Recommendation Strategy of Kuaishou to Enhance User Community Engagement

Haoyu Liu[1], Mengjie Chen[2], Sameen Su[2], Yuancheng Cao[3]

[1]Jacobs School of Engineering, UC San Diego
[2]Rady School of Management, UC San Diego
[3]Halıcıoğlu Data Science Institute, UC San Diego

ABSTRACT

This report delves into the recommender system of Kuaishou, a social media platform celebrated for its user-generated content. Our analysis centers on models such as Alternative Least Square, Gradient Boosting, Linear Regression, Random Forest, Autoregression, and Time-series-based Prediction, enhancing the platform's capacity to offer tailored friend and video suggestions. Beyond static algorithms, our report introduces a real-time adaptation mechanism, dynamically adjusting recommendations based on evolving user preferences. Our testing results reveal varying accuracies across different models: a Mean Squared Error (MSE) of 0.0022 for video recommendations, 53% accuracy for new friends suggestions, 81.49% accuracy for daily video play counts forecasting, 87.4% accuracy in predicting user retention, and 65% accuracy for video recommendations that consider temporal dynamics and user behavior sequences. The purpose of our research is to explore how Kuaishou's innovative recommender system enhances user engagement and penalization within its digital community, using data-driven analysis and model insights.

INTRODUCTION

Kuaishou is the first short video platform in China. It's similar to TikTok (known as Douyin in China) and offers a platform for users to create, share, and discover videos. Established as a short-form video-sharing app, Kuaishou has cultivated a vibrant ecosystem where users seamlessly contribute self-edited video works, fostering a diverse and engaging content milieu. The app includes a variety of features such as video editing tools, filters, and effects, allowing users to easily create engaging content. Kuaishou also supports e-commerce functions, enabling users and influencers to sell products directly through the platform.

The recommendation algorithm is crucial in Kuaishou, a short video platform, because it helps personalize user experience by suggesting content that aligns with individual preferences and viewing habits. This enhances user engagement and retention, as viewers are more likely to stay on the platform when presented with videos that closely match their interests.

Figure 1: *The Kuaishou App Video Platform*

The effectiveness of these algorithms has been a key factor in Kuaishou's current standing, where it now holds a significant place in the competitive landscape of short video platforms. Despite macroeconomic challenges and slow recovery of marketing customer confidence, Kuaishou maintained high-quality user growth and service expansion in Q3 2023. However, continuous efforts are still needed to promote and distribute high-quality original content and engage existing users efficiently, according to Kuaishou Technology Q3 2023 Results Conference [1].

Therefore, we set our goal to increase user engagemant measured by a series of metrics, including video plays, likes and comments, and the number of users becoming video content contributers. We attempted to establish a number of models towards the improvements of these metrics in several ways.

1. Recommend videos that better match user preferences, increasing the likelihood of user engage-

ment through clicks.
2. Suggest friends whom a user may know or is likely to follow, fostering community interactions.
3. Predict the future play counts of a video, aiding in the refinement of content recommendations and enhancing understanding of video popularity trends.
4. Understand how a user's interests evolve over time, allowing us to dynamically adapt our recommendation strategy.

SOURCE OF DATA AND LITERATURE REVIEW

KuaiRec, the data set we used for this project, is a real-world data set collected from the recommendation logs of the video-sharing mobile app Kuaishou. All interaction records are collected from July 5th, 2020 to September 5th 2020 on Kuiashou App. The large data set consists of 1,411 users and 3,327 videos, with each element representing a user's feedback on a video. There are some additional information provided for further analysis, summarized below with respect to each table:

- **big matrix.csv and small matrix.csv**: Includes the user and video id's, the play duration of each video, and the timestamp of watchings.
- **social network.csv**: The friend ID list of every user.
- **item categories.csv**: The list of tags of each video.
- **item daily features.csv**: The features of each video, including the time duration, background music, and upload date. Other statistics are also available, such as the play count, number of exposures, number of "likes" and "comments" received, etc.
- **user features.csv**: The features of each user, such as whether the user is a video author, the number of followers or fans, and the days since he/she registered Kuaishou.

The KuaiRec dataset is probably the first fully observed data on user-item interactions [2], *i.e.* every user has viewed every video and left feedback. Using this dataset, the Kuaishou team conducted the multi-round conversational recommendations [3] based on the adaption from user feedback in each round. Besides the user-item interaction history [4], additional features of videos and users themselves can provide a basis of personalized recommendations. Throughout the multi-round experiment, the item exposure strategy also has a significant influence on the outcome of conversational recommender system [5].

