



PROJECT REPORT

IRS5001 INTELLIGENT REASONING SYSTEMS

PostCraft: X Tweet Features Recommendation and Text Generation System

Group 9

Liu Siyan (A0285857H)
Lin Zijun (A0285897Y)
Lai Weichih (A0285875H)
Fang Ruolin (A0285983H)

Abstract

With the rise of social media platforms, the landscape of digital marketing has undergone significant transformations. More and more brands and individuals are utilizing social media for both product or service promotion. Platforms like X have become crucial spaces for businesses to showcase their products and engage with potential customers. However, the overcrowded social media space makes it challenging for brands to stand out and effectively reach their target audience. While various data analytics tools are available in the current market, they still require experienced professionals for analysis and creativity in marketing, thereby increasing human resource and time costs. Moreover, existing solutions might not fully meet the unique and real-time promotional needs faced by different good categories in the highly competitive and ever-changing environment.

To address this market pain, we have developed PostCraft, a data-driven tweet feature and text recommendation and generation system. The system aims to provide efficient and intelligent marketing solutions for both individuals and businesses. Postcraft comprises three main components: Tweets Feature Recommender, Image Content Recommender, and Tweet Optimizer. This system integrates multiple machine learning algorithms, NLP techniques, various recommendation algorithms, and LLM tools to establish a comprehensive platform. By analyzing current market trends and key components of popular posts, Postcraft facilitates tweet feature recommendation, tweet content optimization and generation, as well as image content recommendations for tweets.

In terms of performance validation, we sent out 180 tweets from 100 accounts, with 90 generated by PostCraft and 90 by ChatGPT. We use viewing count within a 3-day period as metrics. Using a t-test and assuming equal means for both methods' outcomes, at a significance level of 0.05, we obtained a p-value of 0.0346 and a t-statistic of 2.1293. Therefore, we reject the hypothesis and conclude that tweets generated by PostCraft significantly outperformed those generated by ChatGPT. This finding strongly supports the outstanding performance of our system in social media tweet generation.

PostCraft has been deployed on a website, enabling users to conveniently access it online. It

utilizes Flask as the application framework, HTML/CSS/JavaScript for frontend development to create an intuitive user interface, and Python as the backend to generate recommendations requested through Flask.

Contents

| | | |
|----------|-------------------------------------------------------------------------|-----------|
| 1 | Introduction | 1 |
| 1.1 | Project Background | 1 |
| 1.1.1 | Project Objective | 1 |
| 1.2 | System Description | 2 |
| 1.2.1 | Team Members | 2 |
| 2 | Business Justification | 4 |
| 2.1 | User Hierarchy Analysis | 6 |
| 2.2 | Competitive Product Research | 7 |
| 2.3 | SWOT Analyse | 8 |
| 3 | System Design | 10 |
| 3.1 | system architecture overview | 10 |
| 3.2 | System Feature | 11 |
| 3.3 | frontend design | 11 |
| 3.4 | Backend design | 16 |
| 3.4.1 | Tweets Feature Recommender | 16 |
| 3.4.2 | Image Content Recommender | 18 |
| 3.4.3 | Tweet Optimizer | 20 |
| 4 | Model | 21 |
| 4.1 | Natural Language Process | 21 |
| 4.1.1 | Word2Vec | 21 |
| 4.1.2 | TF-IDF | 24 |
| 4.2 | Apriori based association rule mining | 24 |
| 4.2.1 | Data discretization | 24 |
| 4.2.2 | Text features recommendation based on association rule mining | 25 |
| 5 | Dataset Processing | 27 |

| | | |
|-----------|--------------------------------------------------|-----------|
| 5.1 | Dataset | 27 |
| 5.1.1 | Data Collection | 27 |
| 5.1.2 | Data Description | 27 |
| 5.2 | Data Processing | 27 |
| 5.2.1 | Data Cleaning | 28 |
| 5.2.2 | Text Preprocessing | 28 |
| 5.2.3 | Post Clustering | 29 |
| 5.2.4 | Interactive Indicators | 29 |
| 6 | Demonstration and Test | 34 |
| 6.1 | Usage Demonstration | 34 |
| 6.1.1 | Tweet Optimizer | 34 |
| 6.1.2 | Tweets Feature Recommender | 34 |
| 6.1.3 | Image Content Recommender | 35 |
| 6.2 | Practical Test and Result | 35 |
| 6.2.1 | Test Method | 36 |
| 6.2.2 | Result | 36 |
| 7 | Conclusion and Future Work | 38 |
| 8 | Appendix A: Project Proposal | 39 |
| 9 | Appendix B: Mapped System Functionalities | 42 |
| 10 | Appendix C: Installation and User Guide | 43 |
| 10.1 | Introduction | 43 |
| 10.2 | Browse/Install | 43 |
| 10.3 | Overview | 43 |
| 10.4 | Services | 46 |
| 10.4.1 | Part 1 | 46 |
| 10.4.2 | Part 2 | 48 |

| | |
|-------------------------------------------------|-----------|
| 10.4.3 Part 3 | 49 |
| 11 Appendix D: Individual Project Report | 51 |
| 12 Literature Cited | 57 |

List of Figures

| | | |
|----|---------------------------------------------------------------------------------|----|
| 1 | Social Media Users Over Time[1] | 4 |
| 2 | Monetizable Daily Active Usage of X[2] | 6 |
| 3 | X user K-means elbow map and cluster map | 7 |
| 4 | System Design | 10 |
| 5 | Frontend SD | 12 |
| 6 | introduction page1 | 14 |
| 7 | introduction page2 | 14 |
| 8 | Tweets Feature Recommender1 | 15 |
| 9 | Tweets Feature Recommender2 | 15 |
| 10 | Image Content Recommender | 16 |
| 11 | Tweet Optimizer | 16 |
| 12 | Tweets Features Recommender's flow | 18 |
| 13 | Image Content Recommende's flow | 19 |
| 14 | TweetOptimizer's flow | 20 |
| 15 | CBOW model architecture diagram for single word | 21 |
| 16 | CBOW model architecture diagram for multiple word | 22 |
| 17 | Enter Caption | 23 |
| 18 | X Crawler Workflow | 27 |
| 19 | Text Preprocessing | 28 |
| 20 | X user K-means elbow map and cluster map | 29 |
| 21 | The correlation between the number of comments, retweets, likes and views . . . | 30 |
| 22 | Distribution of Log(Interation index) | 31 |
| 23 | Result of Tweet Optimizer | 34 |
| 24 | Result of Tweets Features Recommender | 35 |
| 25 | Result of Image Content Recommender | 35 |
| 26 | Histogram of Two Method | 37 |
| 27 | System Flowchart | 41 |

| | | |
|----|--------------------------------------------------|----|
| 28 | Home page | 44 |
| 29 | Features section | 45 |
| 30 | Recommendation based on features input | 46 |
| 31 | Services section 1 | 47 |
| 32 | NEW MODEL | 48 |
| 33 | Picture recommendation | 49 |
| 34 | Recommendation based on given text | 50 |

List of Tables

| | | |
|---|--------------------------------------------------------------------------|----|
| 1 | Contributions of Team Members | 3 |
| 2 | Comparison of features across different social media platforms | 7 |
| 3 | Discrete Transformation of Features | 25 |
| 4 | Feature Labels of Raw Data | 32 |
| 5 | Feature Labels of Raw Data | 33 |
| 6 | Mapped System Functionalities | 42 |

1 Introduction

1.1 Project Background

In the field of digital marketing, the rise of social media platforms has provided businesses with a stage to showcase themselves on platforms like X and RedBook, making interaction with potential customers more convenient. However, the crowded social media landscape presents challenges for brands in breaking through information barriers and reaching their target audience accurately. The effectiveness of advertising posts is influenced by various factors such as content length, image quality, tag selection, and the timing of publication. Brands need to identify the optimal combination of these factors when devising their promotional strategies.

In the context of different platform-specific search mechanisms and user preferences, the impact of various factors on the performance of different content posts varies at different times. One of the primary challenges brands face in their promotional campaigns is determining the best combination of article features to ensure the success of their campaigns and to guide their content creation strategies. However, due to a lack of real-time, data-driven insights, this task has become increasingly difficult.

While there are various advertising platforms and analytics tools in the market that provide an overview of audience engagement and ad performance, they often lack personalized, real-time recommendation systems to help brands optimize their advertising strategies on social media platforms. Additionally, existing solutions do not fully cater to the unique needs of brands in the competitive and ever-changing social media environment.

This project aims to address these challenges by developing an intelligent recommendation system, filling a gap in the market and providing customized advertising solutions for brands.

1.1.1 Project Objective

This project aims to develop an intelligent recommendation system that leverages data analysis, machine learning, and natural language processing techniques to provide intelligent and personalized marketing solutions for individuals and businesses. It is designed to address the

challenges of product promotion on social media, including information overload, intense competition, and the need for real-time responses.

1.2 System Description

This project aims to fill an existing market gap by developing an intelligent recommendation system. The system is designed to provide in-depth market analysis for companies, brands, or individuals looking to promote their products. By analyzing popular posts and their key components in the current market, it will offer clients writing recommendations or generate promotional content. This platform will serve as a valuable resource for writing suggestions or automatically generating promotional articles.

Additionally, for multi-channel network (MCN) companies, this project will offer detailed analysis of promotional posts on the X platform (formerly known as Twitter). By delving into the effectiveness of posts by affiliated influencers, it will provide guidance to optimize their advertising strategies.

1.2.1 Team Members

| Full Name | Student ID | Work Items |
|-------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Liu Siyan | A0285857H | Model Design and construction, Back-end development and Integration, Data collation, cleansing and preliminary analysis, Practical validation, Documentation and Figures |
| Lin Zijun | A0285897Y | Front end development, Front end and back end Integration, Deployment, User Guide |
| Lai Weichih | A0285875H | Data collection and crawling, Back-end development, Documentation |
| Fang Ruolin | A0285983H | Business analysis, Data collection and crawling, Documentation |

Table 1: Contributions of Team Members

2 Business Justification

As of now, social media has successfully penetrated half of the global population of 7.7 billion. Over the past decade, the user base of social networking platforms has experienced a nearly threefold increase, growing from 970 million in 2010 to 4.48 billion in July 2021[3].

In Figure 1, we have charted the monthly active users of various platforms since 2004. Some major social media websites, such as Facebook, YouTube, and Reddit, have been around for a decade or more, while other major sites are relatively new. For instance, TikTok was launched in September 2016, and by mid-2018, it had already reached 500 million users. To put this into perspective, during this period, TikTok was adding an average of about 20 million new users per month[1]. Although there have been cases of individual platforms declining, most surviving social media platforms have undergone significant changes in the content they provide to users. For example, initially, Twitter did not allow users to upload videos or images. However, since 2011, this has become possible, and today, over 50 percent of the content on Twitter includes images and videos.

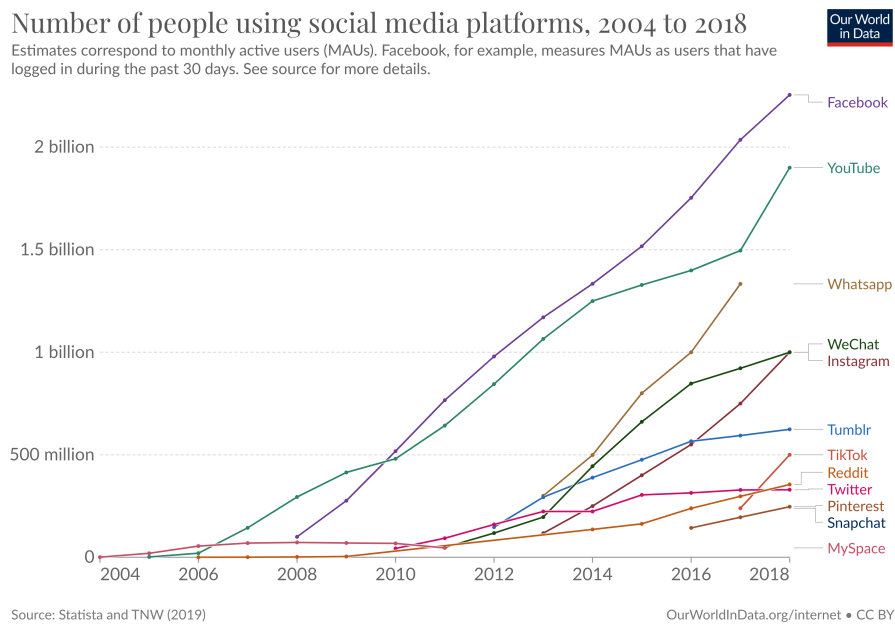


Figure 1: Social Media Users Over Time[1]

In recent trends, we have noticed that social media has gradually become the preferred

platform for people to seek information, learn, interact with others, shop, and seek customer support. For Generation Z, social media has even replaced search engines. This means a vast marketing field for brands[4]. Whether it is through their own accounts, promotion by top opinion leaders, or potential promotion by grassroots opinion leaders, all are visible marketing methods. At the same time, a large number of traditional industries trying to enter internet marketing also need to strengthen their experience in operating on social media platforms, and the considerable human and time costs involved are not negligible, especially in the context of more and more brands and individuals flooding into social media. This growing trend has led to a strong demand: to learn how to conduct high-quality product marketing on social media in a low human and low time cost way.

