DATA SCIENCE PROJECT

Data Science for Scopus Dataset

Ittikorn Pornchaipimolphant 6633293821
Thaniya Kitworakiat 6633108421
Pattanakarn Cheerasemanont 6633163921
Pittaya Khunpitak 6633173121

DIAGRAM

Raw Data

+

ELSEVIER Scopus

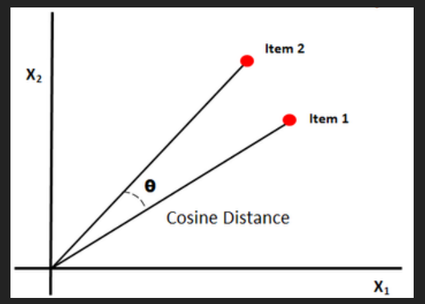
Scrape Data

APACHE Spark

pandas

CSV

all-MiniLM-L6-v2



cosine-similarity

semantic search

Streamlit

Data Extraction **Data Preprocessing** **Data Science** **Data Visualization**

DATA

- language
- title
- abstract
- authors
- keywords
- published_date
- subject_code
- citedby_count
- publisher

DATA

Raw data

(2018 - 2023)

+

ELSEVIER

Scopus

Scrape Data from Scopus API

(2011 - 2017)

DATA PREPROCESSING

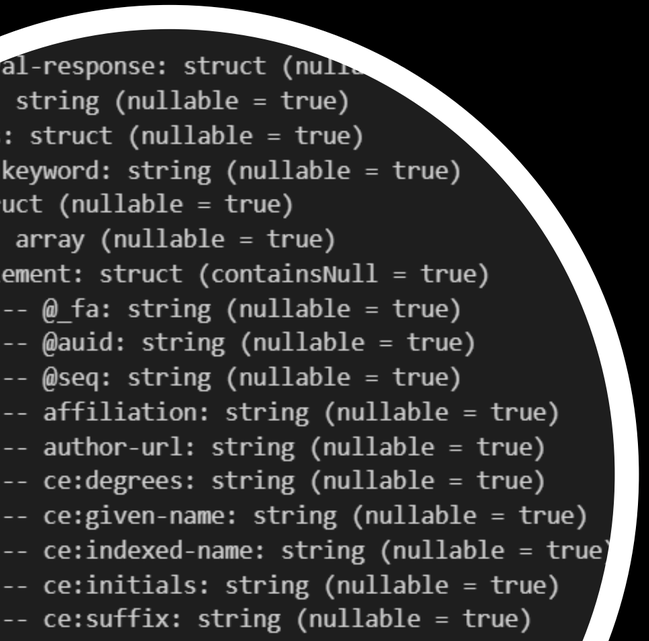
Web Scraping

Understanding Data

Clean Data

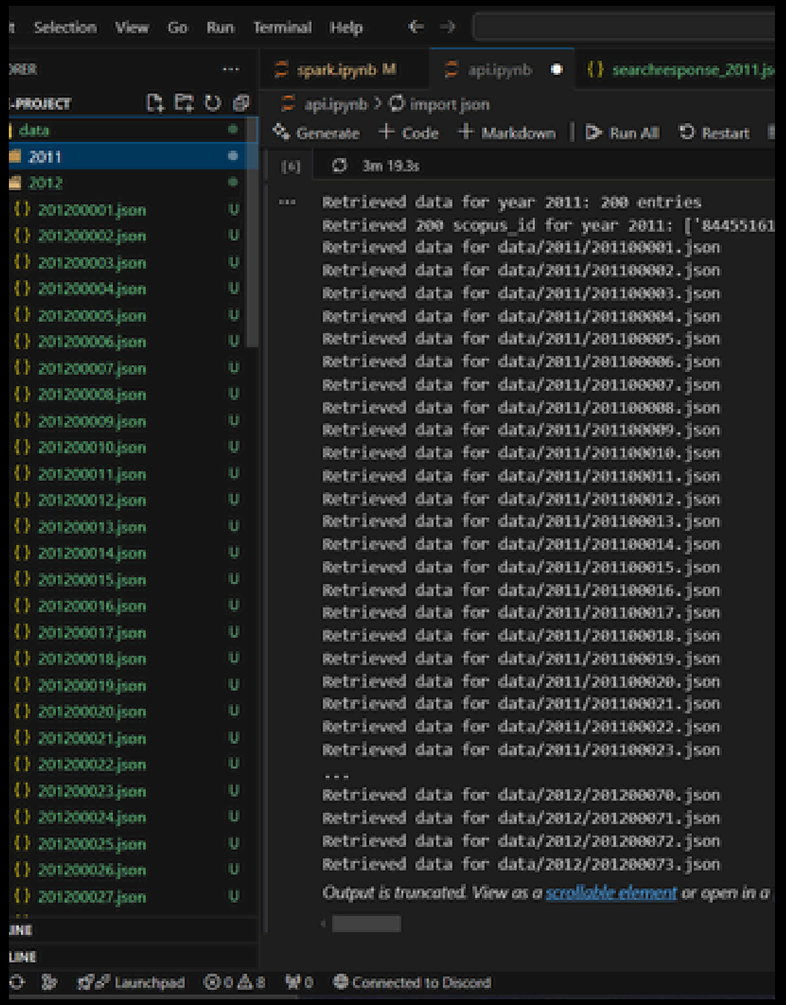
Query

See details



WEB SCRAPING

Next Page

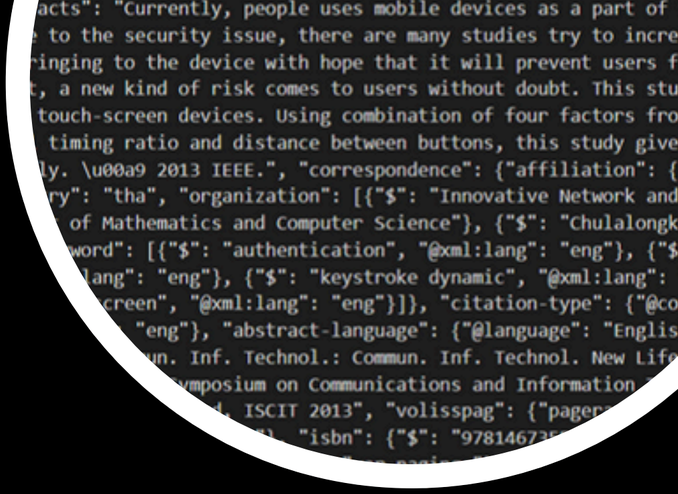


ELSEVIER
Scopus

```python
for year in range(2011, 2018):
    url = f"https://api.elsevier.com/content/search/scopus"
    params = {
        "apiKey": "YOUR_API_KEY",
        "insttoken": "YOUR_INST_TOKEN",
        "httpAccept": "application/json",
        "query": f"PUBYEAR = {year} AND AFFILORG('Chulalongkorn University')",
        "count": 200
    }

    response = requests.get(url, params=params)

    if response.status_code == 200:
        try:
            data = response.json()
            print(f"Retrieved data for year {year}: {len(data.get('search-results', {}).get('
            documentIndexes = []
