Facebook Marketplace Data Analysis

Assignment Summary & Insights

Dataset Overview

- Facebook Live Sellers data from Thailand
- Data includes reactions, comments, shares, types of posts, etc.

CONCEPT OF CORRELATION AND STEPS TO FIND CORRELATION MATRICES

- First, I converted the 'Status Published' column from text format to datetime format and created separate columns for date, hour, and year using methods from the datetime library
- Next, I removed all unnecessary and null columns to make the dataset more concise and suitable for modeling and analysis. I then checked whether any null values were still present in the remaining columns. Then correlation matrix between different columns is shown in the following slides.

Following this, a correlation matrix between different columns is presented in the subsequent slides.

The concept behind the correlation values is as follows: a correlation value between 0 and 1 indicates a positive correlation, a value between -1 and 0 indicates a negative correlation, and a value of 0 indicates no correlation.

- The correlation between time of upload (status_published) and the num_reaction : 0.017016 (**It reprents small positive correlation**).

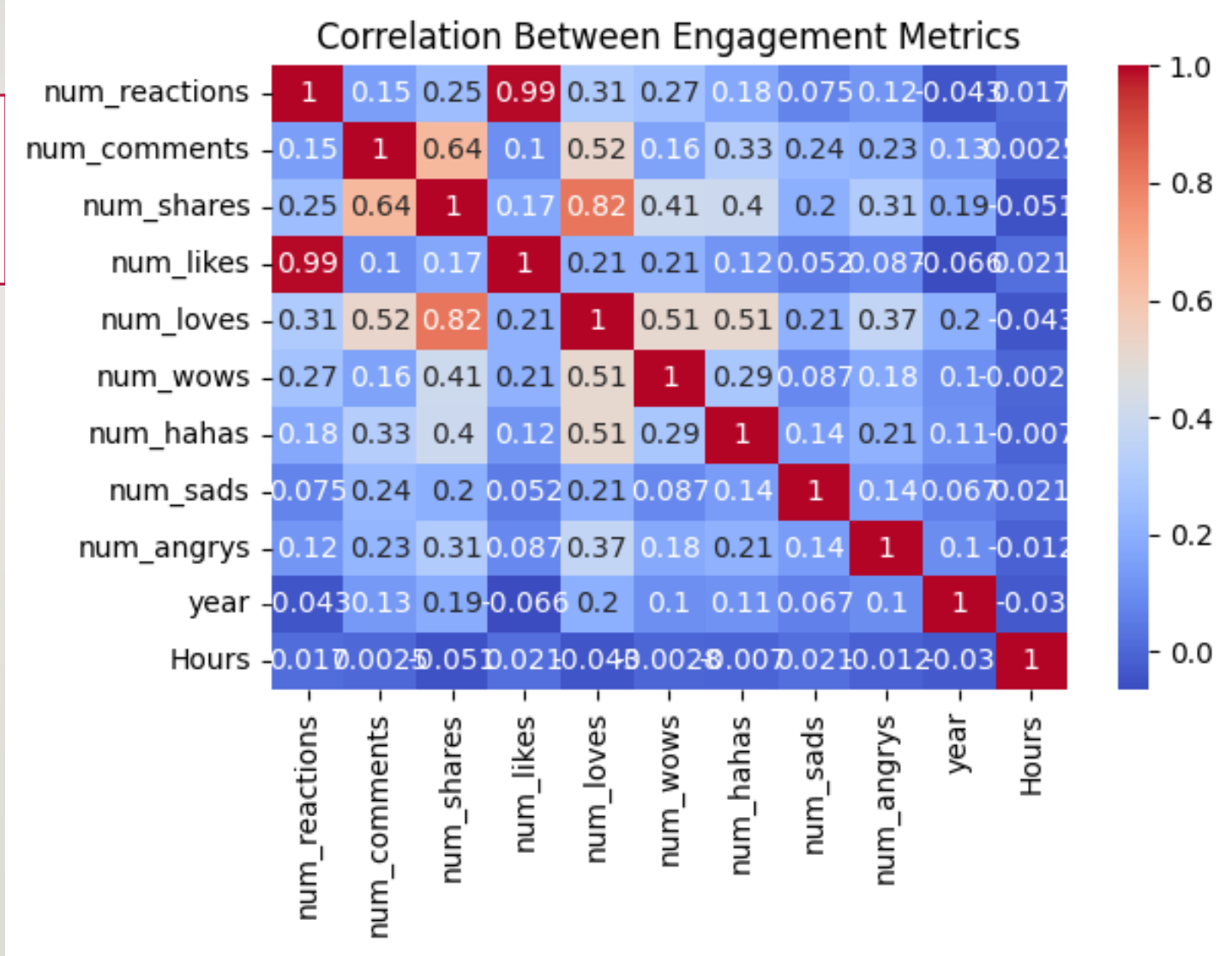
	Hours	num_reactions
Hours	1.000000	0.017016
num_reactions	0.017016	1.000000