There is evidence to suggest that high-quality content is the primary factor that captures people’s attention when they scroll through social media. For marketers seeking the best return on investment in social media, learning how to identify quality bloggers and tweets in specific areas and create content that resonates with people is particularly important, as this represents a significant potential commercial value.

With over 175 million accounts worldwide, Twitter is one of the most widely used and closely monitored social media platforms in the world[2]. Twitter stands out from other social media platforms with its unique features and advantages that cater to the needs of both individual users and businesses. The platform’s real-time nature ensures that users have access to the latest news and information, making Twitter a crucial source for current events. Its 280-character limit promotes concise and easy-to-understand content, facilitating quick information digestion for users. Therefore, we have chosen X (formerly known as Twitter) as our analysis subject to provide relevant data for the construction of our system. As shown in Figure 2, since 2020, X platform’s Daily Active Users (DAU) in its top 10 markets have experienced a double-digit year-over-year growth, highlighting its tremendous potential for development. For brands, X platform is undoubtedly an ideal promotional platform that can effectively enhance brand awareness and achieve market coverage.

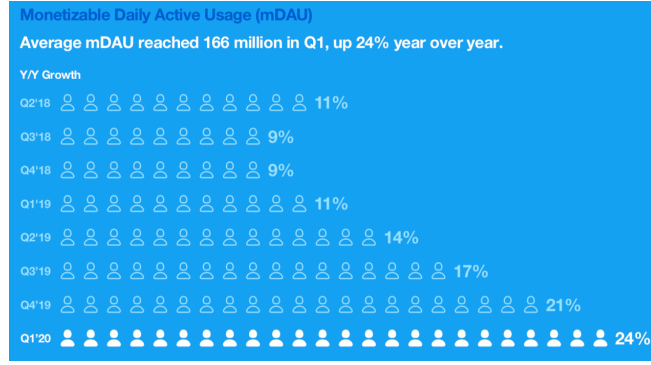


Figure 2: Monetizable Daily Active Usage of X[2]

In today’s highly competitive market environment, brands need to rely on precise advertising strategies to attract and retain the attention of their target audience. While existing advertising platforms and analytical tools can provide macro analysis of audience engagement and advertising performance, they often lack the ability to provide personalized, real-time recommendations to help brands optimize their promotional strategies in the ever-changing social media environment.

To fill this market gap, our project will focus on developing an intelligent recommendation system. This system will utilize advanced data analysis technology, combined with machine learning, NLP technology, and various recommendation system algorithms, to deeply mine the vast data of X platform posts, accurately identify the key factors that affect the success of publicity, and provide real-time, targeted publicity strategy recommendations for brands. This will help brands stand out on the competitive social media platform and maximize their commercial value.

2.1 User Hierarchy Analysis

In order to gain a deeper insight into market trends, we have chosen X (formerly known as Twitter) as our analysis subject. By conducting K-means clustering analysis on the browsing and follower counts of X platform users (Fig. 3), we have categorized users into four distinct clusters. Cluster 0 includes ordinary users with lower follower and browsing counts; Cluster 1 typically refers to KOLs (Key Opinion Leaders) who possess a large follower base and high, stable browsing counts; Cluster 2 encompasses users with a moderate number of followers but not high

browsing counts; and Cluster 3 represents those users who, despite having fewer followers, are able to attract extremely high browsing counts. This indicates that even with a limited number of followers, it is possible to create "hit products" in the market.

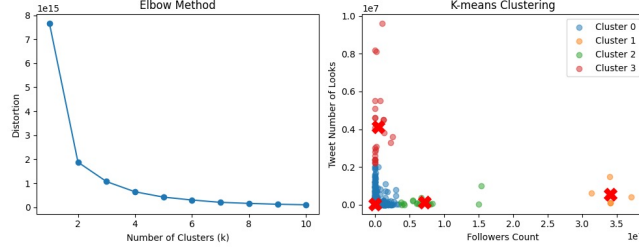


Figure 3: X user K-means elbow map and cluster map

2.2 Competitive Product Research

While conducting an in-depth analysis of the market’s user hierarchy, we also carried out a systematic research on the competitors in the industry (Table 2).

Table 2: Comparison of features across different social media platforms

| Platform | Function | | | |
|----------------------|-----------------|----------------|--------------------|-----------------------------|
| | HISTORICAL DATA | IMAGE ANALYSIS | HASHTAG MONITORING | Personalized recommendation |
| Our system PostCraft | Yes | Yes | Yes | Yes |
| BrandMentions | Yes | No | Yes | No |
| Brandwatch | No | Yes | Yes | No |
| Truescope | Yes | No | Yes | No |

From the table, it is evident that PostCraft is equipped with all four functionalities, showcasing its multifaceted nature and comprehensiveness. Our system’s diverse range of features is designed to meet the varied needs of our clientele, thereby offering a holistic suite of services.

Conversely, BrandMentions lacks the "IMAGE ANALYSIS" feature, which could potentially indicate a shortfall in its proficiency in the realm of image analysis. This absence of service might result in the loss of market share, as it fails to cater to customers in need of image analysis. Similarly, Brandwatch does not support the "HISTORICAL DATA" feature, which might reflect a deficiency in mining historical data. Additionally, the absence of the "Personalized

recommendation” feature could hinder its ability to provide tailored advice to its customers. Truescope, too, does not offer the ”IMAGE ANALYSIS” and ”Personalized recommendation” features, which could limit its service offerings.

2.3 SWOT Analyse

Based on the above research, we conducted a SWOT analysis of our system, PostCraft.

Strengths:

1. In-depth analysis of market dynamics and user behavior, especially focusing on grassroots Key Opinion Leaders (KOLs), ensures that our brand promotion strategy is widely applicable, better meeting the needs of different customers.
2. Adoption of advanced multimodal input technology, which can handle not only text data but also analyze image information, providing customers with comprehensive analysis results to meet their diverse needs.
3. The system integrates real-time analysis functions in the interface, which can display various factors affecting article quality in the current time period, helping customers better grasp market trends and make wiser decisions.
4. Focus on multi-dimensional information, providing customers with accurate and effective decision support by capturing and utilizing various data, thereby helping them stand out in the competitive market environment.
5. Strong ability to draw inferences from one instance, intelligently associating more relevant and popular keywords based on customer-provided keywords, providing customers with rich market information to better seize market opportunities.

Weaknesses: The system currently focuses mainly on text output in content generation to ensure optimal article quality, so it has not yet achieved the generation of tweets with both text and images.

Opportunities: In the current market environment, there is no system that can perfectly integrate text and images to provide tweet recommendations. Therefore, our goal is to break through existing technical limitations, become a pioneer in this field, and provide users with a

unique service experience.

Threats: The forms of tweets on social platforms are constantly evolving, from the initial pure text form to the current multimedia display with both text and images. This rapid change requires us to maintain an agile update pace to keep up with market trends, continuously meet user needs, and occupy a favorable position in the fierce market competition.

3 System Design

3.1 system architecture overview

This system is designed to target the businessmen, men companies, bottom kol or ordinary people who want to promote their products through tweets, etc., and is used for the recommendation of tweeted articles for product promotion and related decision-making suggestions.

In this project, three main functions are implemented to recommend text and features to the user, and a beautiful web interface is designed to interact with the user. The front-end mainly uses HTML, CSS, JavaScript, the back-end is mainly developed using python, as for the entire project framework using flask to integrate the front-end and back-end. The database part is mainly divided into three parts data for cold start, hot start data, and cold data after data processing for image features recommendation. Figure 4 is the system structure diagram:

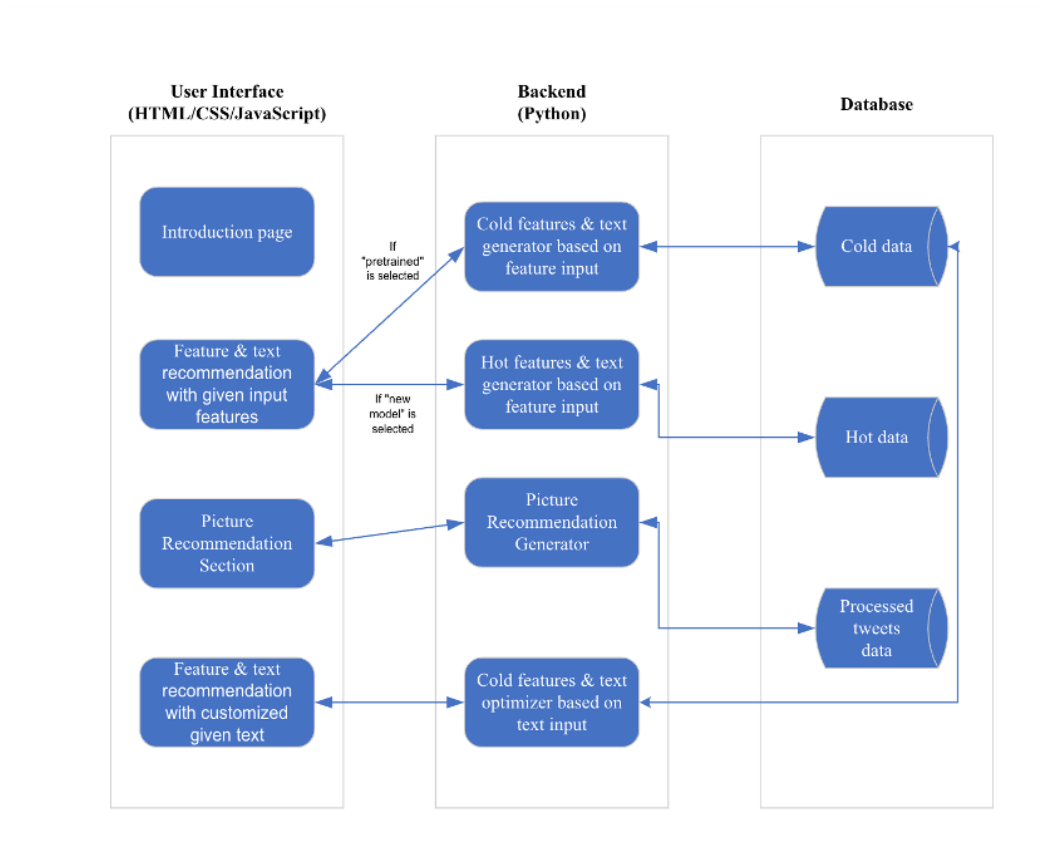


Figure 4: System Design

3.2 System Feature

On this basis, three modules are developed, namely Tweets Feature Recommender, Image Content Recommender and Tweet Optimizer. As the first feature Tweets Feature Recommender shown in the system diagram, in the front-end, we can choose to preprocess the model or train the model based on the user inputs, and the corresponding text generation module in the back-end also differentiates based on different data inputs. The first feature, Tweets Feature Recommender, can be selected in the front-end to preprocess the model or train the model based on the selected date range according to the user's input, and the corresponding text generation module in the back-end is also differentiated based on different data inputs. The second feature is the Image Content Recommender, where the user inputs the image elements he or she wants, and calculates the similarity with the processed data, returning the image features and a detailed description of the image content. The third function is Tweet Optimizer, where the user inputs the item and article description, and outputs the recommended text.

