DSCI 560 Lab 5 Report

Group Code Submission Link:
DSCI560---Shubham/Lab_5 at main · shubhamdarekar/DSCI560---Shubham

Code README:
DSCI560---Shubham/Lab_5/README.md at main · shubhamdarekar/DSCI560---Shubham

Meeting Notes Link:
https://docs.google.com/document/d/1FuQokQmE9pQ65swm9uktWQRXy4isTVdv7BcAi0clgLY/edit?usp=sharing

Team Details: Group 1
Saavani Vaidya 9385579920
Yuxuan Liu 4780355176
Shubham Darekar 1641138809

1. Introduction

This project focuses on web scraping, data preprocessing, clustering algorithms, and real-time data processing. The goal is to collect and organize data from Reddit, preprocess and clean the extracted content, perform clustering to identify similar messages, and automate the process at fixed intervals.

2. Tools and Libraries Used

The following Python libraries were utilized for implementation:

- Web Scraping & API Access: praw, requests, selenium, beautifulsoup4
- Database Management: pymysql
- Text Processing & NLP: nltk, textblob, gensim
- Machine Learning & Clustering: scikit-learn
- Optical Character Recognition (OCR): pytesseract
- Visualization: matplotlib

3. Data Collection & Storage

3.1 Web Scraping & Reddit API

- Data was collected using the PRAW.
- The script fetched posts from the r/askscience subreddit and stored the specified number of posts in a MySQL database.
- The API’s request limits were handled to ensure large-scale data retrieval efficiently.

3.2 Database Setup

A MySQL database named reddit_data was created with the following table structure:

```
CREATE TABLE IF NOT EXISTS posts (
   id INT AUTO_INCREMENT PRIMARY KEY,
   title TEXT,
   content TEXT,
   timestamp DATETIME,
   subreddit VARCHAR(255),
   keywords TEXT,
   image_text TEXT,
   cluster_id INT,
   cluster_id_doc INT
);
```

4. Data Preprocessing

- Cleaning: HTML tags, special characters, URLs, and usernames were removed from posts.
- Keyword Extraction: Stopwords were removed, and keywords were extracted using tokenization and filtering.
- Text Extraction from Images: pytesseract was used to extract embedded text from post images.
- Data Formatting: Timestamps were converted, and usernames were masked to preserve privacy.

5. Clustering Algorithm Implementation

5.1 Embedding Generation

- Text content was converted into vector representations using Word2Vec.
- The average word vector for each post was used as its representation in the clustering model.

5.2 Clustering with K-Means

- K-Means clustering was applied to group posts into 5 clusters.
- The cluster assignment was stored in the MySQL database.
- Keywords were extracted for each cluster to summarize group topics.

6. Automation

6.1 Periodic Data Collection and Processing

- The script automate.py runs the full pipeline every user-specified minute.
- The automation fetches, preprocesses, and clusters new posts.
- A countdown timer ensures efficient execution between updates.

6.2 Interactive Search

- Between updates, users can input search queries.
- The model predicts the cluster of the query and returns related posts.

7. Results and Observations

- Successfully collected 5000+ posts from r/askscience.
- Preprocessing improved the quality of stored content.
- Clustering provided meaningful groupings of similar content.
- Automation successfully maintained an up-to-date dataset.