- correlation between the number of reactions (num_reactions) and other engagement metrics such as comments (num_comments) and shares (num_shares) are:

	num_reactions	num_comments	num_shares	num_likes	num_loves	num_wows	num_hahas	num_sads	num_angrys	year	Hours
num_reactions	1.000000	0.150843	0.250723	0.994923	0.305003	0.267752	0.176028	0.075138	0.124326	-0.042703	0.017016
num_comments	0.150843	1.000000	0.640637	0.101687	0.521223	0.162394	0.325048	0.236453	0.225184	0.132399	0.002515
num_shares	0.250723	0.640637	1.000000	0.172492	0.820000	0.407628	0.399826	0.199970	0.312513	0.189782	-0.050917
num_likes	0.994923	0.101687	0.172492	1.000000	0.209308	0.207800	0.120784	0.052169	0.087431	-0.065528	0.021375
num_loves	0.305003	0.521223	0.820000	0.209308	1.000000	0.508798	0.507830	0.207600	0.371001	0.204702	-0.042705
num_wows	0.267752	0.162394	0.407628	0.207800	0.508798	1.000000	0.287756	0.086503	0.183087	0.101530	-0.002816
num_hahas	0.176028	0.325048	0.399826	0.120784	0.507830	0.287756	1.000000	0.141421	0.211910	0.113236	-0.006964
num_sads	0.075138	0.236453	0.199970	0.052169	0.207600	0.086503	0.141421	1.000000	0.142072	0.067446	0.020918
num_angrys	0.124326	0.225184	0.312513	0.087431	0.371001	0.183087	0.211910	0.142072	1.000000	0.100654	-0.012327
year	-0.042703	0.132399	0.189782	-0.065528	0.204702	0.101530	0.113236	0.067446	0.100654	1.000000	-0.030198
Hours	0.017016	0.002515	-0.050917	0.021375	-0.042705	-0.002816	-0.006964	0.020918	-0.012327	-0.030198	1.000000

