Analyzing Website Traffic Data

A Comprehensive Report

Prepared by: Satyam Tyagi

Roll Number: 202401100300220

Branch: CSEAI

Section: C

Organization/Institution: KIET Group Of Instituition

INTRODUCTION

In today's digital landscape, understanding website traffic is essential for optimizing online presence and improving user engagement. This report analyzes website traffic data to identify trends, user behavior, and potential areas for improvement.

Objectives

The primary objectives of this analysis include:

- **Understanding overall website traffic patterns.**
- **Identifying peak traffic periods and sources of visitors.**
- **Analyzing user behavior, including time spent on pages and bounce rates.**
- **Providing insights for improving website performance and engagement.**

Scope of the Report

It focuses on key performance indicators such as page views, unique visitors, traffic sources, and user demographics.

By the end of this report, readers will gain a clear understanding of website traffic trends and recommendations to enhance digital strategy.

Methodology

This section outlines the approach used to analyze website traffic data, including data collection, preprocessing, and visualization techniques.

Data Collection

The dataset used in this analysis was sourced from traffic_data.csv, which contains website traffic metrics such as:

- **Page Views – Total number of pages viewed by visitors.**
- **Unique Visitors – Number of distinct users visiting the website.**
- **Bounce Rate – Percentage of visitors who leave after viewing only one page.**

Data Preprocessing

Before analysis, the dataset was cleaned and formatted as follows:

1. **Loading the Data: The dataset was read into a Pandas DataFrame using pd.read_csv().**
2. **Date Handling: The "Date" column was converted to datetime format for proper time series analysis.**
3. **Indexing: The "Date" column was set as the index to facilitate time-based plotting.**
4. **Handling Missing Values: Potential missing values were checked and addressed where necessary.**

Data Analysis & Visualization

To identify trends and patterns, various visualizations were created:

- **Time Series Analysis:**
    - **A line chart was plotted to observe trends in Page Views and Unique Visitors over time.**
- **Correlation Analysis:**
    - **A heatmap was generated to examine relationships between numerical variables such as Page Views, Unique Visitors, and Bounce Rate.**
- **Scatter Plot Analysis:**

- **A scatter plot was used to visualize the relationship between Page Views and Bounce Rate, helping to understand user engagement.**

Tools & Libraries Used

The following Python libraries were utilized for analysis and visualization:

- **Pandas – For data manipulation and preprocessing.**
- **Matplotlib & Seaborn – For creating time series plots, heatmaps, and scatter plots.**

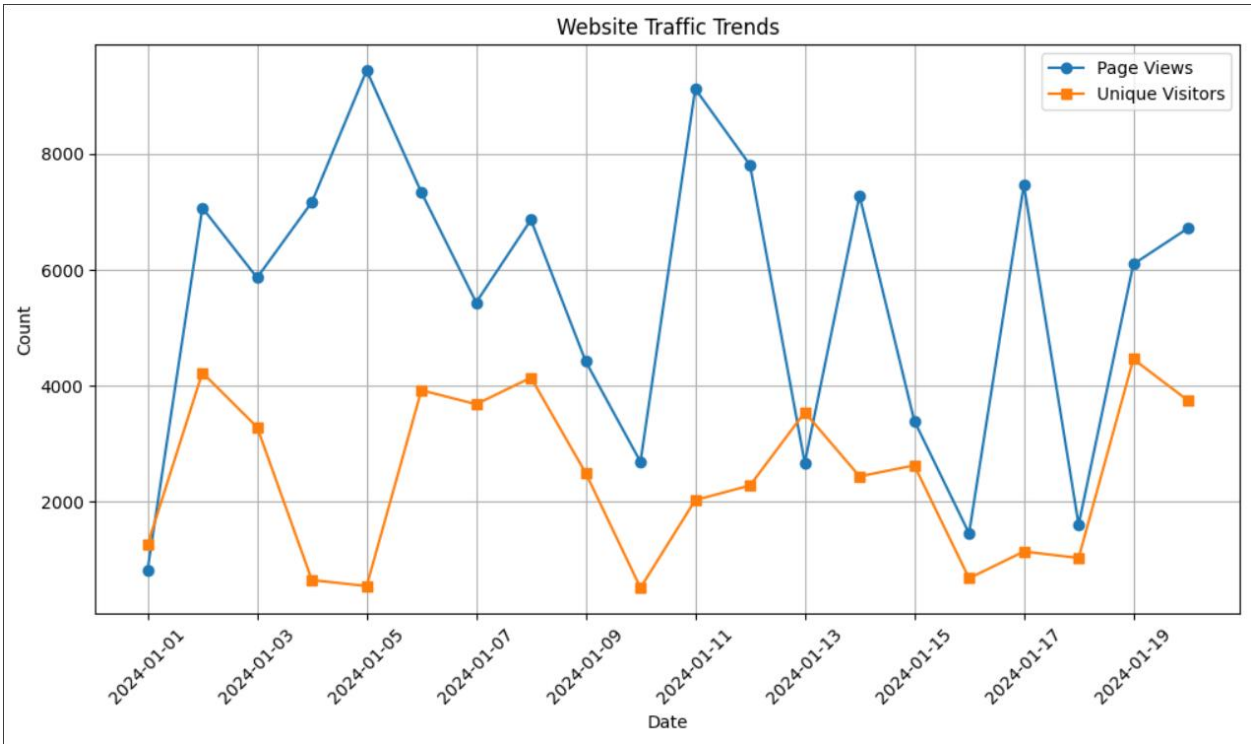
This methodology ensures a structured and insightful approach to understanding website traffic behavior and user engagement.

