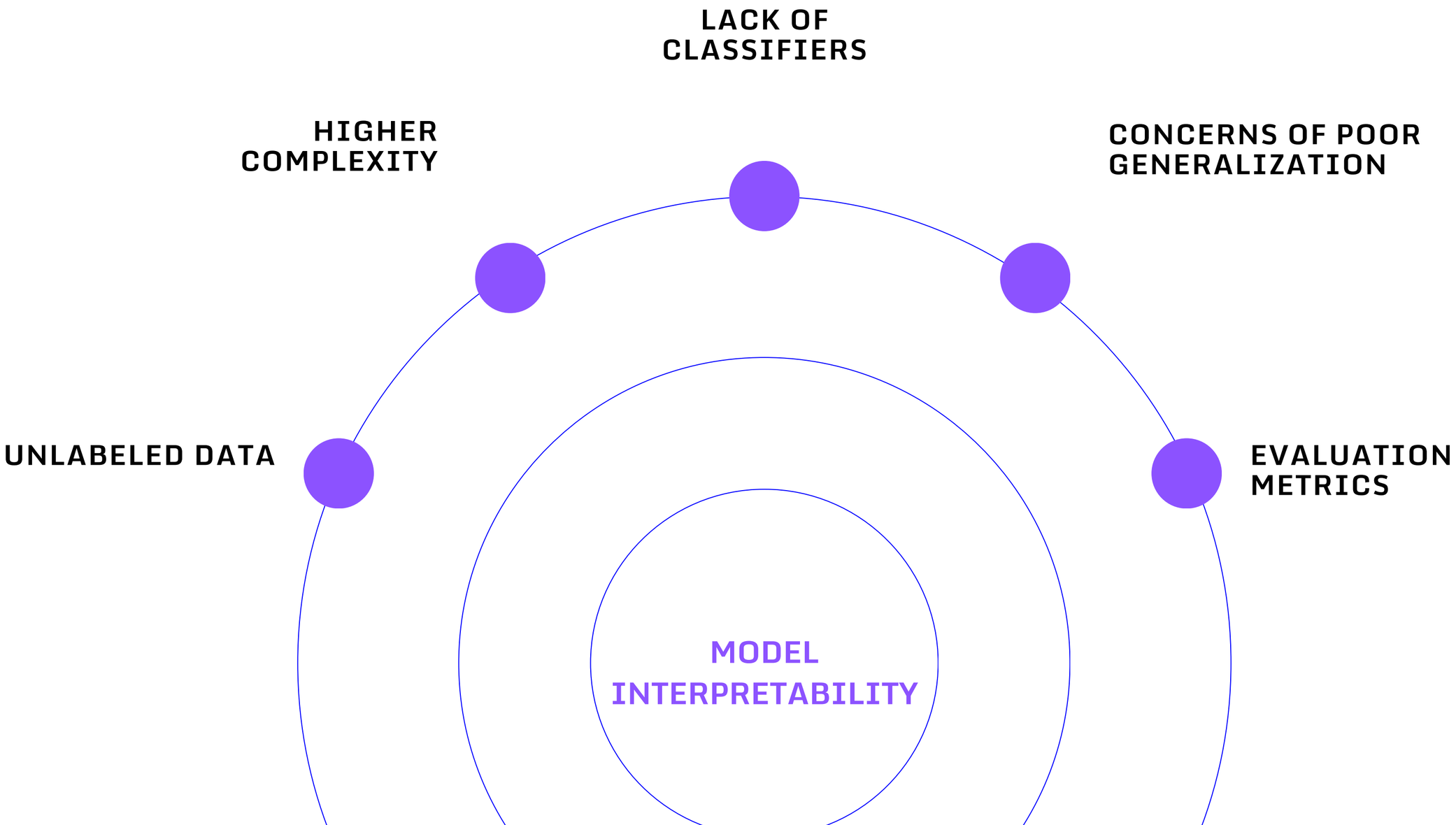
DATA CLEANING & PREPROCESSING

- Lower-casing/Standardizing text
- Removing Special Characters and Punctuation
- Replacing Line Breaks

- Stop Word Removal - Eliminating common words that add little value to the analysis, such as "the", "is", etc., using NLTK's predefined list of stop words.
- Lemmatizating
- Tokenization