            with open(f'searchresponse_{year}.json', 'w') as f:
```

UNDERSTANDING DATA

```
root
 |-- abstracts-retrieval-response: struct (nullable = true)
 |    |-- affiliation: string (nullable = true)
 |    |-- authkeywords: struct (nullable = true)
 |    |    |-- author-keyword: string (nullable = true)
 |    |-- authors: struct (nullable = true)
 |    |    |-- author: array (nullable = true)
 |    |    |    |-- element: struct (containsNull = true)
 |    |    |    |    |-- @_fa: string (nullable = true)
 |    |    |    |    |-- @auid: string (nullable = true)
 |    |    |    |    |-- @seq: string (nullable = true)
 |    |    |    |    |-- affiliation: string (nullable = true)
 |    |    |    |    |-- author-url: string (nullable = true)
 |    |    |    |    |-- ce:degrees: string (nullable = true)
 |    |    |    |    |-- ce:given-name: string (nullable = true)
 |    |    |    |    |-- ce:indexed-name: string (nullable = true)
 |    |    |    |    |-- ce:initials: string (nullable = true)
 |    |    |    |    |-- ce:suffix: string (nullable = true)
 |    |    |    |    |-- ce:surname: string (nullable = true)
 |    |    |    |    |-- preferred-name: struct (nullable = true)
 |    |    |    |    |    |-- ce:given-name: string (nullable = true)
 |    |    |    |    |    |-- ce:indexed-name: string (nullable = tru
 |    |    |    |    |    |-- ce:initials: string (nullable = true)
 |    |    |    |    |    |-- ce:surname: string (nullable = true)
 |    |-- coredata: struct (nullable = true)
...
 |    |    |    |    |-- @_fa: string (nullable = true)
```



