

Al-Enhanced
Information Retrieval
System with Big
Data Analytics and
NLP Transformers

YUNUS EMRE ISIKDEMIR

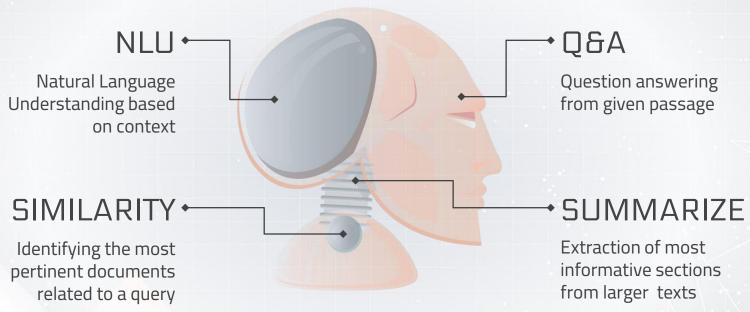
### MOTIVATION

As the volume of data continues to grow, obtaining relevant information from it becomes more challenging, particularly since not all parts of the data are equally important. Therefore, it is necessary to handle, filter, and process the data before modeling it to obtain better inferences. In this regard, Big Data Tools are essential, as they can efficiently process vast amounts of data and retrieve relevant information in a matter of seconds. To this end, various Al-based technologies such as ChatGPT and BARD have emerged, which can extract abstractive information from this massive amount of data.

In response to the challenge of extracting relevant information from large volumes of data, I developed an AI-based information retrieval system in this study. The system enables rapid access to pertinent data and performs specific tasks based on user specifications.

## PROBLEM DESCRIPTION







# Languages, Tools and Frameworks

#### **FASTAPI**

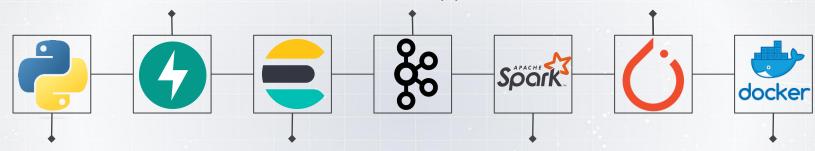
HTTP requests for AI models

### **APACHE KAFKA**

Data streaming and real-time data pipeline

### **PYTORCH**

Training and Finetuning the Al Models



### **PYTHON**

Language to logic implementaion

### **ELASTICSEARCH**

Storing and indexing the documents

### **APACHE SPARK**

Real-time data processing and ETL

### DOCKER

Containerization and isolation of the applications

# DESIGN SPECIFICATIONS



# STAGES OF THE PIPELINE

01

ETL

The data is first preprocessed with Spark, then consumed from Kafka, before being utilized in Applications.

02

QUERY

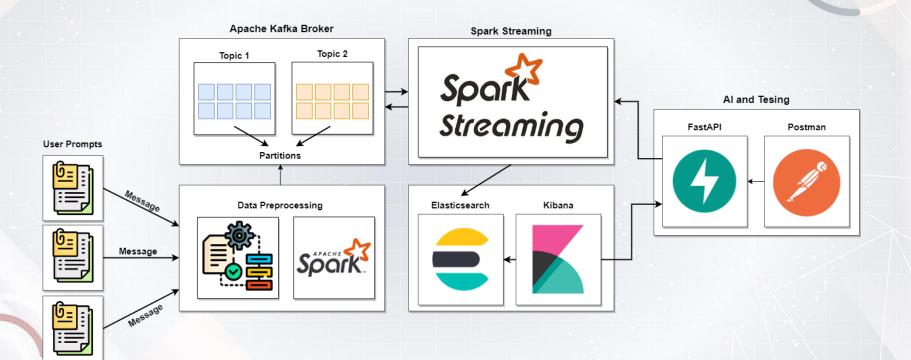
Prompt a query and send a HTTP request to trigger a particular operation

03

RETRIEVE

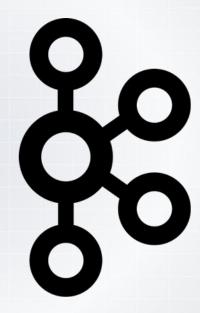
Retrieve relevant data from the database and perform specific actions based on the query.

# DATA INGESTION PIPELINE



### DATA STREAMING

Data is continuously streamed into Kafka in real-time, making it readily available for consumption by all applications.



### DATA WRANGLING

Apache Spark is employed to preprocess text data prior to publishing it, as well as for performing read and write operations with Elasticsearch. By processing queries similarly to indexed documents, the data can be efficiently prepared for Al model execution.



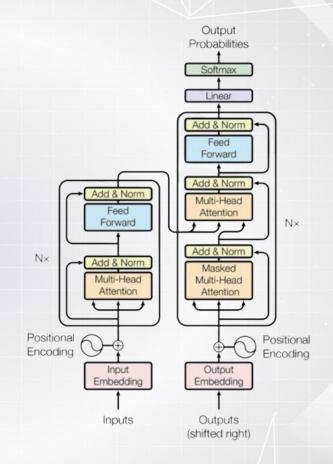
### DOCUMENT INDEXING

Elasticsearch is used to store and index documents, allowing for efficient retrieval of information relevant to a given query.



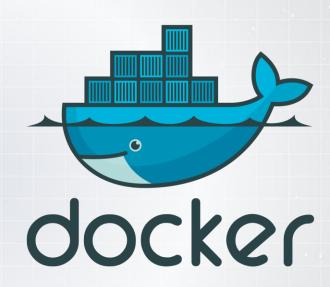
# NLP AND NLU

Cutting-edge NLP architectures are utilized to comprehend the provided prompt and carry out a particular task.



# CONTAINERIZATION

The services are containerized and isolated using Docker, a platform that allows for efficient and consistent deployment of applications across different environments.



# API DEVELOPMENT

The provision of communication between the services is facilitated through HTTP requests via FastAPI.



# 

EXPERIMENTAL RESULTS