The most important features of this project's recommender system are the following:

1. High Quality The system learns all the data from quality tweets that have been filtered by machine learning models and return quality texts with the highest likelihood.
2. Real-time: Users can select the latest or determine the time range of the tweet library for learning by selecting the time node, and it takes only a few seconds to push the results after meeting the submitted features or text.
3. Personalized input In addition to allowing the user to select any inputs they want in the feature columns, the system can also quickly expand the range of products into user-customizable areas.

3.3 frontend design

The project's web front-end design uses three main web technologies, which will be described in detail below. Considering that the front-end needs to interface with the back-end python data processing and algorithmic programs, the flask framework is used as the medium between the

html front-end code and the python script. Figure 5 is a diagram of the relationship between front-end technologies:

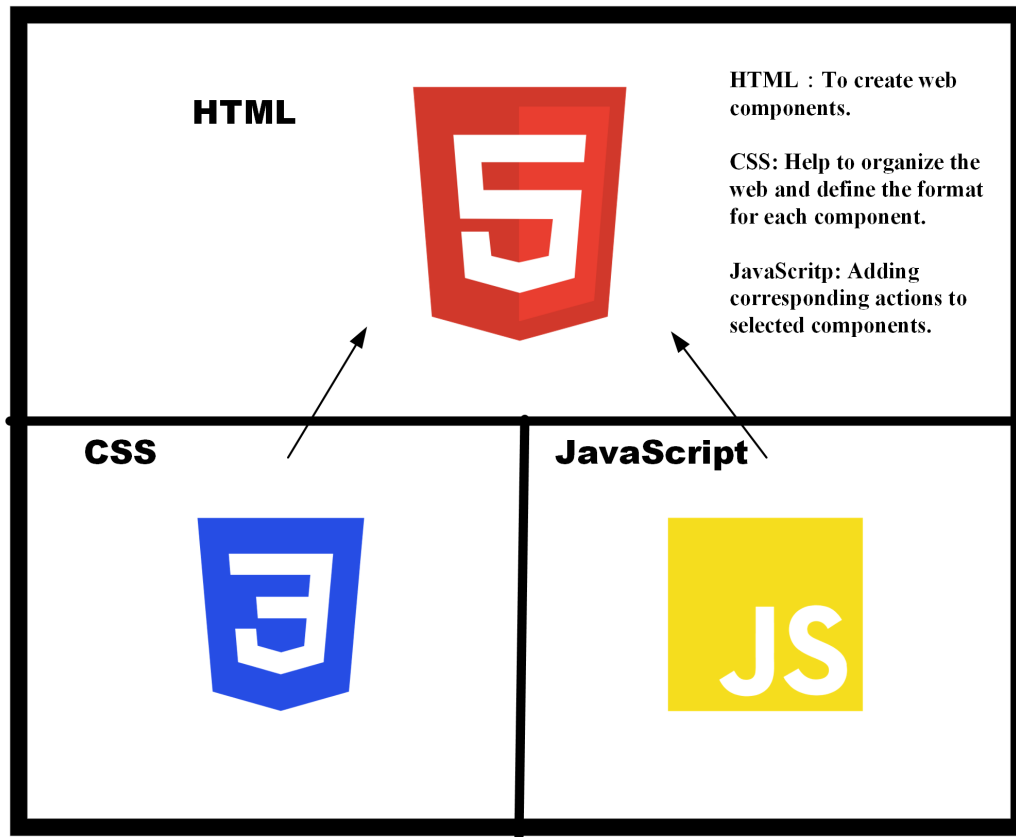


Figure 5: Frontend SD

The three main web technologies are HTML, CSS, and JavaScript, of which HTML builds all the elements that can be seen on a web page, including paragraphs, headings, images, tables, and so on, and HTML tags (such as `<div>`, `<p>`, `<h1>`, and ``) are used to define the structure of a web page. However, there is a problem with using HTML alone, and that is the formatting and styling of the text on the page. HTML can be constructed efficiently, but it is not well organized, so CSS is used to enhance this part of the design.

CSS is a style sheet language that defines the look and feel of a web page. It controls the layout, colors, fonts, and other appearance attributes of web page elements, and can be developed using CSS rules (e.g., `color: red;`, `margin: 10px;` etc.) to define the style of web page elements. After the introduction of CSS files as style files, you get a neatly arranged web page. At this

point, the construction of a static web page is basically complete, but the web page not only needs to be aesthetically pleasing for the user to navigate, but it is also very important to interact with the user. For example, if you build a button in a web page, the button must be bound to a trigger, and you have to write the event that occurs when the button is pressed, which is not possible with HTML, so you use JavaScript to write this part of the page.

JavaScript is a dynamic programming language used to implement interactive and dynamic effects on web pages. It can manipulate elements on a web page, handle user input, send network requests, and more. Therefore, by combining these three technologies, we can design a web interface that meets our expectations. HTML defines the structure of a web page and CSS defines the style of a web page. CSS is used to beautify HTML elements and make them more attractive. Elements in HTML can be accessed and manipulated by JavaScript. JavaScript is used to add interactivity, such as button click events, form validation, and so on.

After building a reasonable front-end, considering the part of data processing can not use html or JavaScript to deal with, because the data processing and algorithms include a lot of must be used in Machine Learning library, these JavaScript is not So to build our back-end must use Python, but The html and Python bases are different and can't be imported directly, so we have to use flask to connect the html front-end to the Python back-end. In the end, the design of our entire project is complete.

Our page is divided into four blocks, corresponding to the introduction page and three functions Tweets Feature Recommender, Image Content Recommender and Tweet Optimizer.

The Introduction page displays the features of the program as well as an introduction. As shown in Figure 6 and Figure 7:



Figure 6: introduction page1

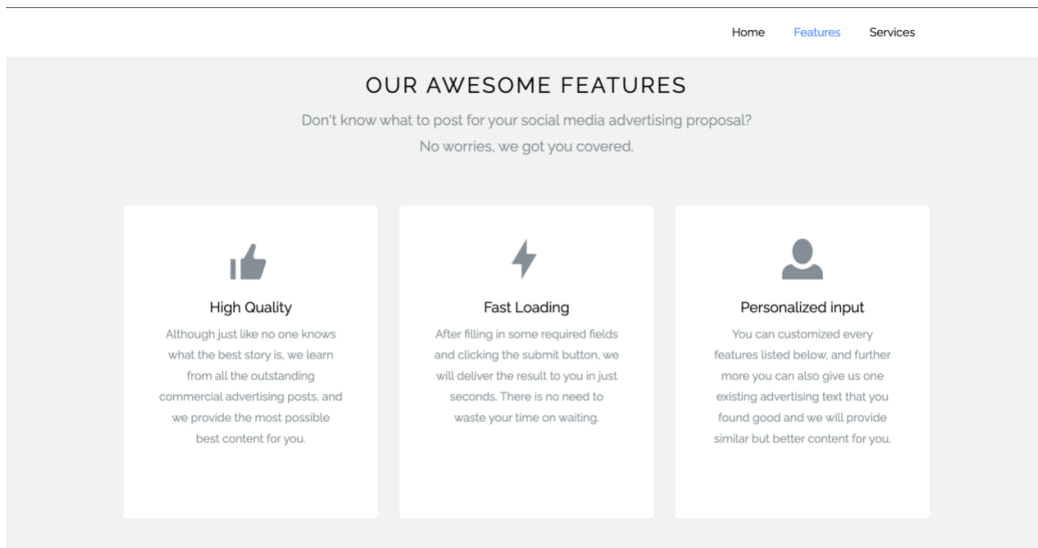


Figure 7: introduction page2

Tweets Feature Recommender for the first function, the page is after crawling twitter posting data extracted from the article features, a total of 8, and the space below can write on their own articles in the keywords you want to generate the article features recommended and recommended text output as shown in Figure 8 and Figure 9:

Categories

Lipstick

Features

Content max length

normal

Sentiment number

positive

Mention count

No special need

Number of hashtags

few hashtags

Word count

medium

Image count

No special need

Paragraph count

No special need

Interaction index

No special need

Top 3 keywords

lipstick, hot, summer

SUBMIT

Figure 8: Tweets Feature Recommender1

Recommended features:

Content length: normal

Sentiment number: negative

Mention count: not mentioned

Number of hashtags: no hashtag

Word count: medium

Image count: one image

Paragraph count: 1

Interaction_index: medium

Recommended content:

Enhance your beauty with the perfect shade of lipstick!

Whether you're looking for something bold and daring, or

subtle and classy, find the ideal hue for you that will make

you look and feel your best. #LipstickLove

#MakeupEssentials #BeautyGoals

Figure 9: Tweets Feature Recommender2

Image Content Recommender is the second function that outputs image features and text by inputting the desired features in the image, where the text describes the content of the image in detail. This is shown in Figure 10:

...Or you can get the picture recommendation below

Picture keywords

SUBMIT

Recommended image labels:

Forehead; Joint; Skin; Lip; Chin; Shoulder; Eyelash; Neck; Eye liner; Makeover

I can imagine an image of a woman with long brown hair looking into the camera with her forehead, lips, chin, neck, and shoulders in full view. She is wearing subtle eye liner and has long eyelashes. Her skin is lightly made up, giving her a natural glow. Overall, she looks confident and beautiful.

Figure 10: Image Content Recommender

Tweet Optimizer, which inputs the type of item the user wants, as well as the text of the advertisement or a text description, and outputs the text and recommended features, as shown in Figure 11:

Category

The advertising copy

...that you found great

use earphone for better experience

SUBMIT INPUT

Recommended features:

Item category: earphone

string length of content: 92

Sentiment of content (-1 for negative and 1 for positive): 0.5

Mention number of content: 0

Number of Hashtags: 0

Recommended content:

Check out our #Naturalrate earphones that are perfect for a wedding. They are comfortable, good quality, and okay for any occasion. Get yours now! #Wedding #Okay #Earphones

Figure 11: Tweet Optimizer

3.4 Backend design

3.4.1 Tweets Feature Recommender

The data used is cold data processed according to the data features. Firstly, the continuous values are mapped into discontinuous values and the mapped data is processed by onehot coding.

In the next step, the frequent itemset is mined from onehot data using apriori algorithm and the minimum support is set to 0.2. Association rules are generated from the frequent itemset using `association_rules` method and filtered according to the user inputs to get the items related to the user inputs.

Extract the longest set of related association rules and connect them with the user input to get the longest feature combination. Convert the text data into a vector of values and calculate the cosine similarity between the converted data and the data items in the quality dataset, and finally return all the recommended feature values. Finally, based on the article features and also the input keywords, passed into the LLM to generate the recommended text. The main flow is shown in Figure 12:

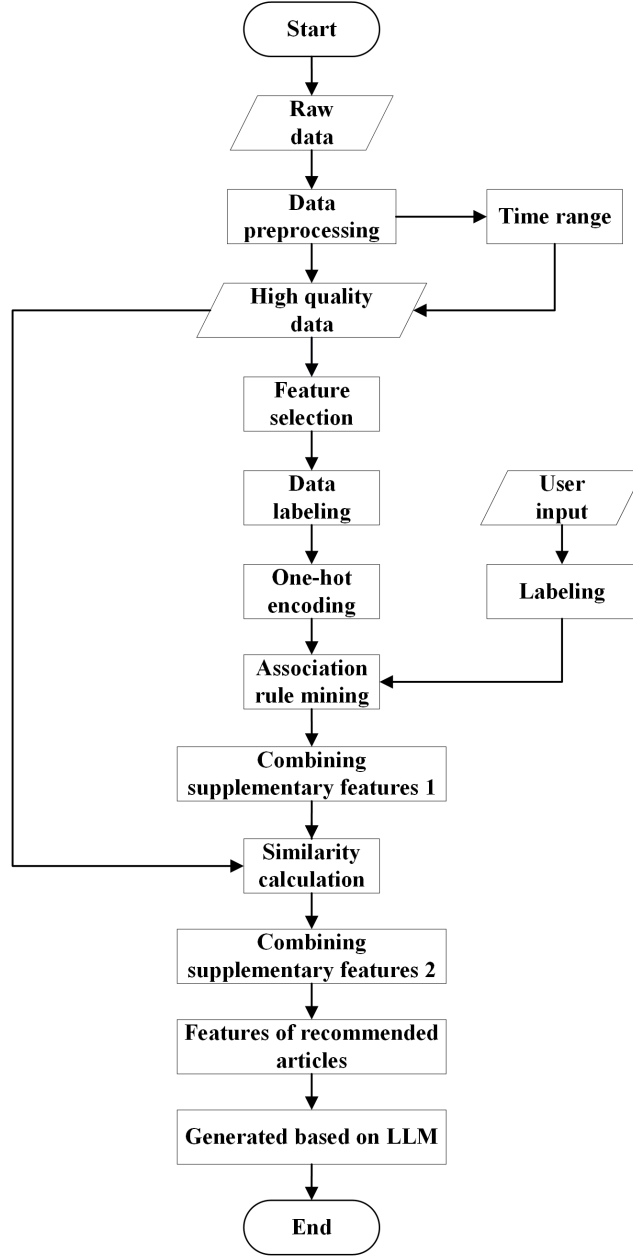


Figure 12: Tweets Features Recommender's flow

3.4.2 Image Content Recommender

In the front-end, the user can input the keywords of the elements they want to include in the image, and the back-end converts the text of the keywords into bag-of-words feature vectors according to the user's input, and then uses the pre-trained model to predict the image features, and finally converts the predicted image features back into a list of strings and returns them.