Next Page

QUERY DATA

```
export_DF = paperDF.select(
    col("abstracts-retrieval-response.coredata.dc:title").alias("title"),
    col("abstracts-retrieval-response.coredata.dc:description").alias("abstract"),
    col("abstracts-retrieval-response.coredata.dc:creator.author.ce:indexed-name").alias("authors"),
    col("abstracts-retrieval-response.coredata.prism:coverDate").alias("published_date"),
    col("abstracts-retrieval-response.subject-areas.subject-area.@abbrev").alias("subject_code"),
    col("abstracts-retrieval-response.language.@xml:lang").alias("language"),
    col("abstracts-retrieval-response.coredata.citedby-count").alias("citedby_count"),
    col("abstracts-retrieval-response.coredata.dc:publisher").alias("publisher"),
    col("abstracts-retrieval-response.authkeywords.author-keyword").alias("keywords")
)
```

CLEAN DATA

```
# Handle missing values
data_cleaned = data.dropna(subset=['title', 'abstract', 'published_date','subject_codes'])

# Drop rows where 'keywords' column is an empty list
data_cleaned = data_cleaned[data_cleaned['keywords'].apply(lambda x: x != [])]

# Fill missing values in the 'cited' column with 0
data_cleaned['citedby_count'] = data_cleaned['citedby_count'].fillna(0)

# Convert the list in 'authors' column to a tuple to make it hashable
data_cleaned['authors'] = data_cleaned['authors'].apply(tuple)

# Remove duplicate rows based on the title and authors combination, assuming they uniquely identify a document.
data_cleaned = data_cleaned.drop_duplicates(subset=['title', 'authors'])
```

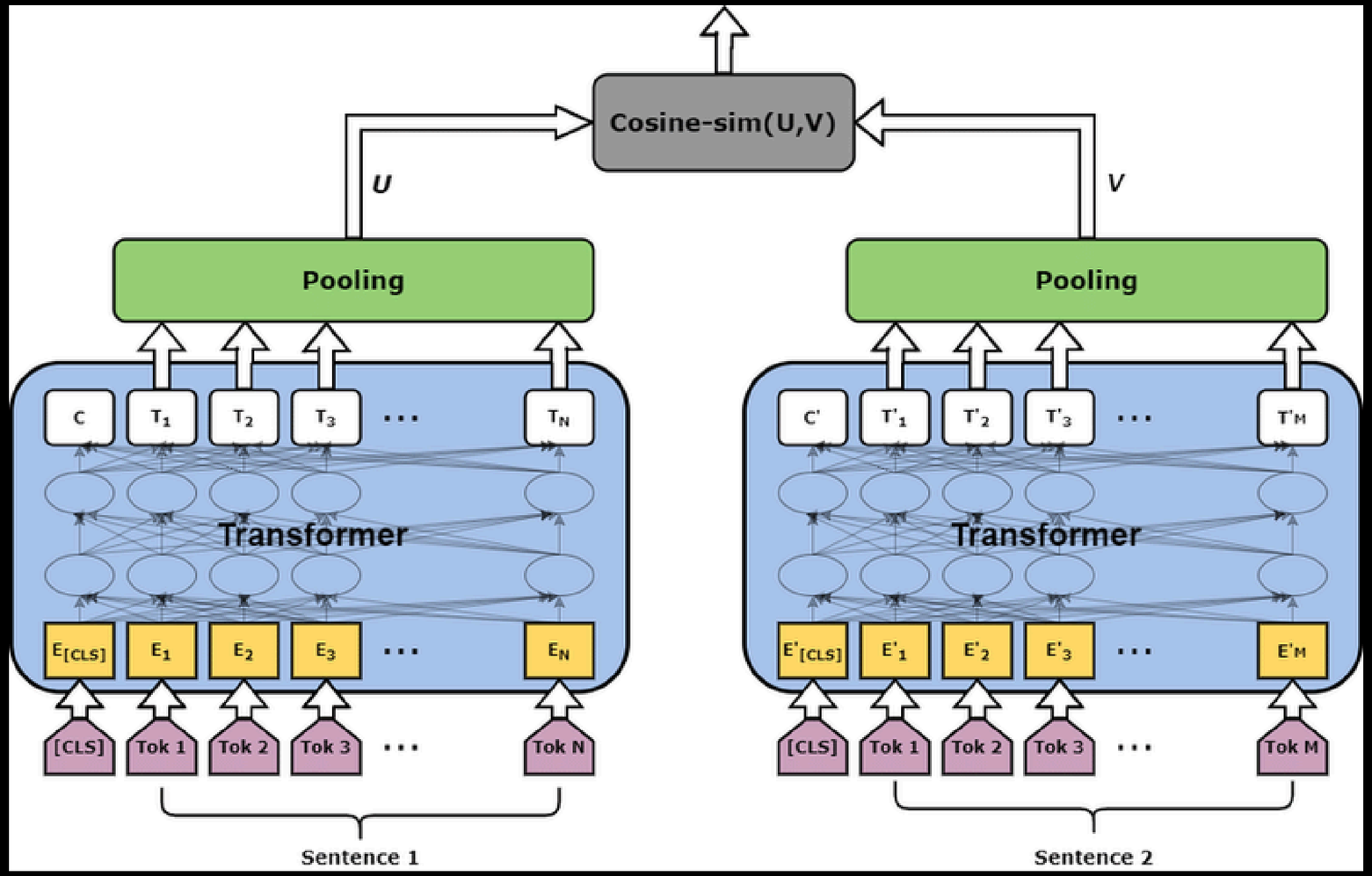
```
def clean_subject_codes(subject_codes):
    # Remove duplicates and join back as a comma-separated string.
    unique_codes = ", ".join(sorted(set(subject_codes)))
    return unique_codes

# Apply the cleaning function to the 'subject_codes' column
data_cleaned['subject_codes'] = data_cleaned['subject_codes'].apply(clean_subject_codes)
```