HEATMAP OF CORRELATION MATRICES

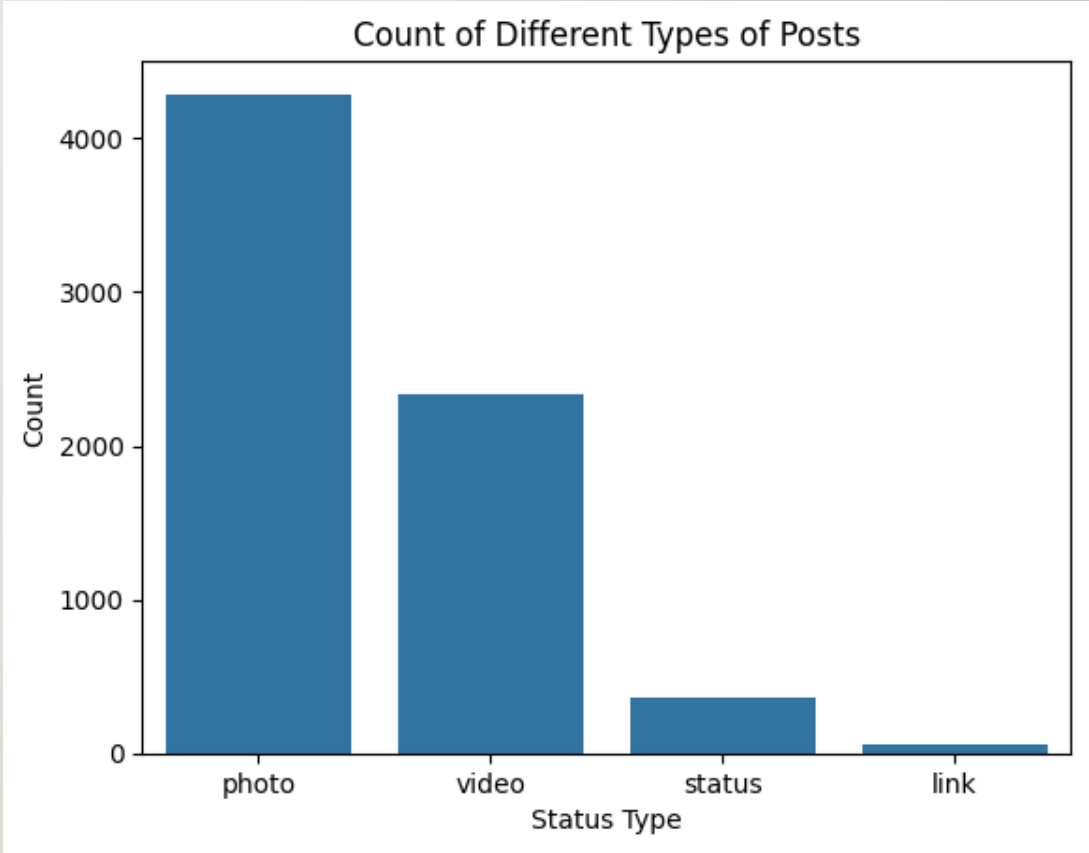
Correlation value close to 1 represents higher positive correlation. The diagonal value of the matrix always 1 because of same column in x and y axis.
And the colour bar represents the weightage of the correlation in the heatmap representation.



Count of different types of post and average of the required columns

- The count of different types of posts in the dataset are:

status_type	count
photo	4288
video	2334
status	365



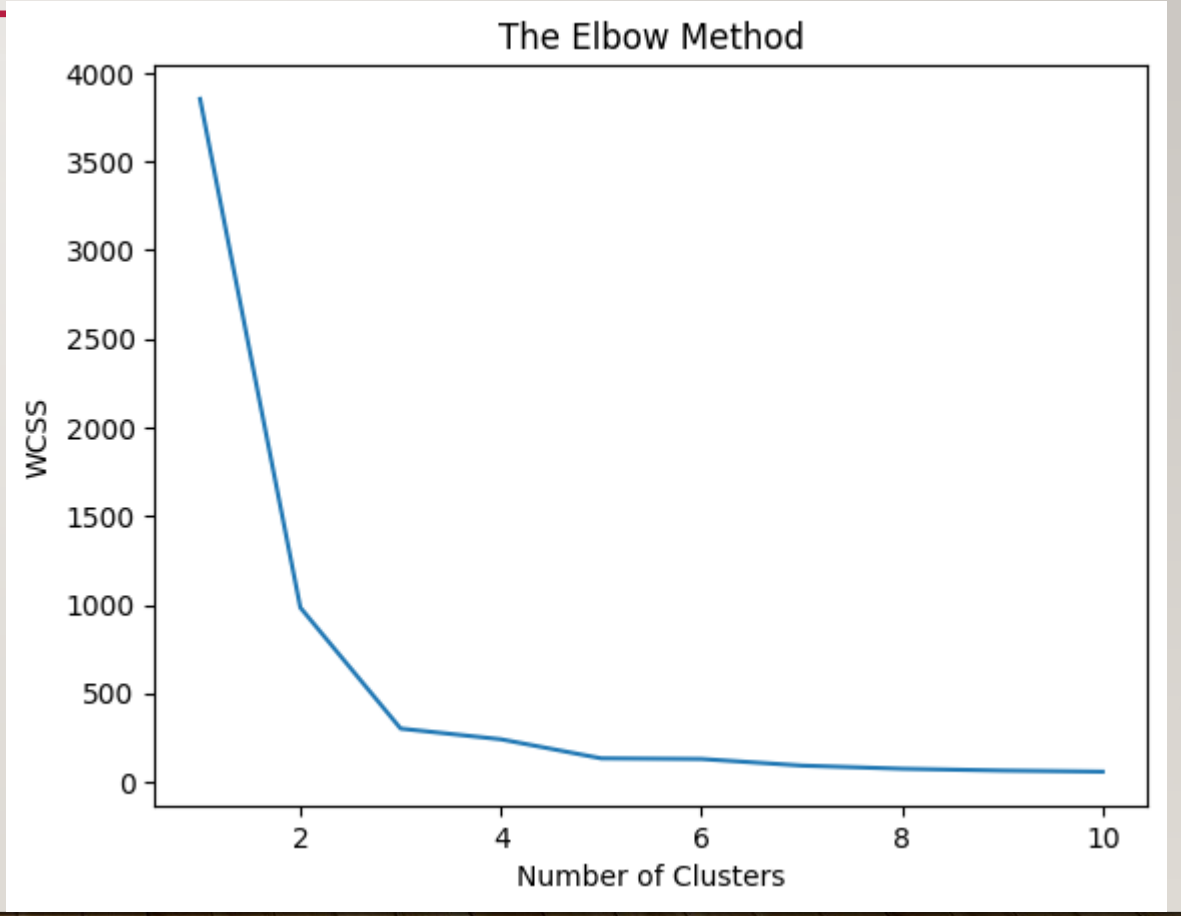
- The average value of num_reaction, num_comments, num_shares for each post type are **230.11716312056737, 224.3560283687943 and 40.022553191489365** respectively.

