1. Update the **BugHub** Graph Data Model (use **Netowkrx** Package) to attach/add to **GitHub** node a new node called **TrendingRepos** that has **top-10** repos that have the repos with highest number of Stars, Issues, Pull Requests. (Trending line chart for past 12 months data).

import pandas as pd

import requests

import json

import matplotlib.pyplot as plt

from datetime import datetime, timedelta

query\_params = {

"q": "language:python stars:>100",

"sort": "stars",

"order": "desc"

}

response = requests.get("https://api.github.com/search/repositories", params=query\_params)

if response.status\_code == 200:

    trending\_repos = response.json()["items"]

else:

    print(f"Error getting trending repositories: status code {response.status\_code}")

today = datetime.now()

last\_year = today - timedelta(days=365)

last\_year\_str = last\_year.strftime("%Y-%m-%d")

today\_str = today.strftime("%Y-%m-%d")

trending\_data = {}

for repo in trending\_repos:

# Get repository owner and name

    repo\_owner = repo["owner"]["login"]

repo\_name = repo["name"]

# Make request to GitHub API to get data for repository

url = f"https://api.github.com/repos/{repo\_owner}/{repo\_name}/stats/commit\_activity?since={last\_year\_str}&until={today\_str}"

response = requests.get(url)

# Check if response was successful and retrieve data for repository

if response.status\_code == 200:

    try:

        result = json.loads(response.text)

        if result:

            commits = [week["total"] for week in result]

            trending\_data[repo\_name] = {

                "commits": commits,

                "stars": repo["stargazers\_count"],

                "issues": repo["open\_issues\_count"],

                "pull\_requests": repo["pulls\_url"]

            }

    except ValueError as e:

        print(f"Error decoding JSON for {repo\_name}: {e}")

else:

    print(f"Error getting data for {repo\_name}: status code {response.status\_code}")

top\_10 = sorted(trending\_data.items(), key=lambda x: x[1]["stars"], reverse=True)[:10]

for repo\_name, repo\_data in top\_10:

    plt.figure(figsize=(10, 6))

plt.title(f"{repo\_name} - Trending Data")

plt.xlabel("Weeks")

plt.ylabel("Count")

# Plot the line chart for number of stars

stars = [repo\_data["stars"]] \* len(repo\_data["commits"])

plt.plot(stars, label="Stars")

# Plot the line chart for number of issues

issues = [repo\_data["issues"]] \* len(repo\_data["commits"])

plt.plot(issues, label="Issues")

# Plot the line chart for number of pull requests

pull\_requests = requests.get(repo\_data["pull\_requests"]).json()

pull\_requests\_count = [len(pr) for pr in pull\_requests]

plt.plot(pull\_requests\_count, label="Pull Requests")

# Plot the line chart for number of commits

plt.plot(repo\_data["commits"], label="Commits")

plt.legend(loc="upper left")

plt.show()

Graphical user interface, application

Description automatically generated

# Get interest over time data for the specified keywords

import pandas as pd

from pytrends.request import TrendReq

import matplotlib.pyplot as plt

pytrend = TrendReq()

keywords = ["Angular Material", "React", "D3", "Flask", "TensorFlow"]

pytrend.build\_payload(keywords, timeframe='today 12-m', geo='US')

try:

    interest\_data = pytrend.interest\_over\_time()

    plt.figure(figsize=(10, 6))

    plt.title("Interest over Time")

    plt.xlabel("Time")

    plt.ylabel("Interest")

    for keyword in keywords:

        interest = interest\_data[keyword].astype(int)

        plt.plot(interest, label=keyword)

    plt.legend(loc="upper left")

    plt.show()

except Exception as e:

    print(f"Error getting interest over time data: {e}")

Chart, line chart

Description automatically generated

import pandas as pd

from pytrends.request import TrendReq

import matplotlib.pyplot as plt

import networkx as nx

# Create a GitHub graph

G = nx.Graph()

G.add\_node("GitHub")

# Get top 10 trending repositories

trending\_repos = ["facebook/react", "tensorflow/tensorflow", "d3/d3", "angular/material", "pallets/flask", "vuejs/vue", "scikit-learn/scikit-learn", "twbs/bootstrap", "apache/airflow", "axios/axios"]

# Add trending repositories to TrendingRepos node

G.add\_node("TrendingRepos")

for repo in trending\_repos:

    G.add\_node(repo)

    G.add\_edge("TrendingRepos", repo)

# Get interest over time data for the specified keywords

pytrend = TrendReq()

keywords = ["Angular Material", "React", "D3", "Flask", "TensorFlow"]

pytrend.build\_payload(keywords, timeframe='today 12-m', geo='US')

try:

    interest\_data = pytrend.interest\_over\_time()

    plt.figure(figsize=(10, 6))

    plt.title("Interest over Time")

    plt.xlabel("Time")

    plt.ylabel("Interest")

    for keyword in keywords:

        interest = interest\_data[keyword].astype(int)

        plt.plot(interest, label=keyword)

    plt.legend(loc="upper left")

    plt.show()

except Exception as e:

    print(f"Error getting interest over time data: {e}")

Chart, line chart

Description automatically generated

1. Update the **BugHub** Graph Data Model (use **Netowkrx** Package) to attach/add to **StackOverflow** node a new node called **TrendingTags** that has **top-10** Tags that have the most asked questions. (Trending line chart for past 12 months data).