The pre-training model in this module first extracts the Tweet_Content and Imagelabels in the dataset as the text and image features of the training samples, converts the text into bag-of-words feature vectors, trains them using SVM model, and stores them for use in prediction.

Then the returned list of strings is output as image features and simultaneously fed into LLM for text generation. Eventually, the user input Keywords are used to return both the image feature recommendations and the actual image content description text, the main flow is shown in Fig. 13:

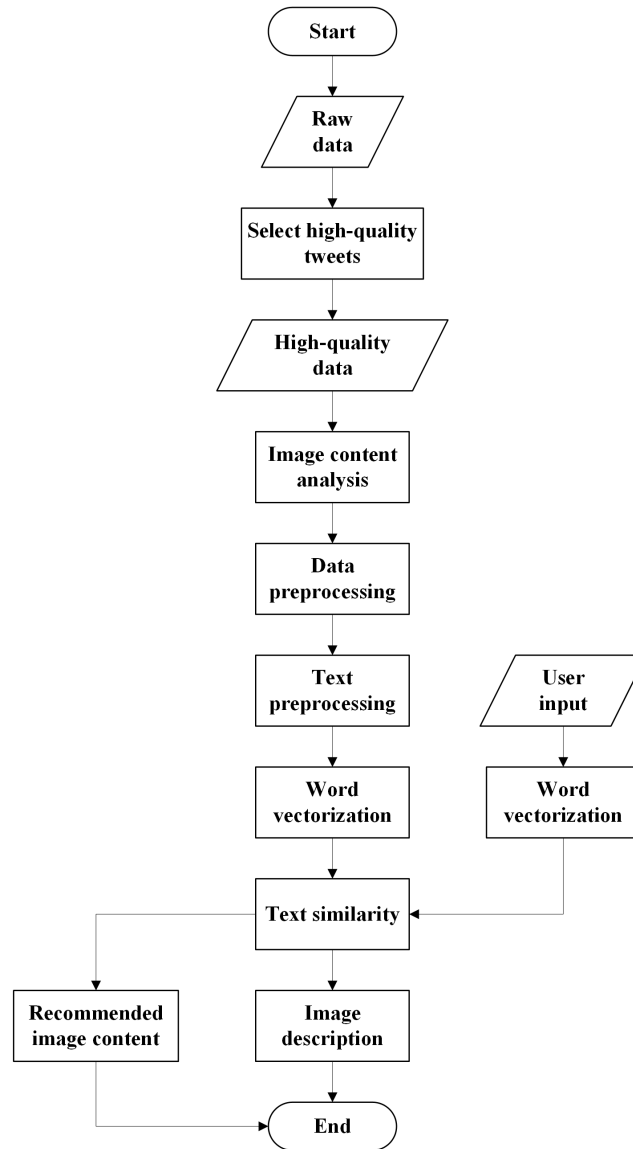


Figure 13: Image Content Recommender's flow

3.4.3 Tweet Optimizer

The text data is obtained from the user's input articles, the text data is trained by word2vec and spliced with other user's desired features, then the spliced data is checked for similarity with the dataset and the most similar data is found out in the dataset to be combined with the text, and finally it is passed into LLM to generate the text and output. The main flow is shown in Figure 14:

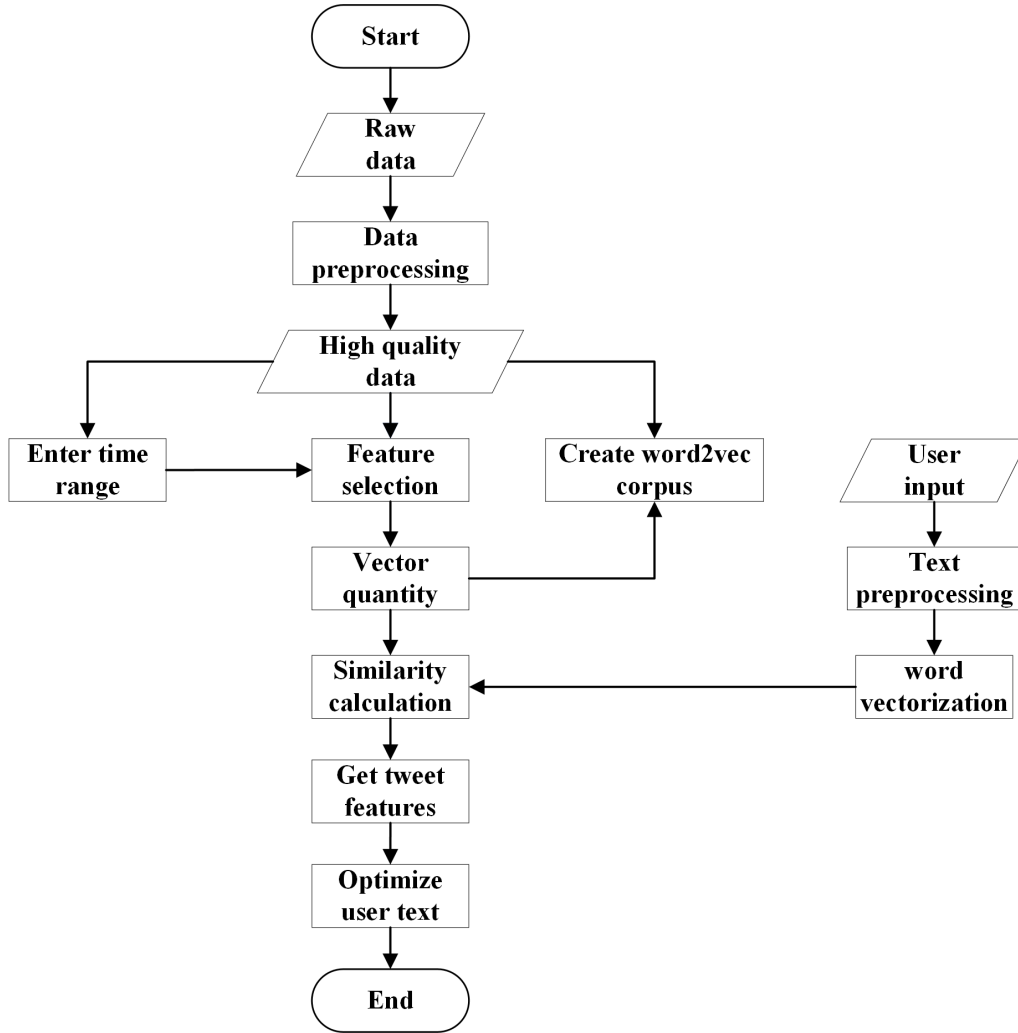


Figure 14: TweetOptimizer's flow

4 Model

4.1 Natural Language Process

The processing of text plays a key role in our project. We have gone through text data preprocessing and in the initial stage of recommender system, we have used two different approaches, Word2Vec as well as TF-IDF, to mine text data and word vectorization.

4.1.1 Word2Vec

Word2vec modeling maps each word in a text to a high dimensional vector space. This technique can help us capture the semantic relationships between words to better understand the importance and relationships in text data. With Word2Vec, we are able to achieve more accurate text similarity computation and contextual understanding to provide more precise information to the recommender system. It can be obtained using two methods: the Continuous Bag of Words (CBOW) and Skip Gram.

The CBOW model takes the context of each word as input and tries to predict the word that corresponds to the context, the following figure 15 shows the architecture of this approach.

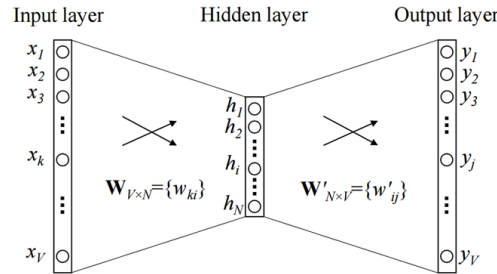


Figure 15: CBOW model architecture diagram for single word

The input consists of a one-hot encoded vector of size V , representing the context word. The hidden layer comprises N neurons, and the output is a V -dimensional vector with elements as softmax values. The parameters in the diagram, $\mathbf{W}_{v \times n}$ denote the weight matrix ($V * N$ dimensions) mapping the input \mathbf{x} to the hidden layer, while $\mathbf{W}'_{n \times v}$ represents the weight matrix ($N * V$ dimensions) mapping the hidden layer output to the final output layer. The neurons in the hidden layer

simply copy the weighted sum of inputs to the next layer without using any activation functions like sigmoid, tanh, or ReLU. The only non-linearity is introduced in the softmax computation at the output layer. This model predicts the target using a single context word. The structure employing multiple context words is illustrated in the diagram below.

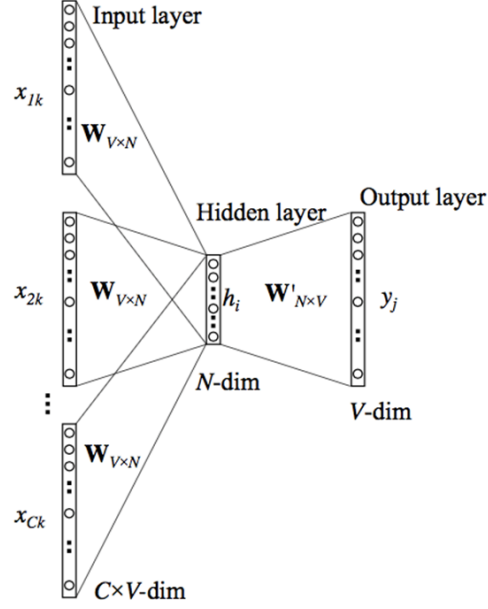


Figure 16: CBOW model architecture diagram for multiple word

The CBOW model is the opposite of Skip-Gram in that its goal is to predict the center word [5] given the surrounding context words. The advantage of the CBOW model is that it performs better on small-scale datasets because it can learn the representation of the center word from the co-occurrence information of the context words. It works better with common words and small size corpus.

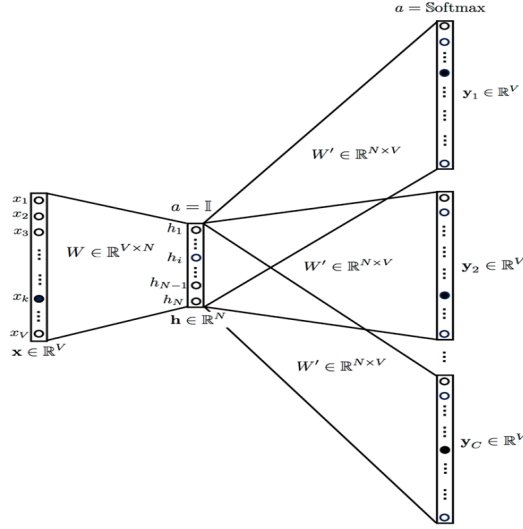


Figure 17: Enter Caption

In the Skip-Gram model, given a central word (the target word), the goal of the model is to learn to predict the probability distribution of the surrounding context words given the central word. For each center word, the goal of the model is to maximize the conditional probability of the surrounding context words given the center word. Through this process, the model learns a vector representation of each word, allowing these vectors to capture the semantic relationships between words.

The advantage of the Skip-Gram model is that it can handle large-scale corpora because it requires relatively few training samples. It works better when dealing with rare words and large-scale corpora.