DATA PREPROCESSING FOR THE K MEANS CLUSTERING MODEL

- Since categorical values cannot be directly used in machine learning models, all categorical columns must be converted into numerical representations that the models can interpret. To achieve this, I used the OneHotEncoder from the sklearn library to transfer the status type categorical variable to numerical variable..
- Next, I applied feature scaling using the MinMaxScaler class, which normalizes the feature values to a common scale. This step is crucial to eliminate bias caused by features with larger ranges dominating the learning process.
- Important: Do not include the columns generated by OneHotEncoder in the feature scaling process."

Steps to find elbow curve

- The Elbow Method is used to determine the optimal number of clusters (K) for clustering analysis. It relies on the Within-Cluster Sum of Squares (WCSS), which measures the compactness of the clusters. A lower WCSS value indicates more accurate and well-defined clustering.
  Choosing the right value of K is important
- If K is too small, distinct clusters may be grouped together, leading to poor classification.
- If K is too large, it may overfit the data by assigning each point to its own cluster, increasing computational cost unnecessarily.
  To identify the optimal K, I ran a loop for K ranging from 1 to 10. Based on the WCSS vs. number of clusters graph, the 'elbow point' suggests that K = 5 is the most suitable number of clusters for this dataset



RESULTS OF CLUSTERING AND GRAPHICAL REPRESENTATION

- After selecting K = 5, the dataset was successfully classified into five distinct clusters based on the selected features.
- I used random_state = 42 to ensure that the clustering process produces consistent results each time it is run, while still introducing randomization to avoid any bias. The cluster labels were stored in a new column named y_means.
- To visualize the clustering results, I plotted three different curves using selected feature columns. These visualizations are shown in the following slide.

Fig.1)num_loves vs num_wows
Fig. 2)num_reactions vs num_comments
Fig .3) num shares vs num_likes