DATA SCIENCE

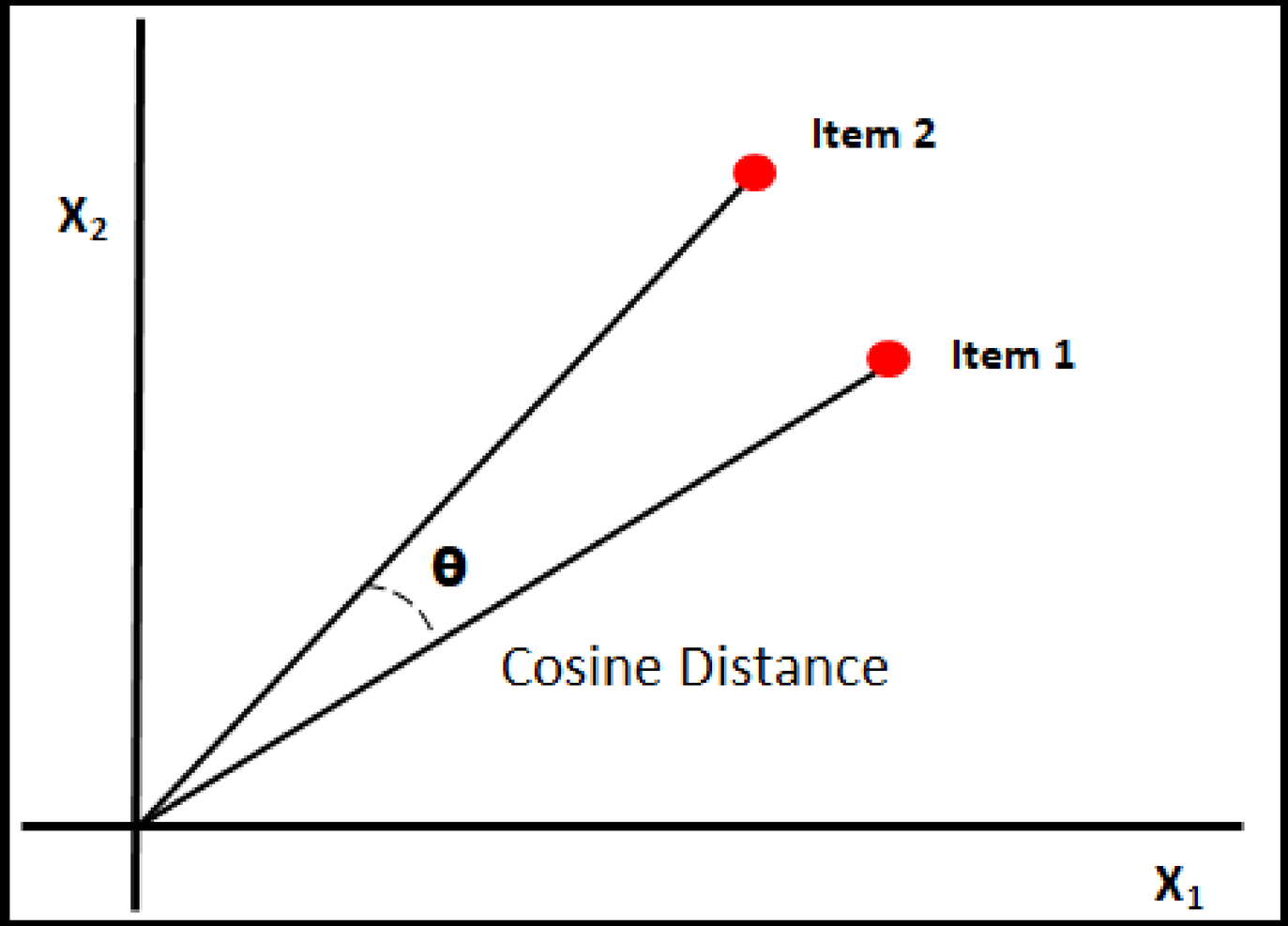
AI/ML

SEMANTIC SEARCH

SENTENCE-TRANSFORMERS