Overall, the Skip-Gram model is better suited to handle large-scale corpora and rare words, while the CBOW model is better suited to handle small-scale datasets and common words. Which model to choose depends on your application scenario and data characteristics. For the characteristics of the dataset in the project, analyzing the content of the posting articles and is a small dataset, so this project uses CBOW's model for training.

In the Tweet Optimizer section, we vectorized the pool of tweets text in the dataset through the word2vec model of the CBOW method, and combined it with other feature vectors to splice it into a complete vector, which serves as a processing before the similarity-based recommendation model.

4.1.2 TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) is a statistical method for measuring the importance of a word in a collection of text. It identifies the most critical words in a text by analyzing the frequency of the word in the text. In our model, we use this method to extract the keyword of tweets.

The calculation of TF-IDF includes two steps

Step1 Term Frequency (TF): Measure the importance of a word in the single text, the formula shows in below:

$$TF(t, d) = \frac{\text{Number of occurrences of word } t \text{ in document } d}{\text{Total number of words in document}} \quad (1)$$

Step2 Inverse Document Frequency (IDF): Measure the importance of a word in the whole text pool, the formula shows in below:

$$IDF(t, D) = \log \frac{\text{Total number of documents in text set } D}{1 + \text{Number of documents containing the word } t} \quad (2)$$

Step3 TF-IDF : TF-IDF weights are obtained from the product of word frequency and inverse document frequency.

$$TF - IDF(t, d, D) = TF(t, d) * IDF(t, D) \quad (3)$$

4.2 Apriori based association rule mining

This section is used in Tweets Feature Recommender. The user's input is part of the text feature which is recommended by this model to get the recommended text feature. This section describes the algorithm and processing of this part.

4.2.1 Data discretization

In Tweets Feature Recommender, we use Apriori based association rule mining to build the recommender system, firstly, we perform discretization transformation on the selected representative multiple features. For example, one of them, text sentiment, has a value from -1 to 1,

which is not applicable in both user input and labeling judgments, so it is discretized to three labels: negative, neutral, and positive. For example of this type, this is done for all continuous values in this sub-dataset. In the below table are the discrete category:

Table 3: Discrete Transformation of Features

| Feature | Discrete Value |
|-----------------|--------------------------------------------|
| Content length | very short, short, normal, long ,very long |
| Sentiment | negative, neutral, positive |
| Mention count | not mentioned, mentioned |
| Hashtags count | no hashtags, few hashtags, some hashtags |
| Word count | less, medium, many |
| Image count | no image, one image, some images |
| Paragraph count | one, two, three |

Next, onehot is performed on all discrete valued features and then the association rule is performed using the Apriori algorithm.

4.2.2 Text features recommendation based on association rule mining

Association rule mining based on apriori is a data mining technique used to discover relationships among items in a dataset. It involves identifying sets of items that frequently occur together (frequent itemsets) and then generating association rules from these itemsets.

After one-hot coding of the discrete transformed features, the features of each tweet in the dataset become discrete values of 0 and 1, so that each high-quality tweet contains a different set of "item" with different sets of features. When tweets containing only a small set of features are input, the model can recommend more representative combinations in recent months based on historical data and splice them with the original features.

However, the set of features after such a recommendation does not necessarily contain all the features, and we use strategy combination to deal with this problem.

1. Try enough combinations of confidence and support to find pairs of values that can rec-

ommend more features when it makes sense.

2. After combining the original and recommended features, compare the similarity with the feature set of high-quality tweets within the original dataset, and concat the features from the highest similarity tweets which are still not in the current combination.

Assuming the user input is ["content_str_len_very long", "sentiment_positive", "Num_Hashtags_few hashtags"], the following results can be seen in the table below.

| Antecedents | Consequents | Ant Sup | Cons Sup |
|-----------------------------------|----------------------------|---------|----------|
| sentiment_positive | not mentioned | 0.3046 | 0.8958 |
| sentiment_positive | paragraph_1 | 0.3046 | 1 |
| not mentioned, sentiment_positive | paragraph_1 | 0.2529 | 1 |
| paragraph_1, sentiment_positive | mentioned | 0.3046 | 0.8958 |
| sentiment_positive | not mentioned, paragraph_1 | 0.3046 | 0.8958 |
| no hashtag, sentiment_positive | paragraph_1 | 0.2055 | 1 |

After getting the returned data, based on the obtained features, cosine similarity is calculated between the data and the mapped dataset to find the most similar post and complement the features not selected by the user to return all the feature recommendations.

5 Dataset Processing

5.1 Dataset

5.1.1 Data Collection

We use the official API of X (Twitter) to obtain different information of specific category. The API that X uses to grab information about tweets also has libraries in Python by the name of Tweepy available. The workflow is shown in Fig. 18.

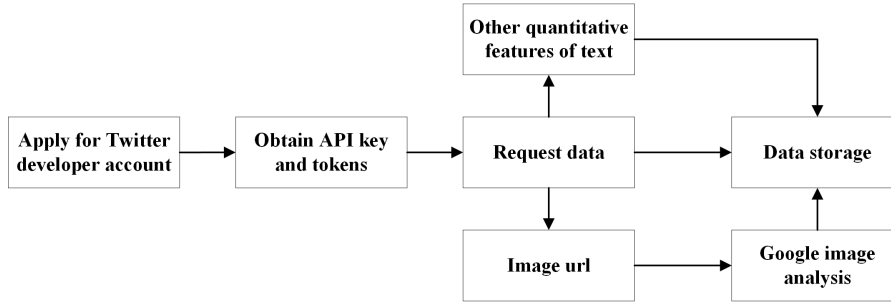


Figure 18: X Crawler Workflow

5.1.2 Data Description

We get the raw data by using the API of X. We collect post in makeup from 21-10-2022 to 20-10-2023, totally 7230 posts. Then we compute the text content of the tweet and the image url to get the specific features of the text, including . In addition, for the analysis of the image content, we use the Google Vision API to extract a lot of tags of the image and store them as the information of the tweet image, shown as ??.

5.2 Data Processing

In order to provide a usable base dataset for the model, this raw dataset requires a series of processing and new feature creation.

5.2.1 Data Cleaning

We remove samples without timestamps. Typically each tweet has a timestamp. The lack of timestamps leads to errors in determining the time period for high views and other interaction metrics, and is also unusable in real-time model runs.

In addition, the other irrelevant features like `tweet_website`, `tweet_video_URL` are dropped.

5.2.2 Text Preprocessing

Before the traditional text preprocessing, we learn more features of the text.

1. Through the content of the tweets, number of images, number of paragraphs, number of hashtag mentions in the tweets were counted to turn them into features as well.
2. Utilize $TF - IDF$ to extract the top 3 keywords in each tweet from the already cleaned tweet content.
3. Utilize the Cloud Natural Language API of Google cloud Analyzing Sentiment to analyze the sentiment of the cleaned tweets and get the sentiment values in the interval between -1 and 1. The sentiment values represent the degree of sentiment of the tweets from negative to positive.

We perform data cleansing on the text. Since there is a lot of redundant information in text-based data, we first preprocess the text-based data before analyzing the data Fig. 19)

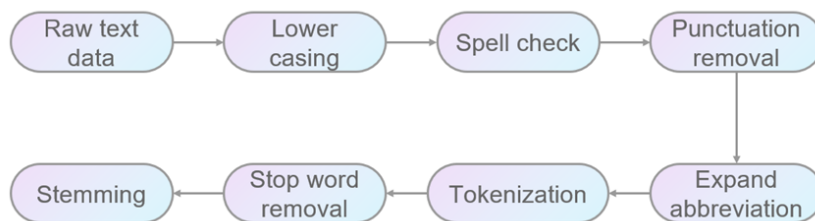


Figure 19: Text Preprocessing

5.2.3 Post Clustering

For users with different follower counts, it is not reasonable to directly use the interaction metrics of the postings because users with higher follower counts, especially the head opinion leaders, are more likely to get higher views and interactions regardless of the quality of the postings.

In order to be able to better understand and analyze the users, we use the number of followers as well as the number of views of the users as the corresponding features for K-means clustering shown in Fig. 20. We can observe that the views of tweets in groups with small follower size show different distributions. There are a large number of tweets with a high number of views, which suggests that tweets with a small number of followers may also be "explosive".

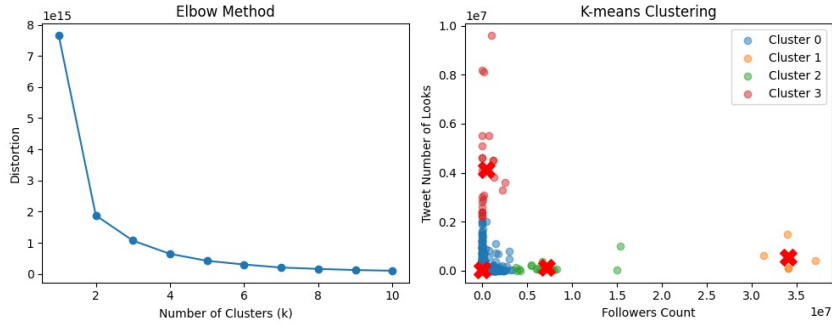


Figure 20: X user K-means elbow map and cluster map

5.2.4 Interactive Indicators

In order to be able to combine all known factors to determine the quality of a tweet (both exposure as well as text quality), we need to create an interaction metric. To do this, we need to combine all four metrics related to exposure and text quality: number of comments, retweets, likes and views.

First, we start by analyzing the correlation between metrics and eliminating metrics with too much correlation to reduce the amount of calculations. Based on the number of comments, retweets, likes and views the correlation between them was obtained (Fig. 21). As can be seen from the figure, the correlation between the number of likes and views reaches 0.87, which is close to 0.9. This indicates that the correlation between the two is very high, so the number of

views is removed.

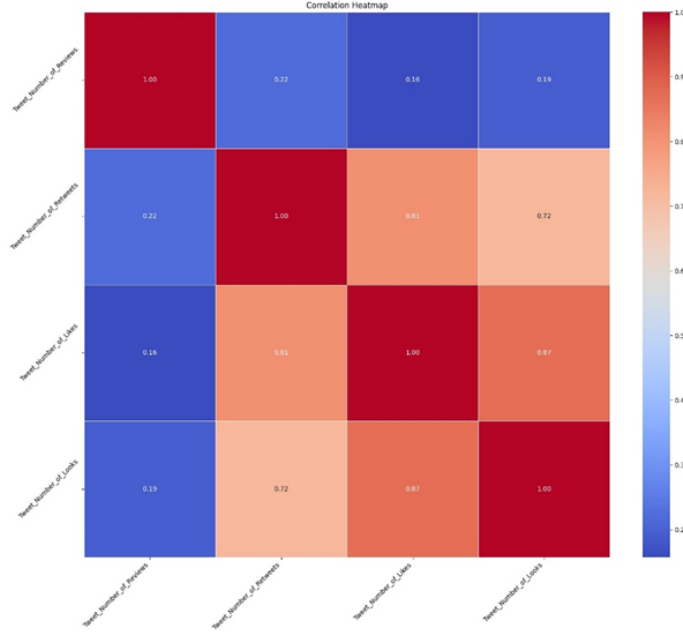


Figure 21: The correlation between the number of comments, retweets, likes and views

Next, we took the logarithm of the sum of these three as the interaction metrics and obtained the logarithmic distribution of user interaction metrics (Fig. ??). As can be seen from the figure, most of the tweets have no or only few interaction indexes, which confirms the current confusion of brands and companies about the strategy of choosing tweets related to the promotion of their products, and also reflects the importance of our project. When the interaction index we will take 100 (horizontal coordinate 2), the curve ushers in an inflection point, so we will use the interaction index of 100 as a threshold to classify superior and inferior tweets.

