Project: Analysis of Indian Startup Funding and Deals Over Time

Introduction

This project involves web scraping data from the website 'entrackr.com', which provides information about funding and acquisitions in Indian startups. The goal is to extract relevant information, perform data preprocessing and analysis, and visualize the trends in startup funding and deals over time.

Importing Libraries

The project starts by importing necessary libraries, including:

- requests: For making HTTP requests to the website
- BeautifulSoup: For parsing HTML content
- re: For regular expressions
- pickle: For working with pickle files (serializing and deserializing Python objects)
- tqdm: For displaying progress bars during the web scraping process
- pandas: For data manipulation and analysis
- matplotlib.pyplot: For data visualization

Web Scraping: Link Extraction

The link_extractor function iterates over a range of pages on the website to extract links to articles related to funding and acquisitions in Indian startups. The links are extracted from the <a> tags within specific HTML elements.
The extracted links are stored in the links_list.

Web Scraping: Data Extraction

The loop iterates through each link extracted earlier.
For each link, the code retrieves the content of the webpage, extracts the date, and processes the text using BeautifulSoup.
Regular expressions are used to extract the number of deals and total funding amount from the text.
The extracted data (date, number of deals, total funding) is appended to the dataset list.

Saving Extracted Data

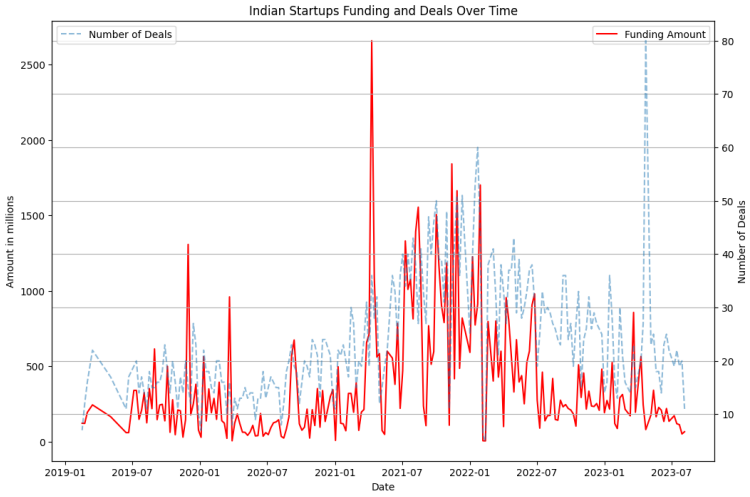
The dataset list, containing extracted data, is saved to a pickle file named "dataset_files.pkl" using the pickle.dump() function.

Data Preprocessing

The dataset list is converted into a DataFrame named dataset_df with columns 'date', 'no_of_deals', and 'total_funding'.
The 'date' column is set as the index and converted to a datetime format.
The data types of the 'no_of_deals' and 'total_funding' columns are converted to 'int' and 'float', respectively.

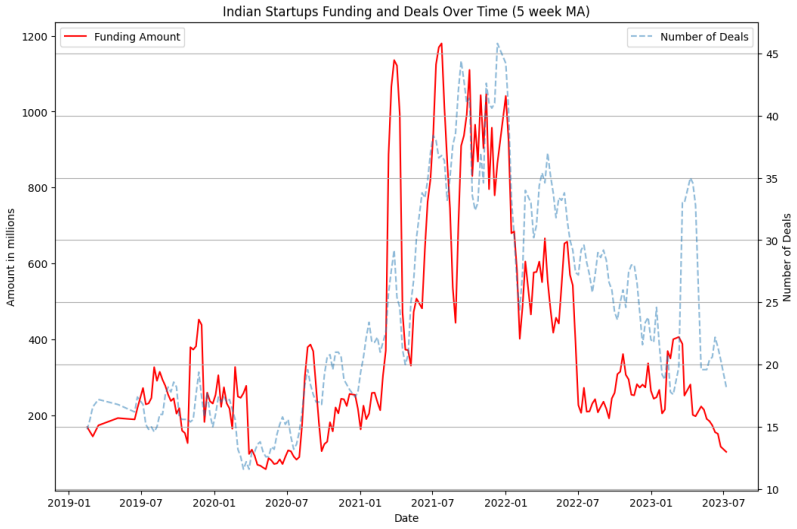
Data Visualization: Line Plot

A line plot is created using Matplotlib to visualize the trends in funding amount and number of deals over time.
Two y-axes are used to display the two different scales for funding amount and number of deals.
X-axis tick labels are rotated for better readability.
Legends are added to the plot, and the title is set.



Data Visualization: Rolling Mean Plot

Another line plot is created to visualize the trends in funding amount and number of deals using a rolling mean (5-week moving average).
Rolling mean is calculated using the rolling() function.
The plot includes legends, title, and grid lines.