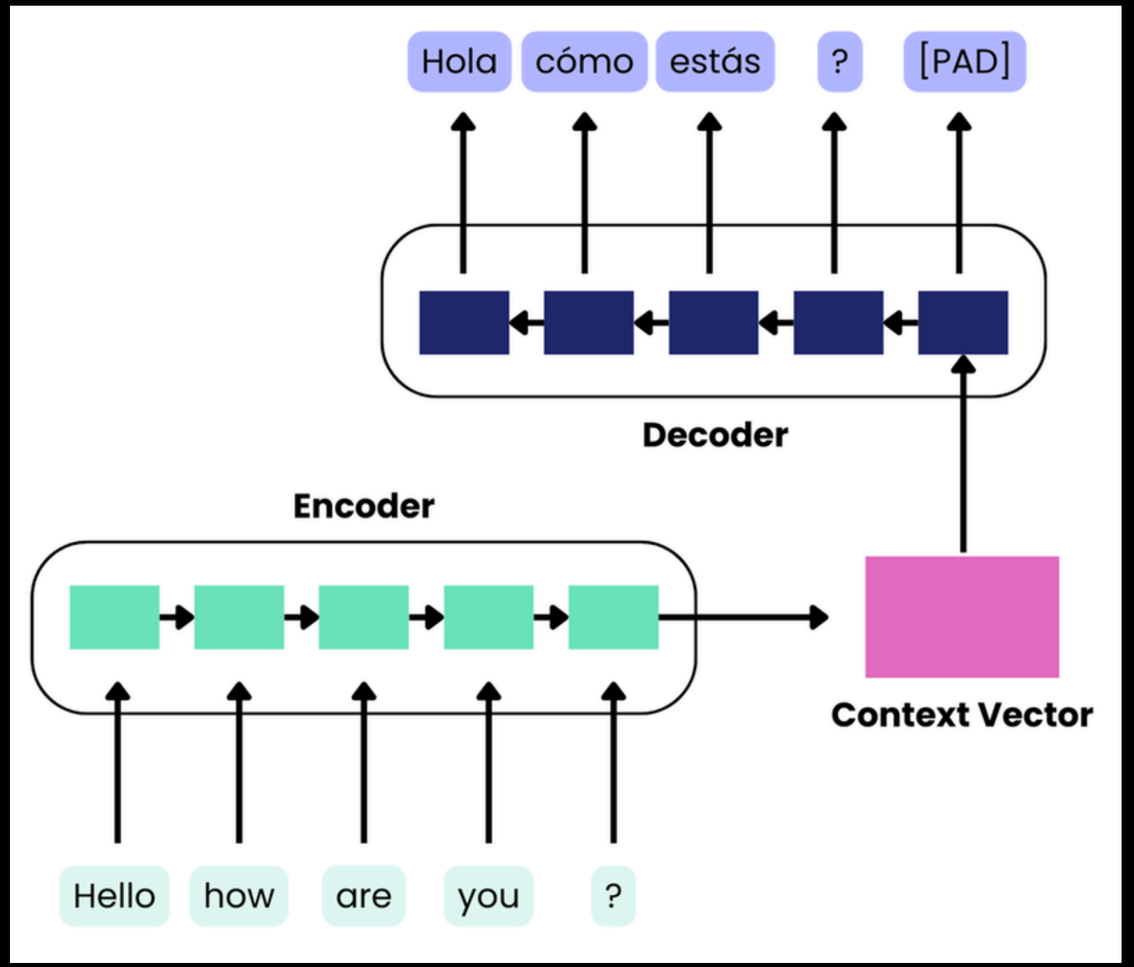


COSINE-SIMILARITY



MODEL

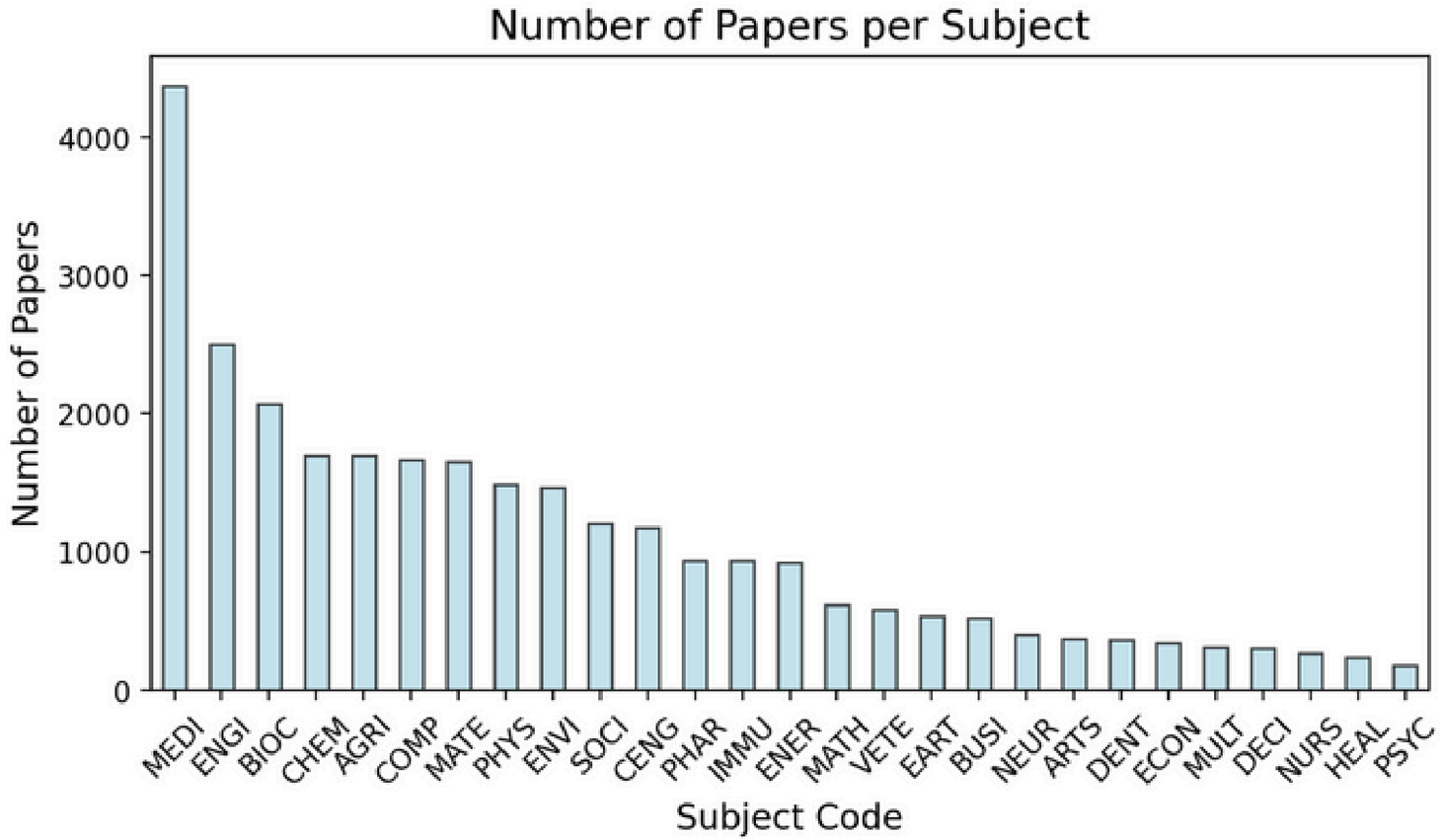
