Assessment Report

"INTERNET USAGE CLUSTERING"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY

DEGREE

SESSION 2024-25

in

CSE(AI)

By

NAME- Ravi Kant Raj

Roll NO. - 202401100300197

Introduction

This project aims to **group internet users into clusters** based on their behavior. We analyze three main features:

- daily_usage_hours: Total time spent using devices daily.
- site_categories_visited: Number of different website categories visited.
- sessions_per_day: Number of distinct usage sessions per day.

Clustering users based on these patterns can help in behavior analysis, personalization, and network resource management.

Methodology

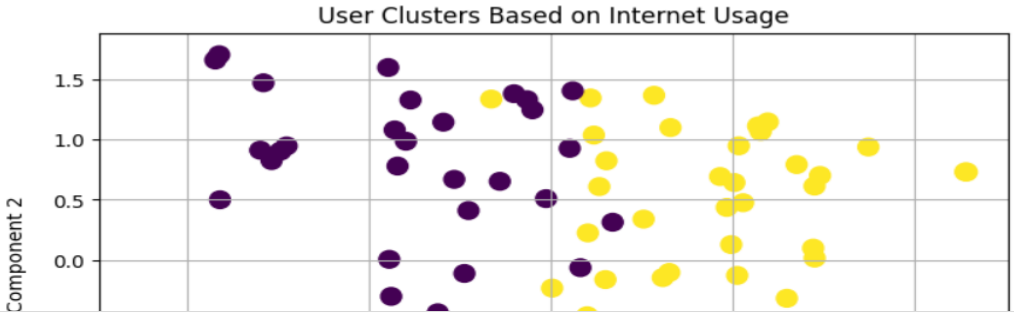
1. **Data Preprocessing**:
    - Load CSV data with relevant user behavior metrics.
    - Normalize the features using StandardScaler for unbiased clustering.
2. **Clustering**:
    - Apply **KMeans clustering** with 2 clusters (can be adjusted).
    - Assign cluster labels to each user.
3. **Visualization**:
    - Use **PCA (Principal Component Analysis)** to reduce the feature space to 2D.
    - Plot the clusters for visual interpretation.