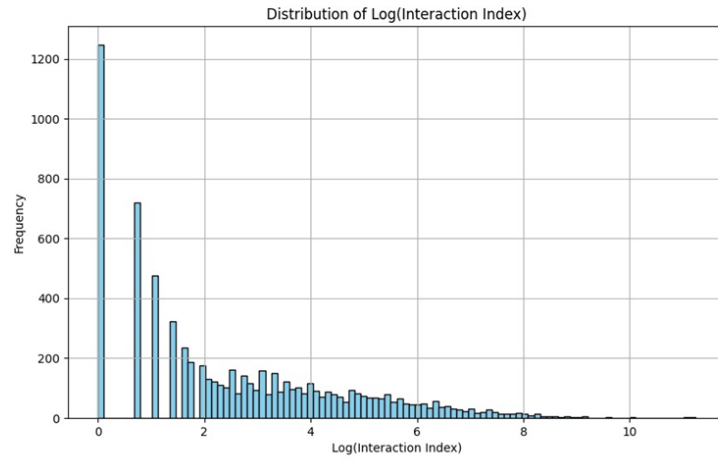


Figure 22: Distribution of Log(Interaction index)

Finally, the features of the dataset is shown in below.

Table 4: Feature Labels of Raw Data

| Feature | Description |
|--------------------------|----------------------------------------------------------|
| Category | The category or type of the data |
| Keyword | Keywords associated with the tweet |
| Tweet_Website | The website linked in the tweet |
| Author_Name | Name of the author of the tweet |
| Author_Web_Page_URL | URL of the author's webpage |
| Tweet_Timestamp | Timestamp when the tweet was posted |
| Tweet_Content | Text content of the tweet |
| Tweet_Image_URL | URL of images included in the tweet |
| Tweet_Video_URL | URL of videos included in the tweet |
| Tweet_AD | Indicates whether the tweet contains advertising content |
| Tweet_Number_of_Reviews | Number of reviews or comments on the tweet |
| Tweet_Number_of_Retweets | Number of times the tweet was retweeted |
| Tweet_Number_of_Likes | Number of likes received by the tweet |
| Tweet_Number_of_Looks | Number of views the tweet received |
| content_str_len | Length of the text content of the tweet |
| sentiment_number | Sentiment score of the tweet |
| Imagelabels | Labels associated with images in the tweet |
| Twitter_Username | Twitter username of the tweet author |

Table 5: Feature Labels of Raw Data

| Feature | Description |
|--------------------------|--------------------------------------------------|
| Keyword | Keywords associated with the tweet |
| Tweet_Timestamp | Timestamp when the tweet was posted |
| Tweet_Number_of_Reviews | Number of reviews or comments on the tweet |
| Tweet_Number_of_Retweets | Number of times the tweet was retweeted |
| Tweet_Number_of_Likes | Number of likes received by the tweet |
| Tweet_Number_of_Looks | Number of views the tweet received |
| content_str_len | Length of the text content of the tweet |
| sentiment_number | Sentiment score of the tweet |
| Imagelabels | Labels associated with images in the tweet |
| Followers_Count | Number of followers of the tweet author |
| Friends_Count | Number of friends/followings of the tweet author |
| Mention_Count | Number of mentions in the tweet |
| Num_Hashtags | Number of hashtags in the tweet |
| Hashtags_Content | Content of hashtags used in the tweet |
| Cleaned_Tweet_Content | Cleaned text content of the tweet |
| Top_3_Keywords | Top three keywords associated with the tweet |
| Word_Count | Number of words in the tweet |
| Image_Count | Number of images in the tweet |
| Paragraph_Count | Number of paragraphs in the tweet |
| Cluster_Label | Cluster label assigned to the tweet |

6 Demonstration and Test

6.1 Usage Demonstration

6.1.1 Tweet Optimizer

In Tweet Optimizer, users can get recommended articles by typing in the article or part of the passage they want to emulate or learn from, after determining the topic related to the article they want to output.

Each time the input content is different, the output result is also different, and the generated articles will also be different. The following is a preset output for reference. The result shows in Fig.23

..Or you can customize your input below

Category

green

The advertising copy

..that you found great

for skin that still feels and looks like skin [❤️ 40 shades](#) of power plush are now available on our site and at @ultabeauty

SUBMIT INPUT

Recommended features:

- Item category: green
- string length of content: 49
- Sentiment of content (-1 for negative and 1 for positive): -0.9
- Mention number of content: 0
- Number of Hashtags: 0
- Top 3 Keywords of post: sickening, still, time
- Number of words: 9
- Number of image: 1
- Number of paragraph: 1

Recommended content:

Sickening to think that green is still an issue in this day and time. #ThinkGreen
#GoGreen #Sustainability

Figure 23: Result of Tweet Optimizer

6.1.2 Tweets Feature Recommender

In Tweets Feature Recommender, by using the features that the user wants to publish an article with, such as text length, post sentiment, and other features, the model will recommend the highest quality features to the user and generate an article based on those features, shown in Fig.24

The interface for the Tweets Features Recommender includes the following sections:

- Categories:** A dropdown menu with "Lipstick" selected.
- Features:** A grid of dropdown menus for:
 - Content max length: short
 - Sentiment number: positive
 - Mention count: mentioned
 - Number of hashtags: few hashtags
 - Word count: medium
 - Image count: one image
 - Paragraph count: 1
 - Interaction index: medium
- Top 3 keywords:** A text box containing "beauty, red, color".
- SUBMIT:** A blue button.
- Recommended features:**
 - Content length: normal
 - Sentiment number: negative
 - Mention count: not mentioned
 - Number of hashtags: no hashtag
 - Word count: medium
 - Image count: one image
 - Paragraph count: 1
 - Interaction index: medium
- Recommended content:**

"When it comes to options, there's no one-size-fits-all solution. Finding the right one requires careful consideration of the pros and cons. @mention #options #decision #choices #experts" #illustration (include one relevant image!)

Figure 24: Result of Tweets Features Recommender

6.1.3 Image Content Recommender

In Image Content Recommender, the user can input the key content of the desired image, and the recommender will output the most suitable image according to the algorithm and suggest the specific content description of the image, shown in Fig.25.

The interface for the Image Content Recommender includes the following sections:

- Picture keywords:** A text box containing "red".
- SUBMIT:** A blue button.
- Result:** A box containing the text: "A bright red rose with a green stem and leaves standing upright in a white vase against a black background."

Figure 25: Result of Image Content Recommender

6.2 Practical Test and Result

To verify that our article recommendation algorithm works, we conduct a test experiment. We used the same keywords, using our model as well as directly letting Chatgpt generate articles separately for a total of 90 pairs of articles. Posting was done on 100 Twitter accounts that we created. These accounts all have 0 followers and between 0 and 6 followers, post nothing before.

The metric for evaluating the quality of the generated articles is the number of views each article receives upon publication. Higher view counts signify better exposure on the platform, which can be inferred that it can bring higher readership in a normal running account.

6.2.1 Test Method

In this task, we utilized an independent samples T-test to compare the means of two groups. The independent samples t-test is employed to determine whether there is a significant difference between the means of two independent groups.

For our study, we aimed to investigate whether the mean of the first set of data is significantly greater than the mean of the second set of data. Our null hypothesis (H_0) states that the means of the two groups are equal ($\mu_1 = \mu_2$), while the alternative hypothesis (H_1) states that the mean of the first group is greater than the mean of the second group ($\mu_1 > \mu_2$).

First, we computed the means and standard deviations for both sets of data:

Next, we calculated the t-statistic using the formula:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\left(\frac{s_1^2}{n_1}\right) + \left(\frac{s_2^2}{n_2}\right)}} \quad (4)$$

The calculated t-statistic is approximately 3.17.

We determined the degrees of freedom (df) using the formula:

$$df = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\left(\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1-1}\right) + \left(\frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2-1}\right)} \quad (5)$$

6.2.2 Result

By referencing the t-distribution table for $df = 178$ at a significance level of $\alpha = 0.05$, Since the P-value is smaller than α and t is greater than 0, we reject the hypothesis zero. This indicates that at a significance level of $\alpha = 0.05$, there is evidence to conclude that the mean of post generated by our model is statistically greater than the mean of that of ChatGPT, the

histogram is shown in Fig.26.

$$P - value = 0.0346 < \alpha = 0.05 \quad (6)$$

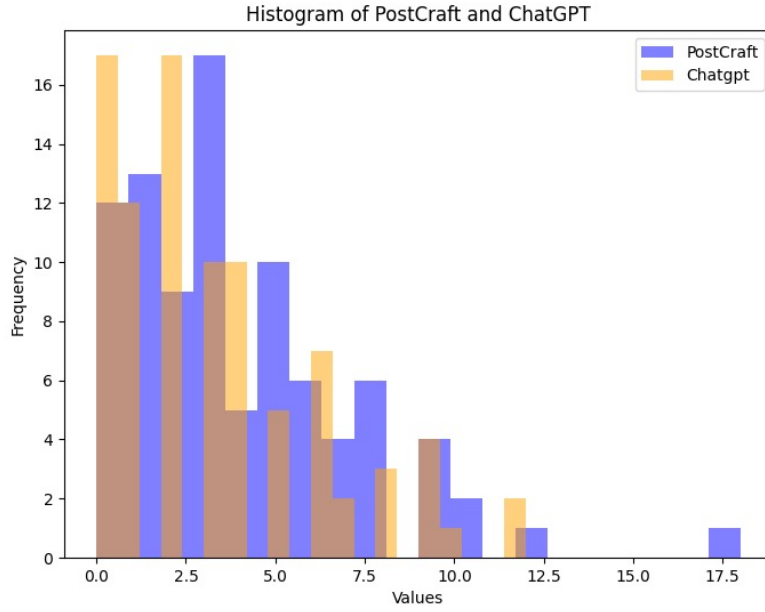


Figure 26: Histogram of Two Method

Although intuitively there is no sharp difference, statistically our model does have better results, and only on a group of accounts that have not been created for a long time and are almost new. At the same time, the effect in the normal operation of the account in need of promotion still need to be observed and calculated.

7 Conclusion and Future Work

The demand for product promotion on media platforms is increasing, especially among traditional businesses unfamiliar with internet marketing. Our model caters to their needs by providing real-time tweet recommendations tailored to their specific fields.

Additionally, our model is not limited to the Twitter platform alone. With slight adjustments to the dataset, it can easily be adapted into a versatile, multi-platform solution, accommodating various media platforms effortlessly.

In the future, we first can improve the user interface for seamless user experience and optimize user interactions. Second, we can expand the dataset to encompass a wider range of product categories and media platforms, enabling a more comprehensive coverage in our recommendations. Finally, Collaborating with more KOLs to validate the model’s effectiveness and making necessary adjustments based on their feedback, ensuring the recommendations align with market trends and user preferences.

8 Appendix A: Project Proposal

Project Title

POSTCRAFT: INSTAGRAM CONTENT RECOMMENDATION AND STRUCTURING SYSTEM

Group Information

Group ID: 9 **Group Members:**

- Liu Siyan (A0285857H)
- Lai Weichih (A0285875H)
- Lin Zijun (A0285897Y)
- Fang Ruolin (A0285983H)

Potential Sponsor/Client

1. Companies, Brands, or Individuals with Product Promotion Needs
2. Multi-Channel Network (MCN) Companies

Background/Aims/Objectives

Background and Context: The digital marketing landscape has undergone a significant transformation with the advent of social media platforms. Platforms like Instagram have become pivotal venues for businesses to promote their products and engage with potential customers. However, the crowded social media space poses a challenge for brands to stand out and effectively reach their target audiences.

Problem Statement: One of the primary challenges faced by brands is identifying the optimal set of article features that would ensure the success of their promotional campaigns.

The lack of real-time, data-driven insights to guide promotional strategies further exacerbates this challenge.

Objectives:

1. Collect and analyze social media post data related to product promotions, and identify key article features that influence promotional success.
2. Develop a recommendation model to guide brands in crafting more effective promotional posts.
3. Implement an interactive chatbot for real-time user engagement and feedback.

Project Descriptions

Recommendation systems are used in daily life in audio-visual and search platforms, as long as they recommend information that is of interest to users, they can increase the click rate and the browsing rate. The ability of recommendation systems to increase exposure and viewership is an issue that many companies are facing. This project is based on this social problem.

System Structure:

1. **Data Processing:** Crawl the desirable data from the streaming platform (Instagram in this case), such as articles, tags, images, likes, etc., as raw data for this project. Extract features from the text by NLP and other text processing methods and machine learning methods, and obtain the feature matrix.
2. **Data Analysis:** Use concordance filtering to get the relevance of the features, the importance of some words, and recommend suitable articles. Add image detection to detect the presentation method of the item in the picture, and use text to recommend how to present the item in a suitable way.
3. **Output Recommendations:** Users input the item type and platform to intelligently recommend the recommended posts and tags for that item. Combine text recommendation and image analysis results to output the recommended article content and product image presentation method.

