Big Data Analysis:
Individual Assignment

Aleksandr Jan Smoliakov
VU MIF DS year 1

Project Overview & Technology

Project: Medium Articles - Topic Modeling and Trend Analysis over Time

- **Objective:** Automatically discover topics within a large dataset of ~190,000 Medium articles and analyze how their popularity has evolved over time (2016-2022).
- **Challenge:** Manually tracking trends in such a massive volume of text is impossible. This project provides a scalable, data-driven solution.
- **Key technologies:**
  - **Apache Spark:** Distributed, large-scale data processing and machine learning functions.
  - **PySpark MLlib:** Text preprocessing pipeline and training the LDA topic model.
- **Input:** Medium articles in CSV format.
- **Output:** Visualizations - a wordcloud for each topic, and topic popularity trendlines.

data_loader.py

Implementation - Data Filtering & Preparation

- **Data loading:** The full dataset of Medium articles was loaded into a Spark DataFrame.
- **Date filtering:** The dataset was filtered to a consistent date range (2016-2022).
- **Text preprocessing:** A Spark ML pipeline was created to prepare the article text for modeling:
  - **Cleaning:** Converted text to lowercase and removed all special characters/punctuation.
  - **Tokenization:** Split the cleaned text into individual words.
  - **Stopword removal:** Removed common English words that do not carry significant meaning.
  - **TF-IDF vectorization:** Converted the tokenized text into numerical features, weighing the importance of each word in the corpus.

Implementation - Topic Modeling and Trend Detection

- **Topic modeling with LDA:**
  - A Latent Dirichlet Allocation (LDA) model was trained on the vectorized text data to identify latent topics within the articles.
  - The model was configured to discover 12 distinct topics. The parameter **`num_topics`** can be set in **`config.yml`**.
- **Trend detection:**
  - **Topic assignment:** For each article, the LDA model calculated the probability distribution across all 12 topics.
  - **Temporal aggregation:** The data was grouped by month and year.
  - **Trend calculation:** The average prevalence of each topic was calculated for every month, creating a time series that shows how topic popularity changed in 2016-2022.

visualization.py

Implementation - Visualizations

- **Topic trendlines graph:**
  - Plots the monthly popularity of all 12 topics on a single graph.
  - Data smoothed to reduce noise while preserving long-term trends.
- **Topic word clouds:**
  - A separate word cloud was generated for each of the 12 topics.
  - Shows the most important and representative words that define a specific topic.

Running the Analysis Pipeline

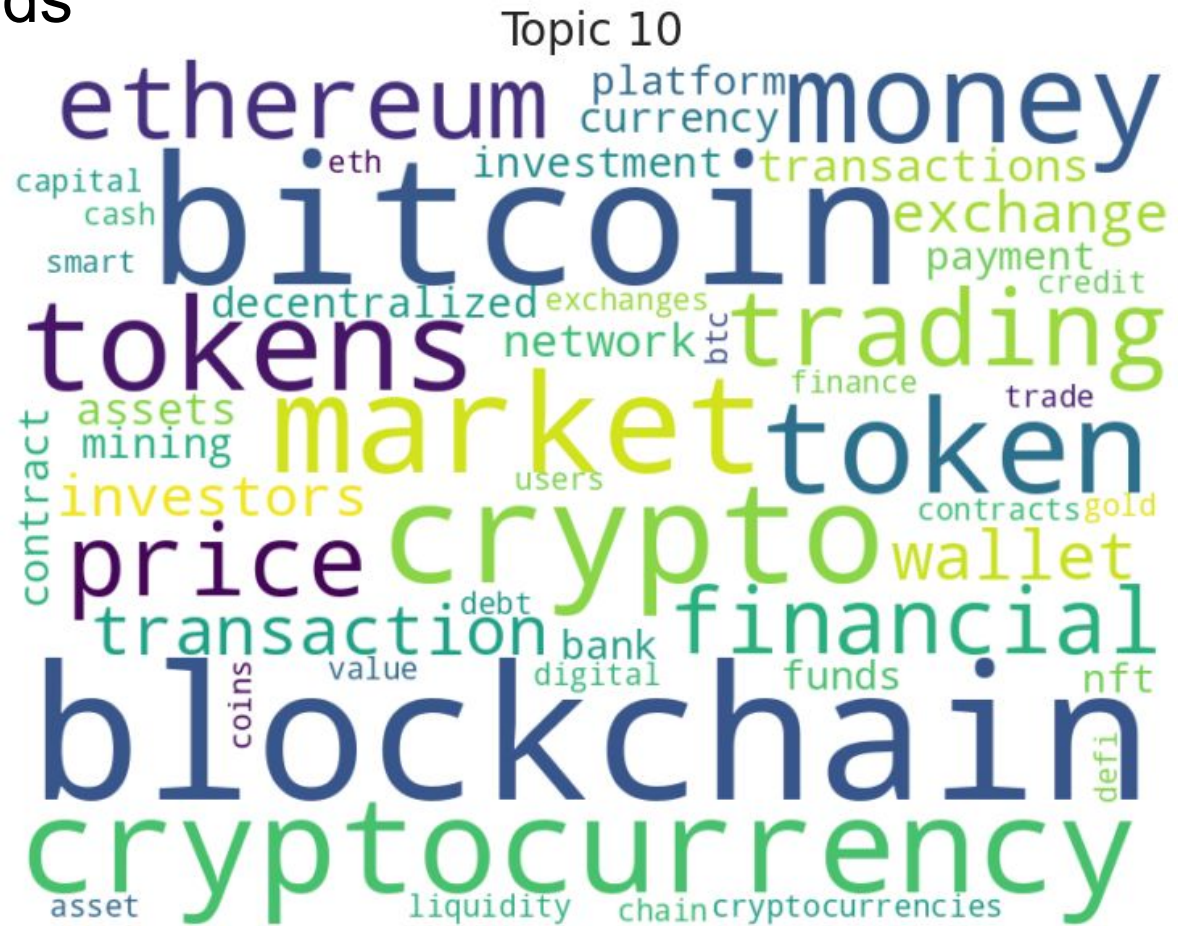
```
$ [poetry run] python main.py
```

```
<timestamp> - INFO - Spark session started.
<timestamp> - INFO - Starting data preprocessing...
<timestamp> - INFO - Loaded dataset with 192368 articles and 6 columns.
<timestamp> - INFO - Filtered dataset to 191708 articles within the date range 2016-2022.
...
--- Discovered Topics ---
<timestamp> - INFO - Topic 0:
<timestamp> - INFO -    - file (weight: 0.0058)
<timestamp> - INFO -    - server (weight: 0.0052)
<timestamp> - INFO -    - cloud (weight: 0.0049)
<timestamp> - INFO -    - docker (weight: 0.0046)
...
<timestamp> - INFO - Topic 11:
<timestamp> - INFO -    - health (weight: 0.0044)
<timestamp> - INFO -    - covid (weight: 0.0039)
<timestamp> - INFO -    - food (weight: 0.0036)
<timestamp> - INFO -    - weight (weight: 0.0033)
...
<timestamp> - INFO - Generating and saving topic trends plot to data/output/topic_trends.png...
<timestamp> - INFO - Generating and saving word clouds to data/output/wordclouds...
<timestamp> - INFO - Stopping Spark session.
```

Results - Word Clouds

LDA model identified 12 topics, each represented by a set of top words.

A wordcloud for **Topic 10** is shown on the right.



Results - Temporal Trends

The trends over time show how the popularity of these topics evolved.

For instance, **Topic 10** (cryptocurrency) peaked in 2018.