CODE

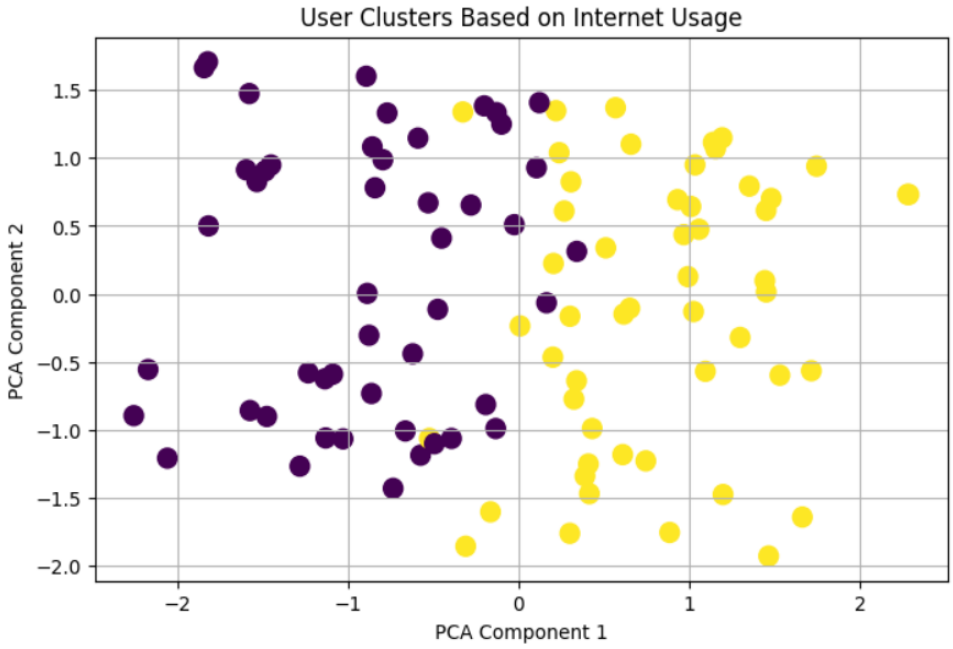
```
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# Load dataset with correct headers
# dataset_path = '/mnt/data/internet_usage.csv'  # Replace with your actual file path
data = pd.read_csv('/content/internet_usage.csv')

# Print dataset to confirm
print("Dataset preview:")
print(data.head())

# Select relevant features
features = data[['daily_usage_hours', 'site_categories_visited', 'sessions_per_day']]

# Scale features
scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)

# KMeans clustering
kmeans = KMeans(n_clusters=2, random_state=0)  # You can change n_clusters as needed
clusters = kmeans.fit_predict(features_scaled)

# Add cluster info to the dataset
data['Cluster'] = clusters
```

```
# Show result
print("\nClustered Data:")
print(data)

# Visualize using PCA
pca = PCA(n_components=2)
reduced = pca.fit_transform(features_scaled)

plt.figure(figsize=(8, 5))
plt.scatter(reduced[:, 0], reduced[:, 1], c=clusters, cmap='viridis', s=100)
plt.title('User Clusters Based on Internet Usage')
plt.xlabel('PCA Component 1')
plt.ylabel('PCA Component 2')
plt.grid(True)
plt.show()
```

```
Dataset preview:
   daily_usage_hours  site_categories_visited  sessions_per_day
0           9.884957                        2                13
1           1.023220                        9                 1
2          10.394205                        9                 3
3           5.990237                        6                16
4           3.558451                        4                 4

Clustered Data:
    daily_usage_hours  site_categories_visited  sessions_per_day  Cluster
0           9.884957                        2                13        0
1           1.023220                        9                 1        1
2          10.394205                        9                 3        1
3           5.990237                        6                16        1
4           3.558451                        4                 4        0
..               ...                      ...               ...      ...
95          3.051110                        4                18        0
96          7.572593                        4                16        0
97          0.299809                        2                 6        0
98          8.648701                        5                13        1
99          6.168280                        4                 4        0

[100 rows x 4 columns]
```



[100 rows x 4 columns]



References/Credits (Elaborated):

1. **Dataset Source**:
   The dataset used for this project, named internet_usage.csv, was created manually based on simulated internet usage behavior data. It includes fields such as daily_usage_hours, site_categories_visited, and sessions_per_day to represent how different users interact with online platforms.
2. **Libraries & Tools**:
   - **Pandas** (pandas): Used for loading, handling, and manipulating tabular data.
     Documentation link
   - **Scikit-learn** (sklearn): Used for data preprocessing (StandardScaler), clustering (KMeans), and dimensionality reduction (PCA).
     Documentation link
   - **Matplotlib** (matplotlib.pyplot): Used to create visualizations like the scatter plot for PCA output.
     Documentation link
3. **Clustering Technique**:
   - The clustering was performed using **KMeans algorithm**, a widely-used unsupervised machine learning method that partitions the data into k clusters based on feature similarity.
   - **PCA (Principal Component Analysis)** was used to reduce high-dimensional data into 2D for visualization purposes without losing much information.
4. **IDE/Environment**:
   - The project was implemented in **Google Colab**, a cloud-based Python notebook environment that supports real-time execution and visualization.
5. **General References**:
   - Scikit-learn User Guide:
     https://scikit-learn.org/stable/user_guide.html
   - Python Data Science Handbook by Jake VanderPlas – for understanding PCA and KMeans concepts.
6. **Credits**:
   - Project done as part of a **data analysis or machine learning coursework** or self-study assignment.

- Visuals and screenshots were captured during runtime to ensure real-time demonstration of cluster patterns.