- From here we can build our vector database

CHALLENGES & OBSTACLES

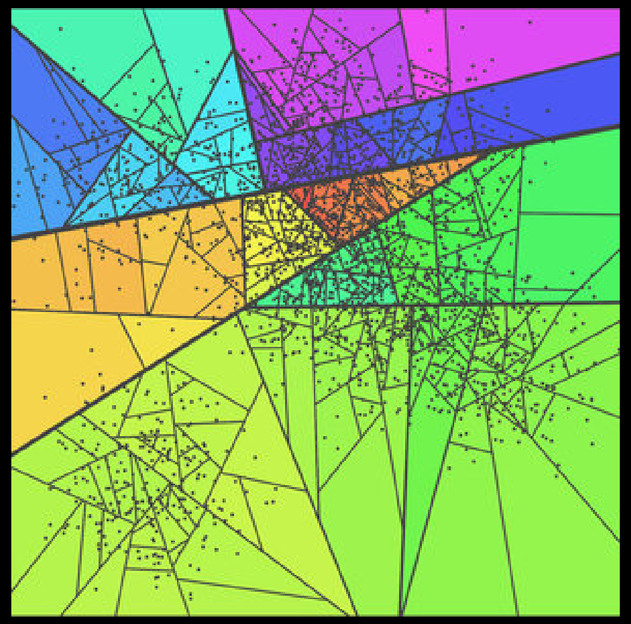


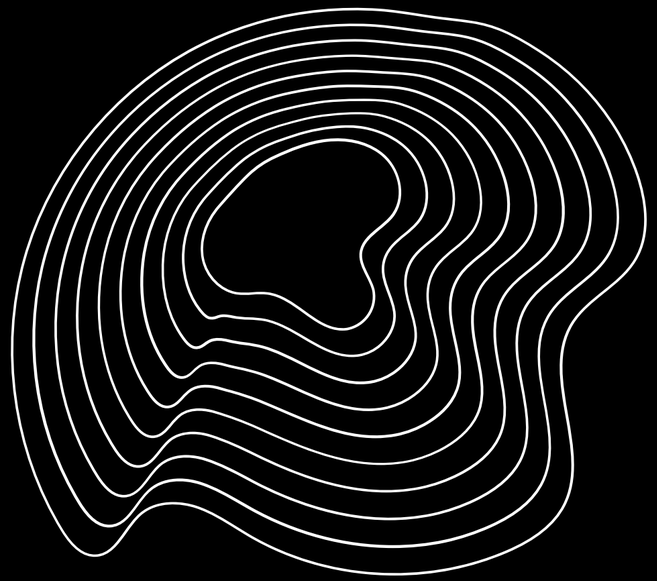
TOPIC MODELING

- LDA - Latent Dirichlet Allocation
- NMF - Non-Negative matrix factorization

Topics from NMF

```
Topic 1: ['state', 'would', 'new', 'president', 'government', 'party', 'american', 'united', 'war', 'election']
Topic 2: ['game', 'player', 'team', 'character', 'series', 'released', 'one', 'also', 'sonic', 'version']
Topic 3: ['film', 'character', 'role', 'time', 'one', 'scene', 'said', 'also', 'best', 'series']
Topic 4: ['specie', 'also', 'may', 'found', 'one', 'bird', 'used', 'known', 'year', 'two']
Topic 5: ['ship', 'fleet', 'gun', 'two', 'british', 'navy', 'battleship', 'sea', 'naval', 'squadron']
Topic 6: ['first', 'season', 'team', 'match', 'league', 'club', 'second', 'two', 'player', 'run']
Topic 7: ['album', 'song', 'music', 'band', 'released', 'single', 'first', 'record', 'one', 'number']
Topic 8: ['city', 'river', 'park', 'area', 'state', 'road', 'highway', 'north', 'building', 'new']
Topic 9: ['king', 'work', 'henry', 'year', 'first', 'one', 'also', 'time', 'england', 'later']
Topic 10: ['army', 'force', 'division', 'war', 'german', 'attack', 'battle', 'japanese', 'troop', 'battalion']
```