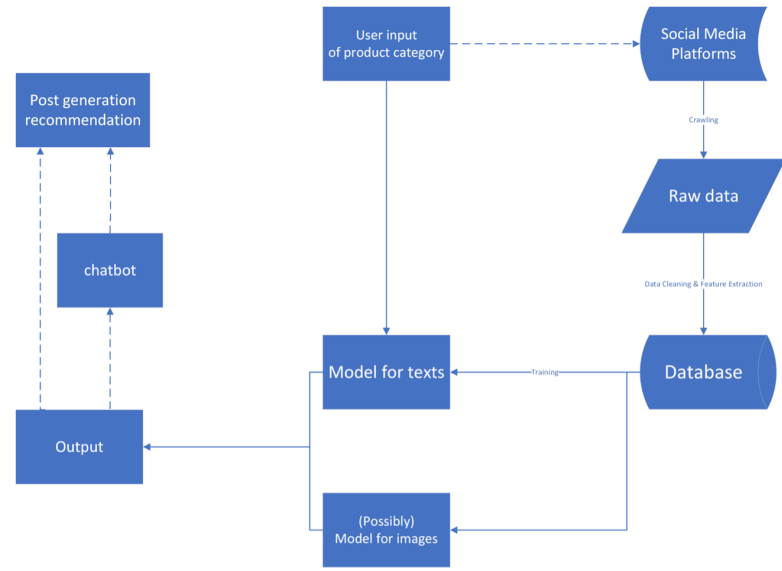


Figure 27: System Flowchart

Limitations and Difficulties

Different recommendation systems for different platforms will lead to the fact that if we want to expand to multiple platforms, we have to build independent models for the platforms that cannot be shared. Secondly, the features of different items may be very different, and the selection of features for different items is one of the most important factors affecting the performance of the recommendation system, so this project selects specific platforms and fixed items.

9 Appendix B: Mapped System Functionalities

The mapped system functionalities to the course content is shown in Table6

Table 6: Mapped System Functionalities

| Modular Courses | System Functionalities / Technique Applied |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Machine Reasoning(MR) | <ul style="list-style-type: none">· Rules: High-quality Tweet Classification· Knowledge Discovery by Machine Learning: High-quality Tweet Clustering |
| Reasoning System(RS) | <ul style="list-style-type: none">· Similarity Based Reasoning: TF-IDF, K-means Cluster· Knowledge Discovery: Some features such as Image count· Fuzzy Logic: Image count trasfer into no image, one image, some image· Association Rule: Aprior based association rule mining· Content-Based Recommender: Tweets Feature Recommender suggests items(posts) to users based on the features of items(posts)· Leveraging NLP methods:Word2Vec |
| Cognitive System(CGS) | <ul style="list-style-type: none">· Cognitive systems: Sentiment analysis of the post text· Intent classification: imge text feature Convert to the word bag feature vector and train by SVM, Tweets Feature Recommender(cosine similarity)· TF-IDF: typical pre-processing: imge text feature Convert to the word bag feature vector, associate text transform to bag feature vector· Word embedding: Tweet Optimizer(User input is trained using word2vec)· LLM: Tweets Feature Recommender(generate articles based on LLM)· Visual foundation model:Extract the characteristics of images in raw data(Call the API) |

10 Appendix C: Installation and User Guide

10.1 Introduction

Welcome to the User Guide for our Postcraft project. Postcraft is an intelligent recommendation system for our users to generate customized, real-time, and high-quality advertising content of commercial social media posts using different kinds of inputs, so that the purpose of this user guide is to help our users to utilize our Postcraft more effectively.

10.2 Browse/Install

POSTCRAFT URL: <https://postcraft-ff142951a5f2.herokuapp.com/>

To get into our page, you can simply open a browser (Chrome would be the best) and type in the URL above and enter.

If you want to use our page locally on your computer, before running our website, your environment will need to have the Python version 3.11.6 or higher, and the following essential libraries installed:

After that, please open a terminal, and here is the process of deploying our website to your local host:

- `cd <path of the system>/POSTCRAFT`
- `python run.py`

In the terminal it will tell you which host it uses (usually the `http://127.0.0.1:5000/`). Then open a browser and type in the corresponding port number and you will get into our project locally.

10.3 Overview

There's no need to install or update anything else. Overall we have 3 sections on our page (as you can see on the navigation bar on top of the website). The first section you see when you

Required Libraries

| Liabraries | Version |
|--------------|---------|
| flask | 2.2.2 |
| pandas | 1.5.3 |
| numpy | 1.24.3 |
| mlxtend | 0.23.0 |
| openai | 0.28.1 |
| scikit-learn | 1.3.0 |
| nltk | 3.8.1 |
| gensim | 3.8.1 |
| gunicorn | 21.2.0 |
| Werkzeug | 2.2.2 |
| openpyxl | 3.1.2 |

first click into our page is the home page section:

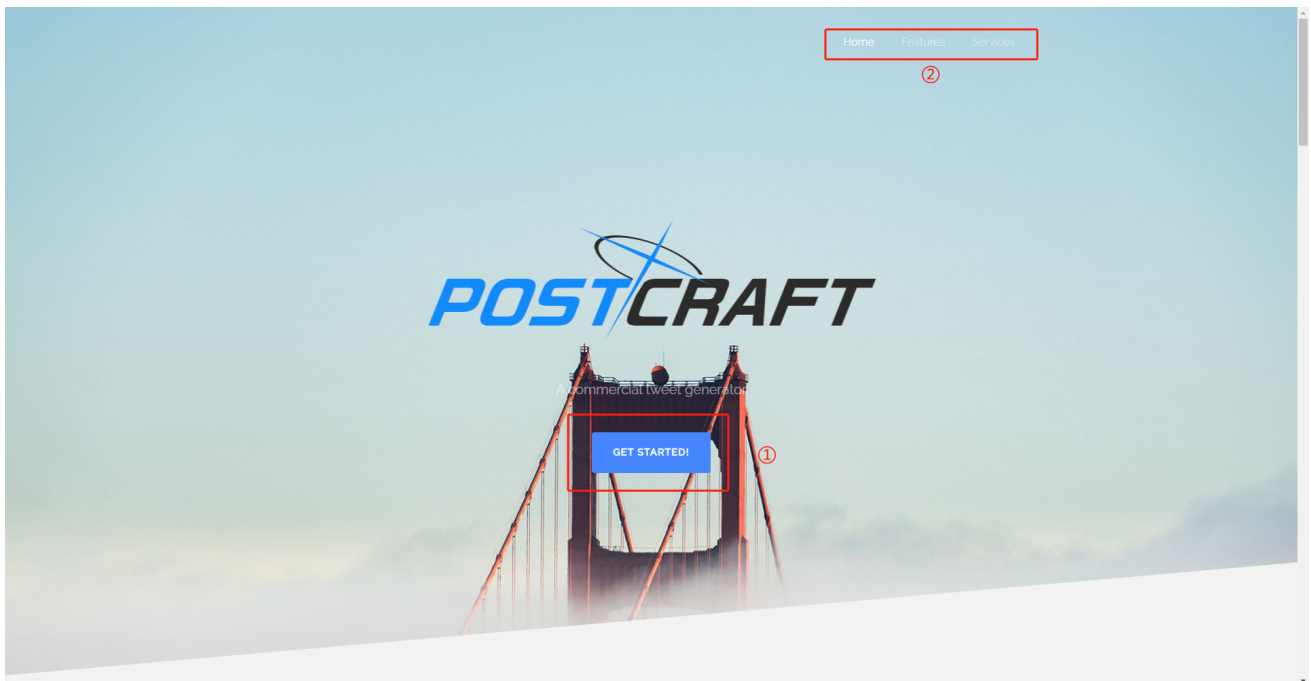


Figure 28: Home page

Here you can see a "GET STARTED!" button (1) in the middle and a navigation bar (2) on the top which contains three sections of our page. For the "GET STARTED!" button, if you click it, it will direct you to our service section, and for the navigation bar, it will direct you to each indicated section of our page (Home, Features, Services).

Next, if you scroll down or click on the "Features" in our navigation bar, you will go to the second section of our page which is the Features section:

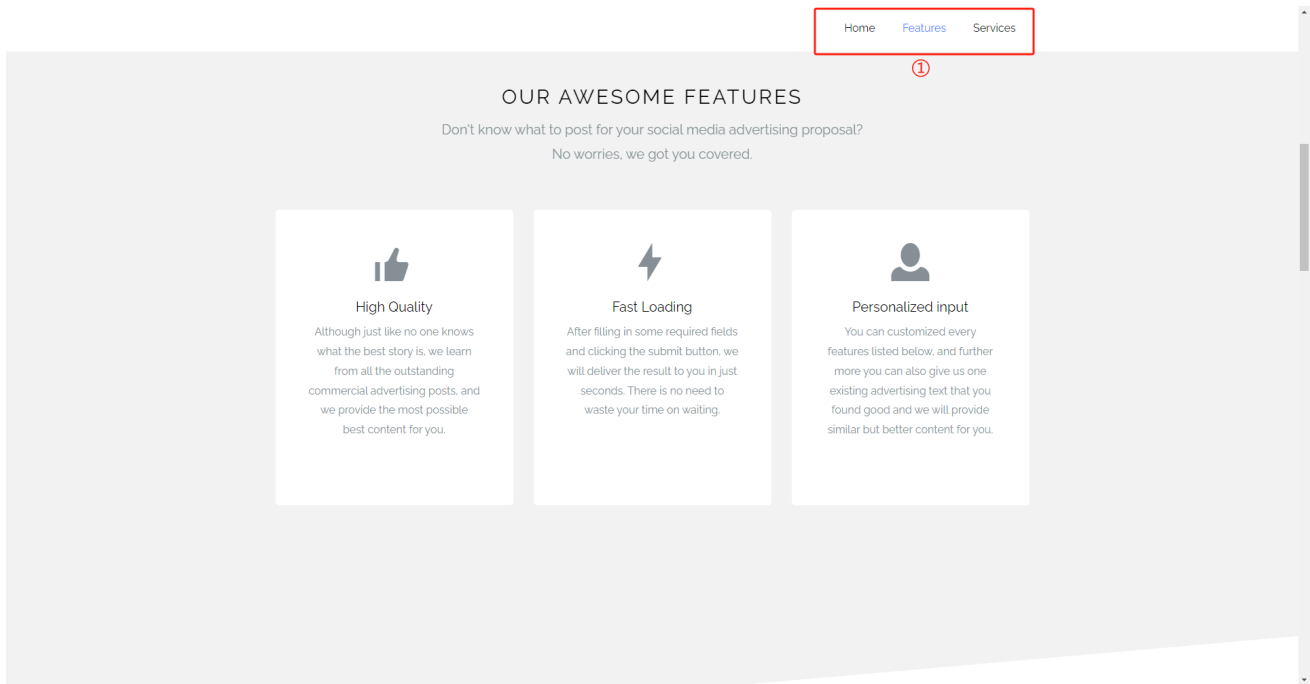


Figure 29: Features section


Here we illustrate the advantages of choosing to use Postcraft. The only thing worth mentioning is that at the top of the page, you can see our navigation bar (1) will always be following you when you browse our page.

Next, if you keep scrolling down or click on the "Services" in our navigation bar, you will go to the last section of our page which is also our core section: Services section.

[Home](#) [Features](#) [Services](#)

SERVICES

Please choose your intended category and output features below.



PRETRAINED

NEW MODEL

Categories

Select an product category ▾

Features

Content max length

No special need ▾

Sentiment number

No special need ▾

Mention count

No special need ▾

Number of hashtags

No special need ▾

Figure 30: Recommendation based on features input

For the following part of the User Guide, we will emphasize and illustrate every detail of how to use everything in our Services section.

10.4 Services

10.4.1 Part 1

This is the first function area of our page: generating recommended features and recommended text based on your input of our preset elements:

Home Features Services

PRETRAINED NEW MODEL ①

Categories ②

Select an product category

Features ③

Content max length No special need

Sentiment number No special need

Mention count No special need

Number of hashtags No special need

Word count No special need

Image count No special need

Paragraph count No special need

Interaction index No special need

Top 3 keywords

Enter top 3 keywords

SUBMIT

Recommended Features Recommended Text ④

Figure 31: Services section 1

Here you can see four major areas and one submit button. Let us go through them one by one starting from the top which is also the best order to use this part of our services.