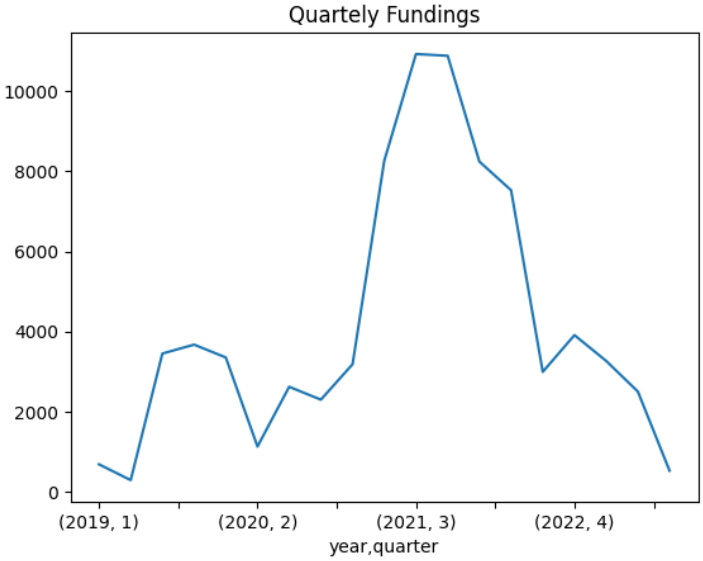
Data Aggregation: Quarterly Analysis

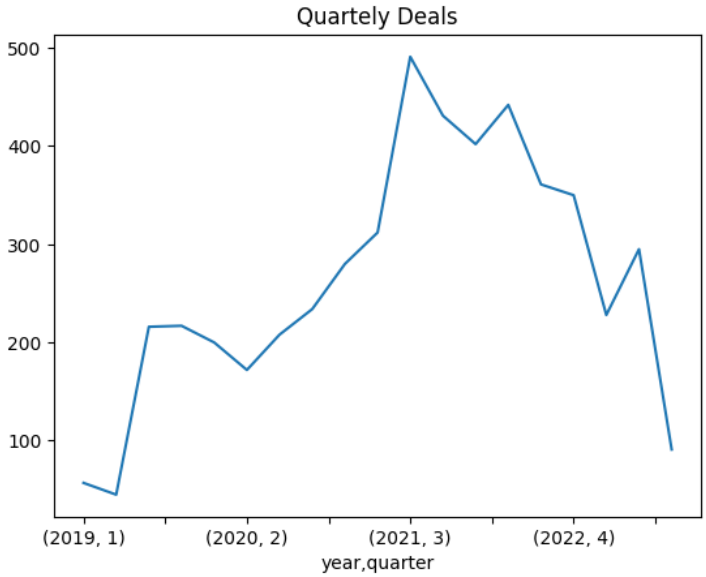
A new column 'quarter' is added to dataset_df to represent the quarter of each date.
Data is grouped by year and quarter, and the sum of total funding and number of deals is calculated.

A DataFrame named dataset_df_quarter is created to store the quarterly aggregated data.

Data Visualization: Quarterly Analysis

Line plots are created to visualize the quarterly trends in total funding and number of deals.
Legends, titles, and labels are added to the plots.



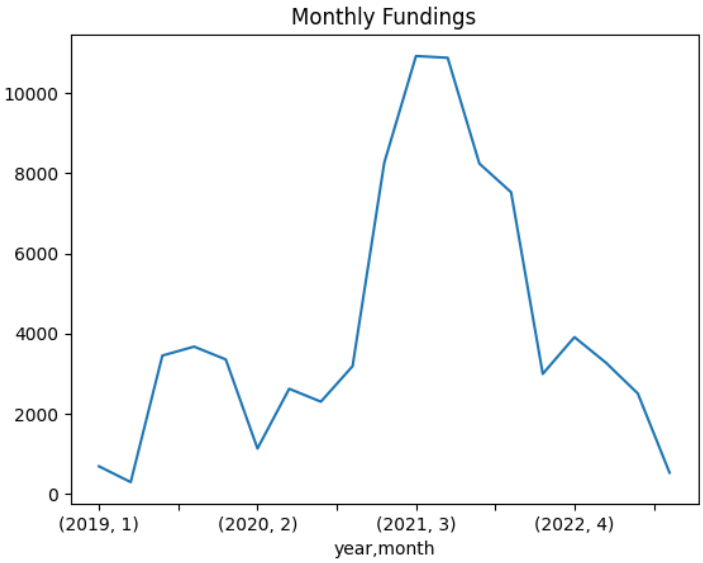


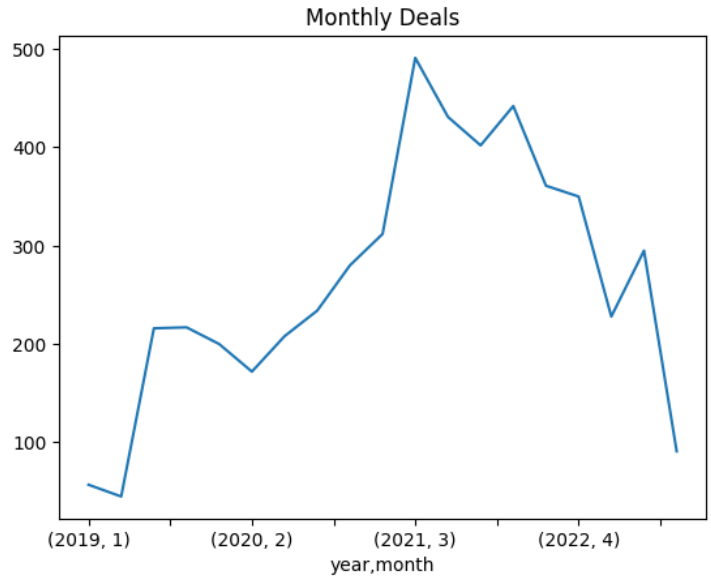
Data Aggregation: Monthly Analysis

Columns 'month' and 'year' are added to dataset_df to represent the month and year of each date.
Data is grouped by year and month, and the sum of total funding and number of deals is calculated.
A DataFrame named dataset_df_monthly is created to store the monthly aggregated data.

Data Visualization: Monthly Analysis

Line plots are created to visualize the monthly trends in total funding and number of deals.
Legends, titles, and labels are added to the plots.


Monthly Fundings
10000
8000
6000
4000
2000
0
(2019, 1)
(2020, 2)
(2021, 3)
(2022, 4)
year,month


Monthly Deals
500
400
300
200
100
(2019, 1)
(2020, 2)
(2021, 3)
(2022, 4)
year,month