Recent studies on recommendation algorithms have explored diverse datasets and advanced methods to boost user engagement. One significant approach involves formulating the problem of short video recommendation as a constrained Markov Decision Process (MDP), with a focus on optimizing the main goal of user watch time while also accommodating auxiliary responses such as sharing or downloading videos [6]. This method contrasts with traditional recommendation systems that optimize a single objective, highlighting the multifaceted nature of user feedback in social media. [6]. The field is shifting towards more dynamic and multifaceted user engagement models, moving beyond single-objective optimization to a comprehensive approach that balances multiple aspects of user-platform interaction.

As a proceeding of existing studies about short video platforms, our project aims to explore more mechanism to elevate overall activeness of Kuaishou's online community, which will bring extra traffics, increased item exposures and more user-item conversations to facilitate the resource reallocation in multi-round recommendations.

EXPLORATORY DATA ANALYSIS

We analyzed the community status quo in two ways: video topic preferences and user activeness.

Popularity of Video Tags

We used word clouds to visualize video tags where the size of each word indicates its frequency or importance in the context of different metrics: likes, comments, and shares.

The likes-based word cloud as shown in Figure 2 emphasizes which tags are most popular among users, words like, "Movie", "Cat", and "Celebrity" are prominent, which could imply these topics resonate well with the audience or are associated with enjoyable content.

The most striking words are consistent across the three clouds, like "Movie" and "Celebrity", indicating that these topics not only generate a lot of likes and comments but are also shared frequently. The shares-based word cloud shown in Figure 4 features "Healthy" and "Legal Knowledge" as larger words compared to the other two, which might indicate that users are inclined to share content they find useful or educational.

The bubble plot shown in Figure 5 represents the like count associated with each tag corresponding to the play count. On the x-axis, "Total Play Count" indicates

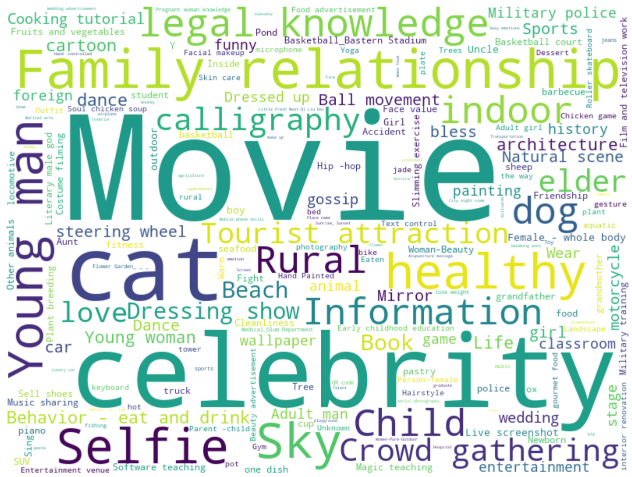

Figure 2: *Word Cloud Based on Likes Count*

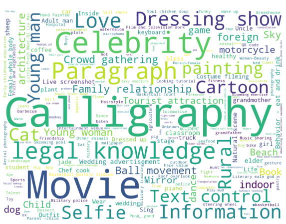

Figure 3: *Word Cloud Based on Comments Count*

Figure 4: *Word Cloud Based on Shares Count*

how many times content tagged with each category has been played, and the y-axis represents the "Complete Play Count" suggesting how often these plays were watched from start to finish. The positioning of the tags provides insights into user engagement with different types of content. In the plot, tags such as "front face", "movie", and "meme" appear to have a high total play count as well as a high complete play count, indicating that content with these tags not only gets played often but also watched thoroughly. Other tags like "celebrity", "backside", and "family relationship" have a smaller bubble size, suggesting fewer likes compared to the others, with varying degrees of total and complete play counts.

The bar chart shown in Figure 6 illustrates the top 10 video labels based on their total play counts in the dataset, providing insights into user preferences and content consumption patterns. The total play counts are presented in billions. The data showcases the popularity of various video tags, with "Front Face" leading the pack followed by categories such as "Movie," "Side Face," and "Meme." One Possible reason for the popularity of the tag "Front Face" could be the large number of videos posted in this tag. Similarly, The bar chart