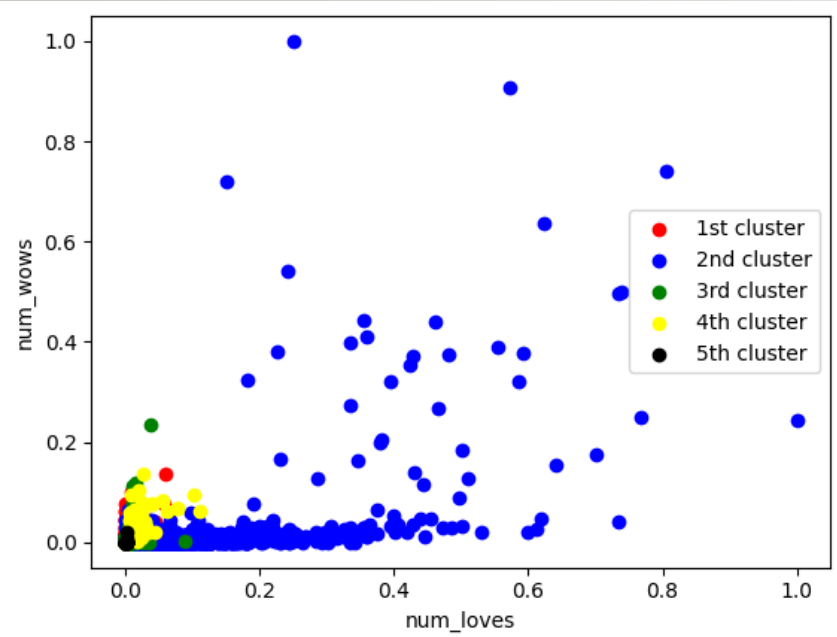


Fig.1

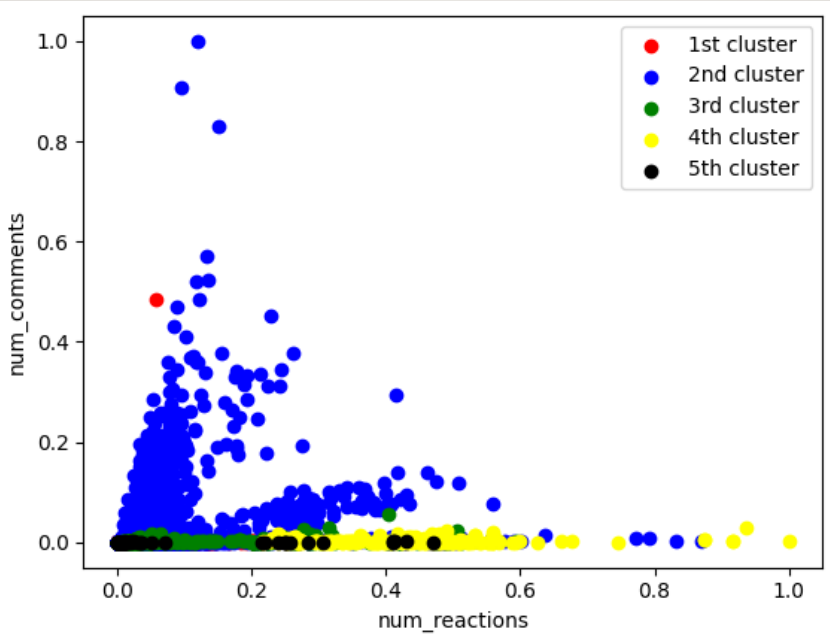


Fig.2

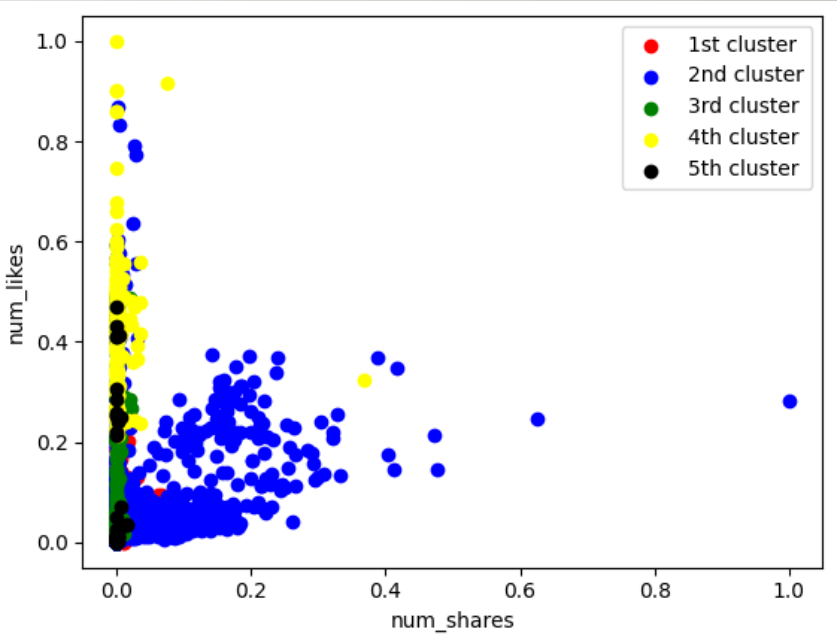


Fig.3