- all-MiniLM-L6-v2



DATA VISUALIZATION

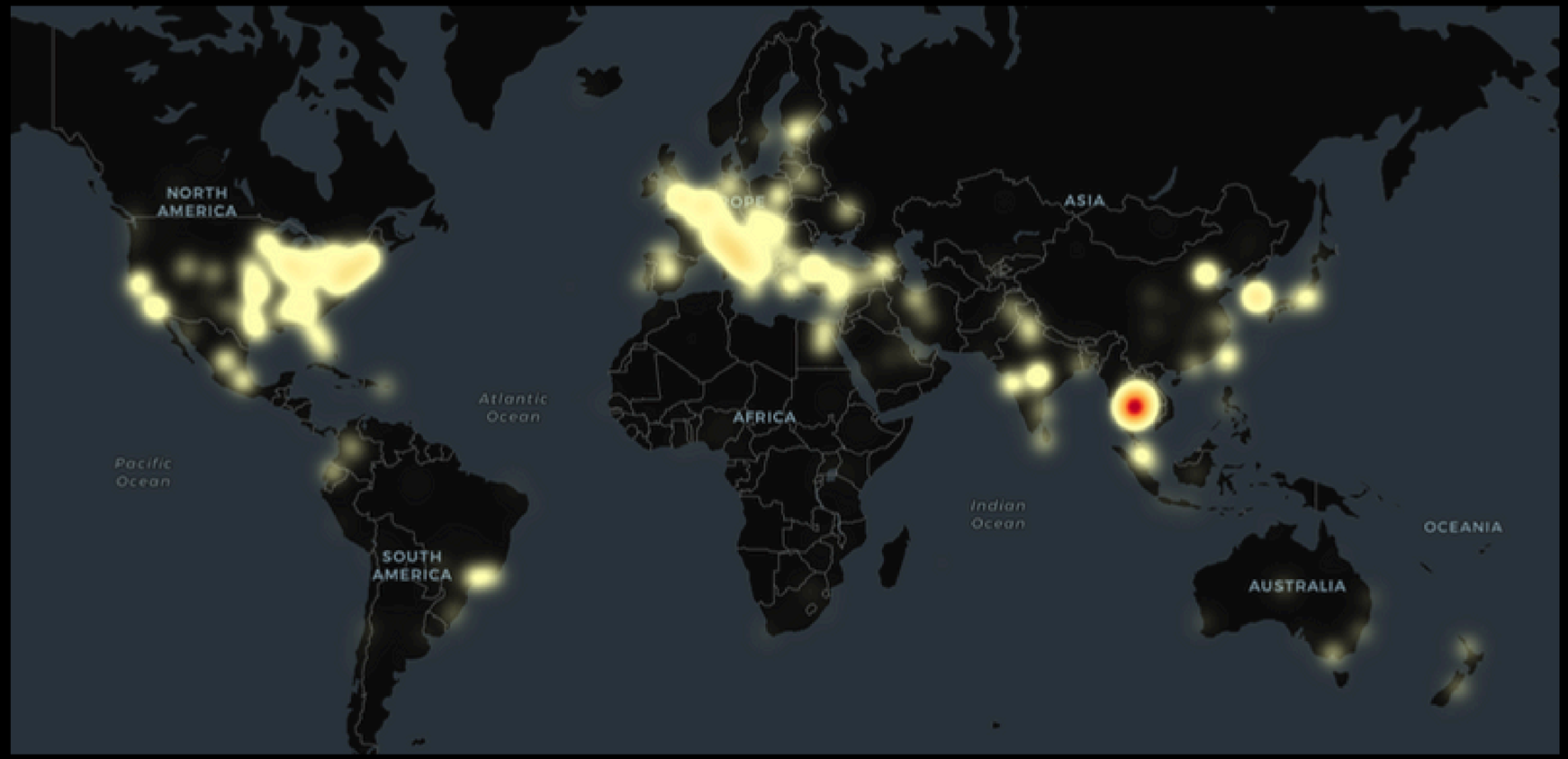
DATA OVERVIEW

Visualization: Papers Per Subject



DATA OVERVIEW

- Heatmap of location of authors



SEARCH...