UNSUPERVISED LEARNING

- K-Means Clustering
- PCA

ANNOY

- Approximate Nearest Neighbors Oh Yeah
- Mixed Results

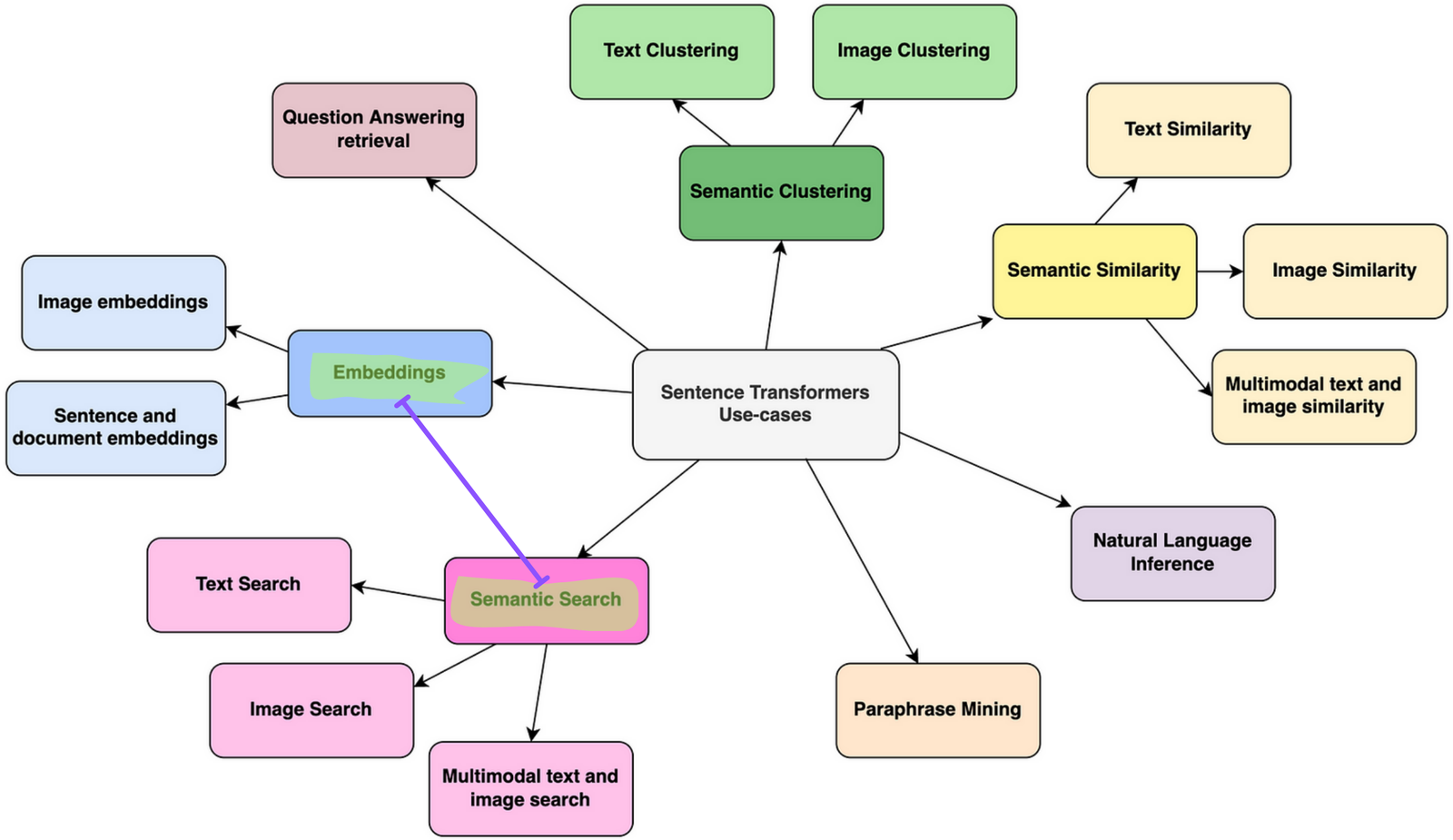
SENTENCE TRANSFORMERS

- Best Performing Model
- Based on cosine similarity score

SENTENCE TRANSFORMERS & MULTI-VECTOR SEARCH

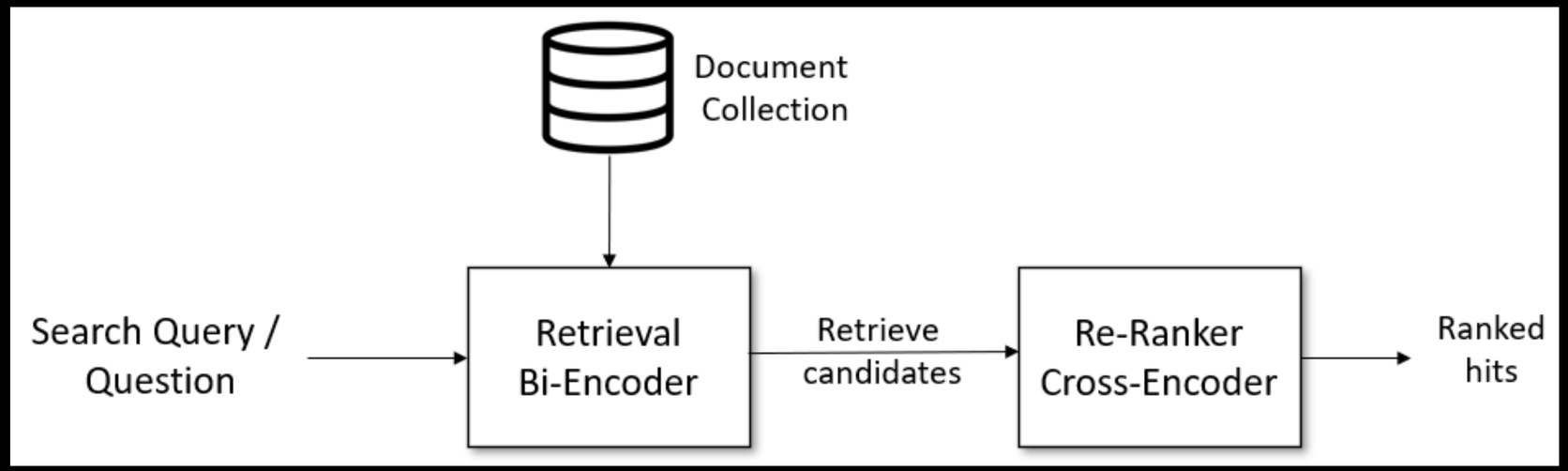
- State-of-the-art sentence, text and image embeddings
- For search or information retrieval, we would like to see accurate search results, encompassing wide-range queries
- SentenceTransformers is built on PyTorch and Transformers architecture