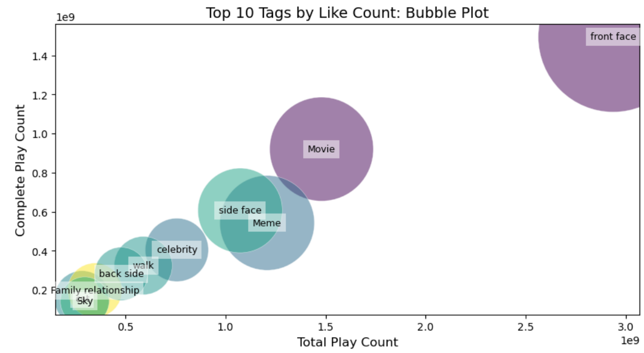

Figure 5: *Likes Count of Tags based on Complete Play Count*

shown in Figure 7 presents the the top 10 video tags based on their average play counts of each single video in this tag. The chart showcases the most engaging video tags, including 'Magic Tutorial,' 'Cup,' 'Street Fitness,' 'Rice,' 'Wrestle,' 'Software Tutorial,' 'Watermelon,' 'Eating,' 'Steering Wheels,' and 'Tomato.'

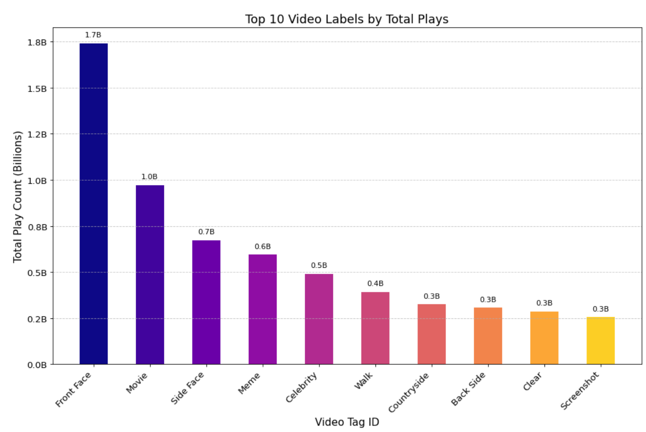

Figure 6: *Top 10 Video Tags by Total Play Counts*

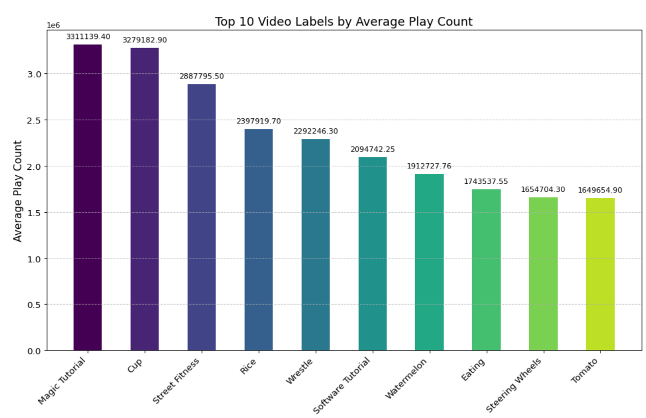

Figure 7: *Top 10 Video Tags by Average Play Counts*

Engagement of Different User Types

The distinctive appeal of the Kuaishou App is derived from the originality and creativity embedded in its content. A myriad of self-media contributors and ordinary users actively enrich the platform's content diversity through the submission of self-edited video works. The act of becoming a video author is evidently consequential for a user's engagement with Kuaishou. Figure 8 classifies users into "video authors" and "non-authors," enabling a comprehensive comparison of various metrics related to their interactions and activity within the application.

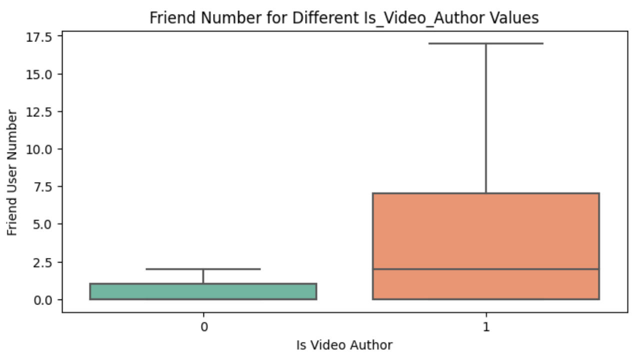

Figure 8: *Number of friends for video authors and non-authors*

The quantitative assessment of user engagement extends to the examination of friendship networks. Upon assuming the role of a video author, individuals experience a marked expansion in their social network, a phenomenon that underscores the substantial impact of content creation on the user's broader social connectivity within the Kuaishou community.