Search Related Papers
Enter sentence or word:
Search
Select Similarity Percentage:
30
100
0
100
number of papers:
10

DEMO

Home About Contact

Ensuring Data Accuracy and Reliability

IMPORTANCE OF DATA CLEANING

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non.

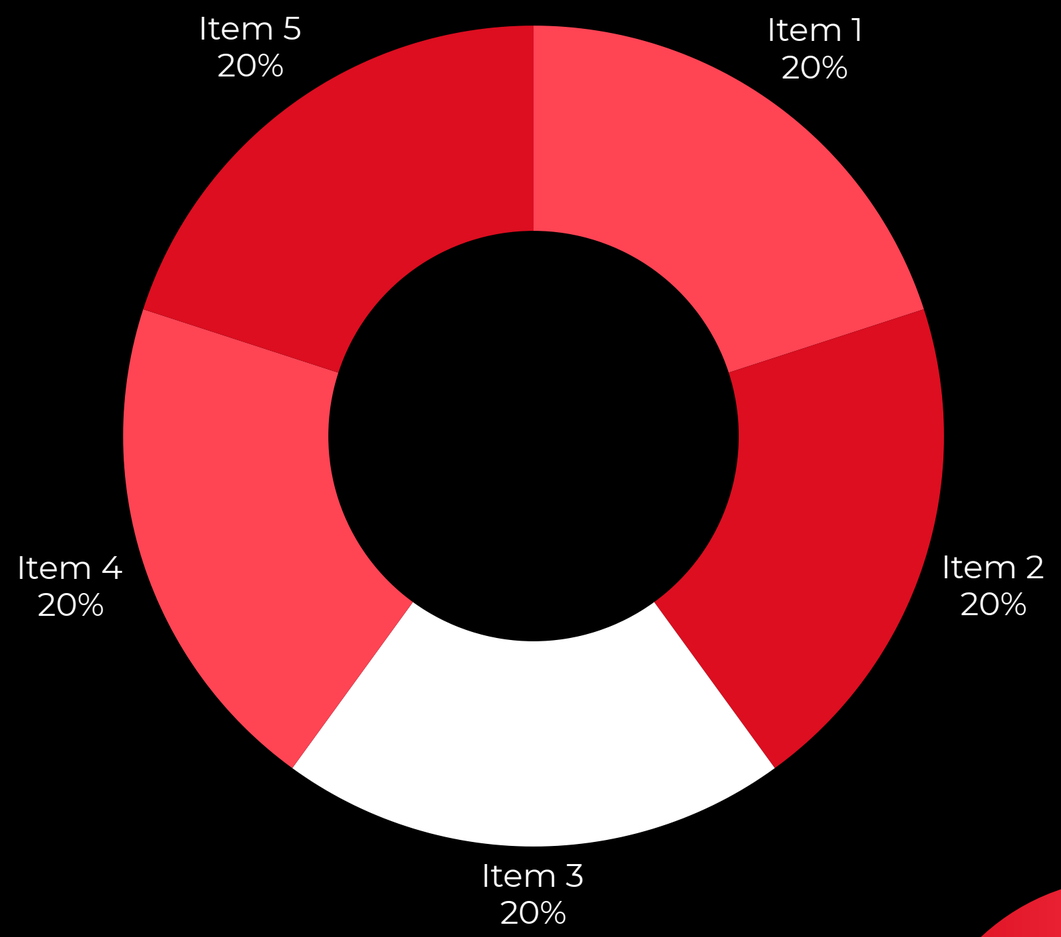
Read More

CHALLENGES IN DATA ANALYSIS

Overcoming Common Obstacles

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non.

Read More



CONCLUSION

The Future of Data Analysis

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed at ipsum vitae lacus lobortis lacinia. Donec tristique arcu massa, at pharetra tortor feugiat non.

Home About Contact

See You Next

THANK YOU

2024 Data Analysis Presentation