SOTA Model: all-MiniLM-L6-v2

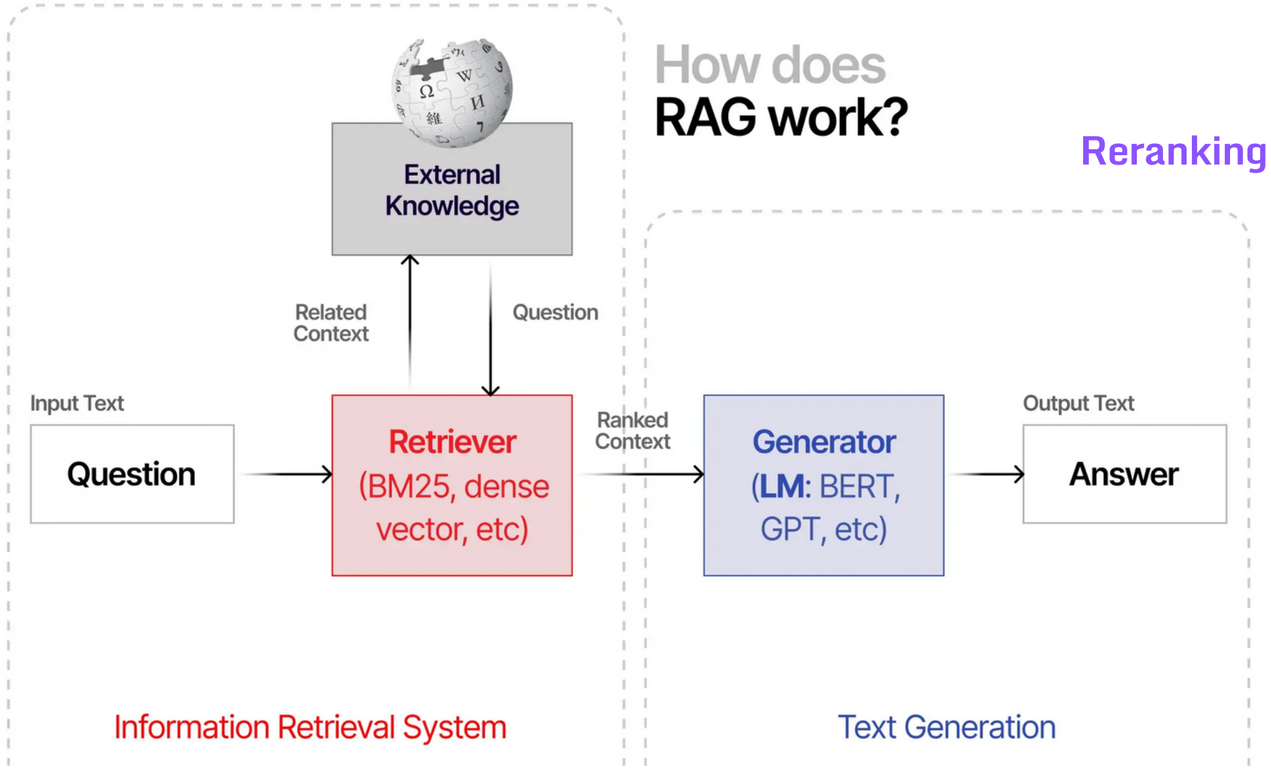


Retrieve & Re-Rank Pipeline

- In order to improve our semantic search, cross-encoders and bi-encoder models were implemented in a search pipeline
- Retrieval system might retrieve documents that are not that relevant for our search query
- Use Cross-Encoder for re-ranking, scores the relevancy of all documents for the given search query



CONCLUSIONS AND STRETCH GOALS



Reranking can be used in a RAG pipeline

- Sentence Transformers are SOTA
- Search is not yet solved but vastly improving

- Utilize Decoder Models, like GPTs to summarize text from relevant documents
- Manually annotate/label data

- RAG
- Retrieval Augmented Generation
- Pipeline chained with LMs

THANK YOU

GitHub

www.github.com/pyamin1878

substack

https://substack.com/@patrickyamin

Patrick Yamin