The first component is two buttons called "PRETRAINED" and "NEW MODEL" (1), and PRETRAINED button is selected by default. By clicking on PRETRAINED button, it indicates we will be using a pre-trained model and preset database which might not be so real-time but it generates recommendations much faster. On the other hand, by clicking on the NEW MODEL, we will be using a new database that only contains data within a specific time range.

The second component is the Category input field (2). Here in the drop-down list, we prepared some product categories for you to choose from. One thing important is that choosing a category is mandatory for using this part of our service.

The third component is the Features input field (3). Here we have 8 drop-down lists and one text field, each of which indicates one type of feature that you can customize before we generate the recommendation for your intended category of products. One thing important here is that you will have to choose at least two specific requirements so that we can provide you with a

meaningful recommendation.

After all of the input fields above, you can click on the SUBMIT button and wait for seconds, and our recommendation will be generated in the area below (4). On the left, it will show you our recommended overall features: what those features should be to maximize the commercial value. On the right, it will provide you with an example of your possible successful advertising text.

One more thing worth noticing is that if you choose to use the new model, we will have two more input areas for you to fill in which indicate your intended start date and end date of our database:

The screenshot displays a web interface for a 'NEW MODEL'. At the top, there are navigation links for 'Home', 'Features', and 'Services'. Below these are two blue buttons: 'PRETRAINED' and 'NEW MODEL'. The 'NEW MODEL' button is selected. A red rectangular box highlights the 'Start Date' and 'End Date' input fields, which are both set to '年/月/日' (year/month/day) and include a calendar icon. Below the date fields is a 'Categories' section with a dropdown menu labeled 'Select an product category'. The 'Features' section contains eight dropdown menus arranged in two rows of four. Each dropdown menu is set to 'No special need'. The features are: Content max length, Sentiment number, Mention count, Number of hashtags, Word count, Image count, Paragraph count, and Interaction index. At the bottom, there is a 'Top 3 keywords' input field.

Figure 32: NEW MODEL

10.4.2 Part 2

After part 1, we will have another service that generates recommendations for the attached picture of your advertising post based on your keywords:

Home Features Services

..Or you can get the picture recommendation below

Picture keywords ①

Enter...

SUBMIT

Recommendation Output Area ②

..Or you can customize your input below

Category

Enter category...

The advertising copy

..that you found great

Type here...

Figure 33: Picture recommendation

Here the keyword input (1) can be any word related to your product, such as the name of your product, the feeling, the shape, the size... After you type in your keywords and click the submit button, we will generate our recommendation of how your attached photo should best be by describing it in words in the below area (2).

10.4.3 Part 3

After the picture recommendation service, next is the last part of our service:

Figure 34: Recommendation based on given text

Here you can type in any intended category of your product (1) and give us one advertising post that you found very good and successful (2), and both input areas are required for this part of our service. After that, you can click the submit input button and we will generate some recommended features of what your advertising text should have on the left and a possible example of a successful post on the right based on the given category and text. For this part of the service, it might take longer time than the first part of our service.

At last, all of the three parts of our services are separated from each other, which means the input for one area will not be brought to any other area. Please enjoy using our Postcraft!

11 Appendix D: Individual Project Report

Individual Report

-Liu Siyan

I initiated the idea for this project, drawing upon my extensive experience in internet marketing during my entrepreneurial ventures. My interactions with various aspects of internet marketing allowed me to identify key trends and market pain points. Through research focused on niche luxury brands, I observed a growing inclination towards leveraging the potential of micro-influencers for promotions. During this exploration, I noticed a significant gap - the absence of platforms facilitating end-to-end article generation directly, especially in the context of utilizing bottom-tier KOLs for potential promotions.

Upon proposing the idea, I took charge of designing the entire architecture of the model. This encompassed three core functions of the platform: recommendation and generation of tweet features, optimization of existing tweets, and recommendation of image content. I successfully implemented the first two models and meticulously planned the workflow for the third model. Additionally, I undertook the responsibility of data cleaning, organization, analysis, and proposed and executed the final statistical validation.

Throughout this journey, I encountered a wealth of knowledge, with the most valuable lesson being the impact of real-world messy data on modeling. Particularly in the realms of natural language processing and machine learning, I realized that the time invested in data cleaning and analysis equaled, if not exceeded, the efforts required for model construction and training.

I firmly believe that versatile approaches to handling complex datasets and patience are indispensable skills in any industrial domain. Additionally, delving into the field of Natural Language Processing expanded my understanding of text processing, from concepts like word2vec to advanced models like BERT.

The skills acquired during this project are not only pertinent to this specific context but are transferrable to various situations in professional settings. Whether it is deciphering intricate datasets or honing expertise in NLP, the knowledge gained in this project has equipped me

to navigate diverse challenges effectively. I am confident that the resilience and adaptability developed during this endeavor will serve me well in my future endeavors, making this project an invaluable learning experience.

Project Report

- Lin Zijun

I have learned really a lot during this journey with my teammates and thank them a lot for putting a lot of effort together to make our proposal for this project come true. I felt really proud of ourselves when our application was deployed successfully on the cloud server and I will not forget about my excitement when we were able to type in a URL to reach and use our application.

Personal contribution:

I contributed to all the front-end development work including coding with HTML, CSS, JavaScript, and the design of our User Interface. I also do the integration of my front-end to our back-end codes using the Flask framework. Besides the coding work, I contributed to the deployment of the Heroku cloud platform so that users could reach our project with just a URL. I also wrote the User Guide for our application.

What learned is most useful for me:

I looked back at what I have learned from our courses, I think for this project the knowledge of the recommendation system from the Reasoning Systems course was the most useful for me. During the progress of developing this application as a team, I also learned that during a team development project, teamwork, which includes more communication among team members, and trying to stick to a reasonable agenda of the development process is also very important and useful for me.

How do I apply the knowledge in other situations:

If I meet a real-life situation, for instance, in my future work, I will be more experienced and try to suggest to the team to finish the overall structure earlier and to produce an available model first and then we can spend more time on fine-tuning the model and debug our codes. For the NLP knowledge part, I will be more experienced in fetching the most plausible dataset

for a specific NLP model. I can be much faster in selecting the corresponding features for our data and constructing a reasonable model more easily.

Project Report

-Fang Ruolin

Personal contribution to group project?

In this group project, my primary responsibilities revolved around business analysis, data collection, and some data processing. I conducted an in-depth analysis of relevant research reports to extract insights into the recent user growth trends on various social media platforms. By specifically focusing on the top 10 markets for X (formerly known as Twitter), I identified a significant upward trend in daily active users (DAU). This analysis led to the conclusion that the X platform holds substantial growth potential, making it a valuable channel for brands to enhance their visibility and achieve effective market coverage. I leveraged the official X API to gather data on user followers and friends, providing valuable insights into user engagement. Additionally, I employed the Google Cloud Vision API to extract the top 5 features from images included in tweets, further enriching our dataset with visual content analysis.

What learnt is most useful for you?

I found the NLP model to be the most valuable asset in this project. Prior to starting the IRS course, I had no knowledge of NLP. However, I was thoroughly amazed by ChatGPT's ability to engage in fluent and logically coherent conversations. This experience ignited a strong interest in natural language processing for me. Throughout the three modules, especially in the Tweet Optimizer, we extensively utilized NLP techniques like Word2Vec and TF-IDF. Word vectorization and similarity calculation allowed me to gain a more profound understanding of how language information can be digitized.

How can you apply the knowledge and skills in other situations or your workplaces?

I believe NLP can be further applied to assist doctors in analyzing medical records. Across the globe, seeking medical treatment in hospitals typically involves making appointments, and a significant reason for this is the time it takes for doctors to analyze medical records and make diagnoses. With the assistance of NLP, by analyzing medical records based on historical data,

doctors' diagnostic efficiency can be significantly improved.

Project Report

-Lai Weizhih

personal contribution to group project

In this project, my main work is divided into three parts, the first part is data collection, the second part is part of the back-end code writing, and the third part is writing the report document. The first part of the data collection, at first I wanted to use the method of requests to crawl the webpage content, but in most of the community platforms have anti-crawler mechanism, which makes the data acquisition more difficult. Then I use octoparse application to crawl the post data of social platforms, through the keywords way to find the relevant post information in twitter, which includes many characteristics, interactive indicators such as the number of visits, the number of likes, the number of comments...etc., but also image url, user url, and so on.

In the second part of the backend code, I mainly implement the image feature recommendation and association rule algorithm. In the data preprocessing part, I use the character ; to determine the number of images, and sentiment analysis of the content of the post, adding feature columns. In the image feature recommendation part, firstly the group parses the content elements in the image by calling the API, based on which I use sklearn to convert the text into bag-of-words vector features, and then I train the SVM to predict the image features after converting the text input by the user, and finally I return the list of image features. For the association rule part, the cold_data is first processed, filtering some text features first, mapping the continuous features into discrete features for onehot coding. Then the apriori algorithm is used to find out the frequent items and generate association rules. After filtering the association rules by user input data, the longest combination of features is finally found out, and this combination of features is compared with the data set before onehot for the similarity calculation to find the most similar rows, and then the missing columns of features are filled in and returned. For the final model validation, I wrote the code for the validation part and the coefficients of the four features Chart Code.

In the final part of the report, I mainly write the system design and some model introduction

documents.

what learnt is most useful for you

I have gained a lot from this project, including knowledge, technology and documentation. In terms of knowledge, since building an intelligent system is a big project that I have never touched in the past, I have a deeper understanding of the complete framework of the system, including from data acquisition to the back-end processing of text data to the front-end structure, and when I was writing the system design document, I learned about the construction of a complete system, the communication between the front-end and the back-end, as well as the design of the system needs to be taken into account in the part. In particular, I have a better understanding of the front-end framework, even though I didn't participate in the html, CSS, and JavaScript code, I have a deeper understanding of the overall framework.

Technically, I have used machine learning algorithms many times in the past, including SVM in this case, but this is the first time that I have applied nlp or recommendation algorithms in my project, and it has deepened the scope of my application of data, and made me understand the value of data development. In the past, I have mostly worked with numerical values or images, but the concepts used in this project, such as TFIDF, Apriori algorithm, and onehot encoding, are all new to me. In addition, the recommendation system has been developing rapidly in recent years, so I believe that recommendation algorithms are a very important skill to cultivate.

I also gained a lot from writing documents, including how to present our products, the application of latex syntax, and the construction of flowcharts, all of which can help us present our projects in a more complete and clearer way. The last but not the least, I learned the importance of team communication and cooperation. Having the right team leader to organize meetings and monitor the progress is also essential in the project, and to communicate with team members when there are problems in the cooperation, as well as to solve the tough problems with high quality and efficiency through teamwork. Finally, I would like to thank all the team members for their contribution to this project.

how you can apply the knowledge and skills in other situations or your workplaces

There are many knowledge and skills that can be utilized in the future work environment. Communication and teamwork skills are indispensable in the workplace, and honesty is extremely

important, not to hide or cheat. In the future, if I encounter the need to build nlp or recommendation systems, I will be able to better understand the process of system design, from the pre-processing of data to the front-end concepts presented to the user. In the text data processing section, I can utilize many techniques to uncover relationships between data, such as similarity computation, concordance filtering, and association rule analysis. Having a holistic understanding of intelligent systems will expand my perspective and approach to various tasks and projects in my future endeavors.

12 Literature Cited

References

- [1] Esteban Ortiz-Ospina. The rise of social media. *Our World in Data*, 2019.
<https://ourworldindata.org/rise-of-social-media>.
- [2] Isabel Anger and Christian Kittl. Measuring influence on twitter. In *Proceedings of the 11th international conference on knowledge management and knowledge technologies*, pages 1–4, 2011.
- [3] Mr B Chandra Sekhar and Suja S Nair. Impact of social media on ethical values in the society. *Journal of Pharmaceutical Negative Results*, pages 3071–3079, 2022.
- [4] John Scott. Trend report social network analysis. *Sociology*, pages 109–127, 1988.
- [5] Kailash Choudhary and Ruby Beniwal. Xplore word embedding using cbow model and skip-gram model. In *2021 7th International Conference on Signal Processing and Communication (ICSC)*, pages 267–270. IEEE, 2021.