INDIA INTERNATIONAL SCIENCE FESTIVAL 2023

SPACE HACKATHON

In Association with

Team Name: Topper

Name of College/University: IIT Mandi

Team Member Details:

- Yash Shrivastava
- Sania Jain
- Khushal Sharma
- Sujal Arora

Problem Statement:

Explain your understanding on Problem Statement:

- To understand the audience
- Investigate visitor activity
- Improve user experience and optimize performance(the flow).
- Improve Security(Restrict specific IPs or identify used protocols and ports)
- Leverage Technical Analysis.

Brief about your approach:

Handling data preprocessing and performing feature engineering to extract relevant insights. Performing Exploratory Data Analysis (EDA) to understand user interaction patterns. Identifying trends and key behaviors that might be important for the analysis. Developing machine learning models (clustering and traditional prediction models) for behavior analysis and prediction. Develop a real-time recommendations system based on user interactions which can adapt accurately with changing user behavior.

Detailed Proposal & Solution Approach

Step 1 - Parsing the Log file : Returning a dataframe with the user logs data with columns as contents of the common log format. This is done using the apache_log_parser library or using the 're.match()' function of python, both of which match a regular expression pattern(flexible) against a log entry.
Regex pattern is modified to omit certain content if it is empty in most of user logs.

Step 2 - Identifying patterns and making inferences : Using the groupby and matplotlib/seaborn, the data is separately studied for each column or set of columns. Using 'count' function of a pandas dataframe group type, we can analyse all the frequency related data, which are the most/least frequent data points(can be done for individual hosts). Also for (status code, request_line) to get the most frequent request_line for unsuccessful requests, etc.

Step 3 - Analysing End points: Using the dataframe operations for group and functions of datetime datatype we define a session in our data according to certain session end threshold condition (duration between 2 request line). This time limit can be determined by using machine learning models such as agglomerative or more clustering algorithms. This data will help to get us all information about end/drop-off points which is used to get insights and make improvements.

Step 4 - User Segmentation: Segmentation of users using machine learning models, clustering algorithms (like DBSCAN or agglomerative/divisive) and statistical analysis based on their behavior and usage patterns. Targeted services, recommendation systems and feature development for segments based on geographical location and user interest(application sectors of geo-exploration, thematic services).

Detailed Proposal & Solution Approach

Step 5 - Predicting the user behavior: Predicting user behavior using machine learning models. This includes session length, session frequency, next request of the user and resource usage patterns. Validating this behavior by splitting the original data on the basis of session and predicting request lines for each session.

Step 6 - User Agent Analysis: Understanding the diversity of devices and operating systems accessing the application to optimize the platform for feature compatibility, responsive design and performance enhancement. Detect outdated browsers or devices with known security vulnerabilities to guide security measures and ensure secure user experience. Identify bot/automated traffic by analysing user agent.

Step 7 - Anomalies detection: Detect potential security threats in the platform using ML models (Isolation Forest, One-class SVM, Autoencoders) and using statistical methods(Z-Score or Grubb's Test) based on patterns and deviations in data. Monitoring access with unusual resource consumption and from suspicious or unusual IP address/geographical location/user agent.

Step 8 - Recommendation System: Providing personalised recommendations to user using the previous session data (Content-based) or data of users with similar past interest (Collaborative). Hybrid of both works better. And to manage general recommendations for the platform based on all the log data of all users. May incorporate user feedback and interactions to continuously improve recommendation system.

Tools and devices used on development

- Python libraries : ‘pandas’, ‘matplotlib’, ‘seaborn’, ‘datetime’, ‘itertools’
- ML Frameworks : ‘scikit-learn’, ‘tensorflow’, ‘keras’, ‘torch’
- Log Parsing: ‘apache_log_parser’, ‘re’, ‘json’
- User Agent Parsing: ‘us-parser’, ‘user-agents’
- Statistical Analysis: ‘statsmodels’, ‘scipy’

Technologies involved/used

- Data Preprocessing: Normalisation, Feature Extraction
- ML/DL techniques: Regression, Classification, Clustering (DBSCAN, agglomerative/divisive), LSTM
- Anomaly detection: Isolation Forest, One-class SVM, Z-score
- Recommendation Systems Approaches: Content-Based filtering, Collaborative filtering

References/Acknowledgement

- https://github.com/amandasaurus/apache-log-parser
- https://github.com/topics/log-analysis
- https://www.researchgate.net/publication/341876853_Analyzing_and_Simplifying_Log_Files_using_Python
- https://advertools.readthedocs.io/en/master/advertools.logs.html
- https://youtu.be/lAqrXDzF-Tw
- https://chat.openai.com