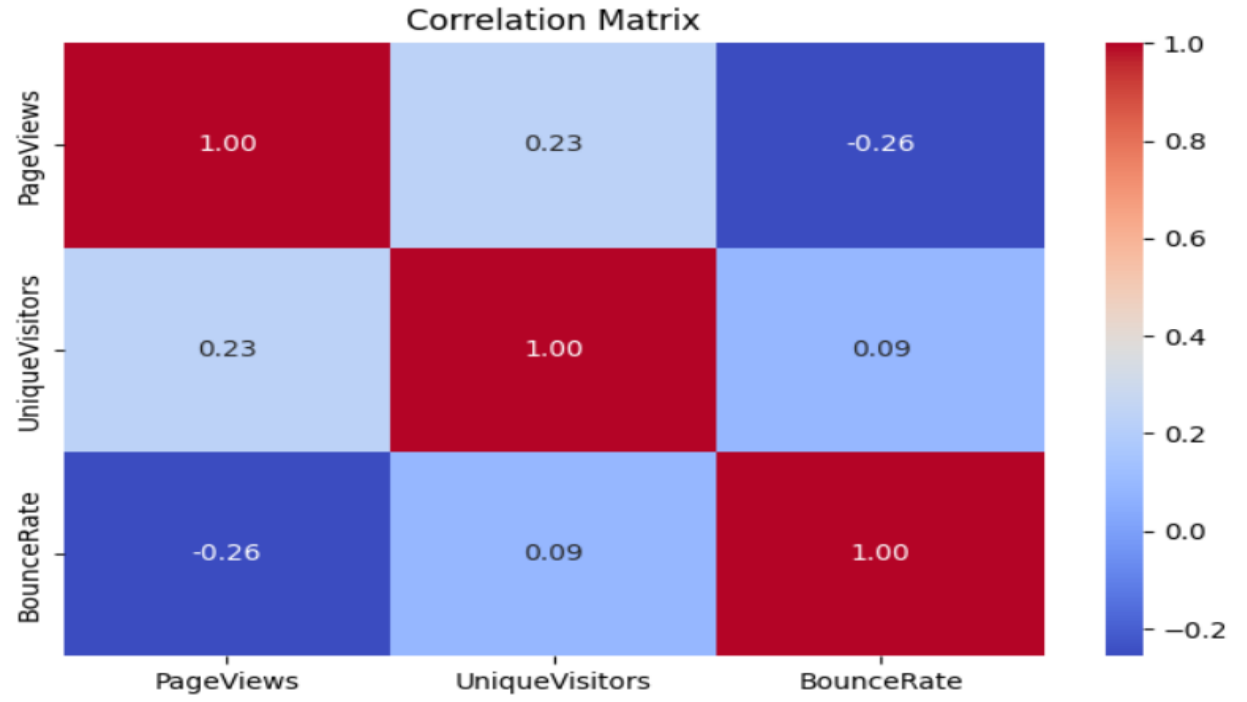
CODE TYPED

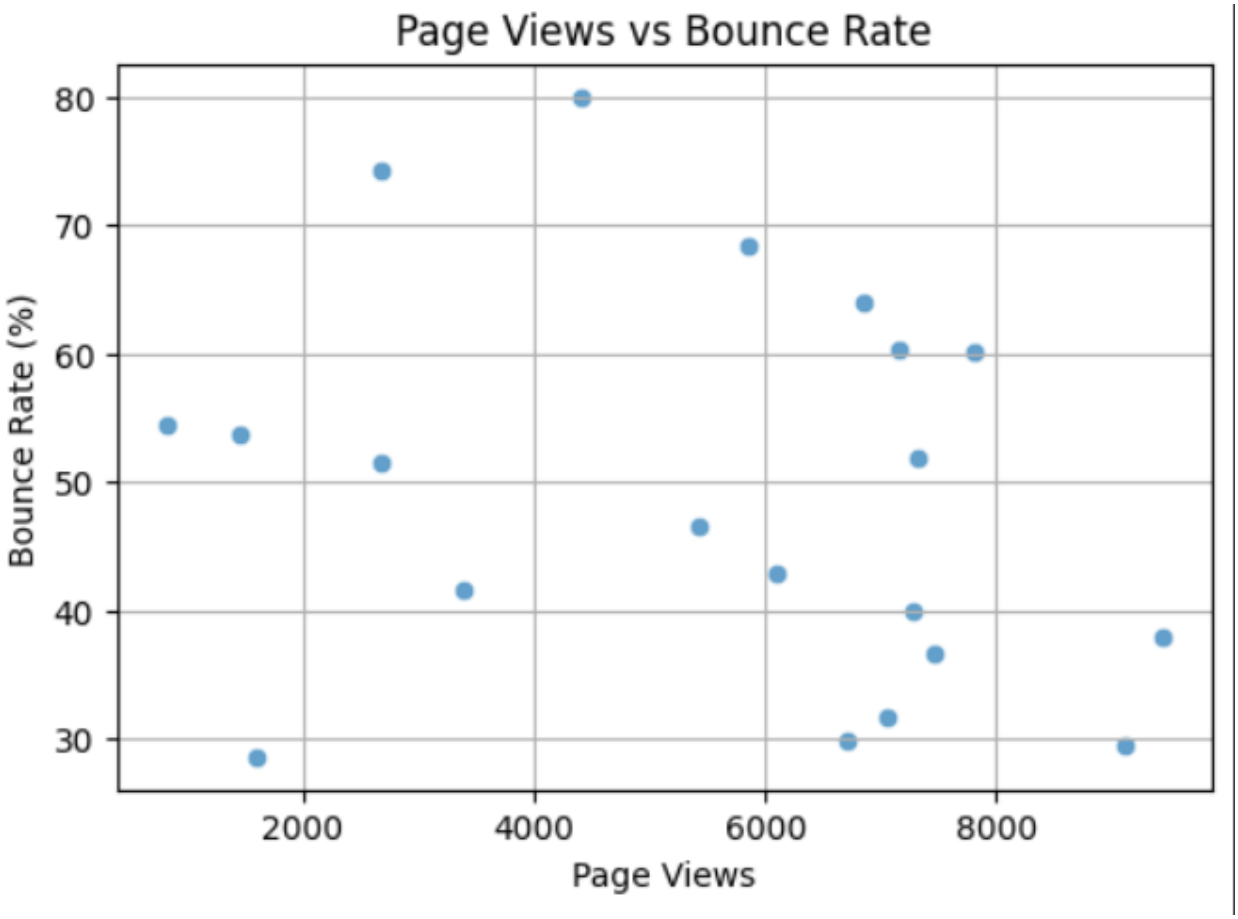
```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns


# Load the dataset

file_path = "/content/traffic_data.csv"

df = pd.read_csv(file_path)


# Convert Date column to datetime format

df["Date"] = pd.to_datetime(df["Date"])


# Set Date as index

df.set_index("Date", inplace=True)


# Plot trends over time

plt.figure(figsize=(12, 6))

plt.plot(df.index, df["PageViews"], marker='o', label="Page Views")

plt.plot(df.index, df["UniqueVisitors"], marker='s', label="Unique
Visitors")

plt.xlabel("Date")

plt.ylabel("Count")
```

```
plt.title("Website Traffic Trends")
plt.legend()
plt.xticks(rotation=45)
plt.grid()
plt.show()

# Correlation heatmap
plt.figure(figsize=(8, 5))
sns.heatmap(df.corr(numeric_only=True), annot=True,
cmap="coolwarm", fmt=".2f")
plt.title("Correlation Matrix")
plt.show()

# Scatter plot of Page Views vs Bounce Rate
plt.figure(figsize=(6, 4))
sns.scatterplot(x=df["PageViews"], y=df["BounceRate"], alpha=0.7)
plt.xlabel("Page Views")
plt.ylabel("Bounce Rate (%)")
plt.title("Page Views vs Bounce Rate")
plt.grid()
plt.show()
```

OUTPUT SCREENSHOT






Page Views vs Bounce Rate
Bounce Rate (%)
80
70
60
50
40
30
2000
4000
6000
8000
Page Views