Additionally, our analysis encompassed the calculation of the number of days since registration for both the "Author" and "Non-author" groups, shown in Figure 9. Significantly, the "Author" group exhibited a tendency toward a longer duration since registration. This observation suggests that assuming the role of a video author contributes to an extension in user retention on the Kuaishou App. The implication is that engaging in content creation serves as a catalyst for cultivating a sustained, longer-term customer relationship and heightened user engagement within the application ecosystem.

Beyond video authorship, the temporal dimension, particularly the chosen time slot, exerts a discernible influence on overall user activity. As depicted in Figure 10, the distribution of play counts varies across different weekdays, with weekends exhibiting larger spans and higher peak values compared to weekdays. This trend

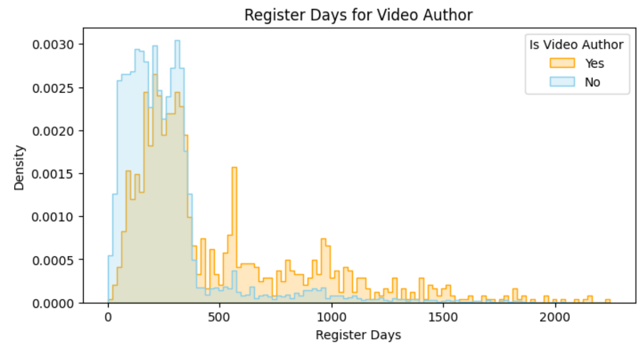

Figure 9: *Register dates of authors and non-authors*

implies a pronounced inclination in resource allocation towards weekends.

In summary, our findings highlight the interplay between user engagement, authorship, and temporal patterns. To enhance Kuaishou community engagement, strategic emphasis should be placed on encouraging authorship and optimizing time slot allocation, recognizing the significance of these factors in shaping a more robust and dynamic user interaction landscape.

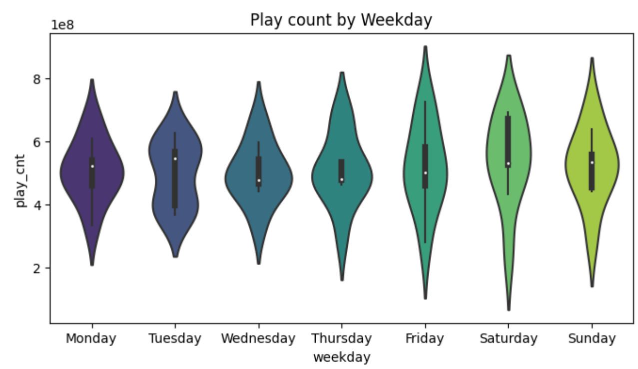

Figure 10: *Play count distributions on each weekday*

Temporal Data Analysis

In the exploration of time-dependent trends within the dataset, We plot two time-series line charts to shed light on the dynamics of video engagement over days after the videos are posted.

The first chart, depicted in Figure 11, is a comprehensive time-series analysis unfolds with a dual-line graph capturing the total play counts and total like counts after video posting. The data comprises 6,106 videos posted between July 5, 2020, and August 19, 2020. It provides insights into the sum of the play counts and like counts of all the videos during the initial 10 days following their respective posting dates, offering a comprehensive overview of engagement trends during this period. The chart provides a holistic view of how both play and like counts fluctuate over time. As we can see, the daily play counts are usually much higher on the first two days after the release of a video, especially on the second day, and gradually decreases after that. Also, the change of the daily like counts follows the same trend.

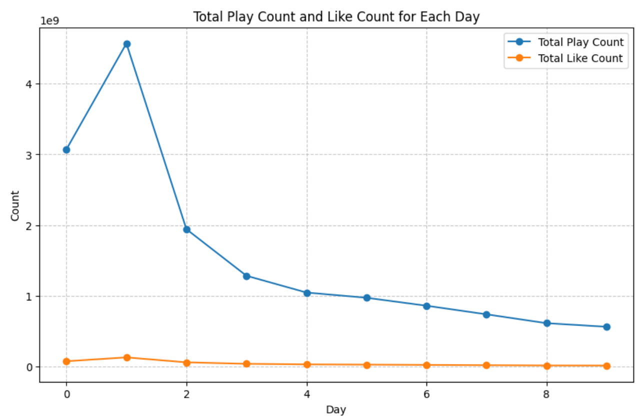

Figure 11: *Total Daily Play Counts and Like Counts After Video Posted*

Moving to Figure 12, which focuses on specific video labels and their respective daily play counts after videos under this tag have been posted. We select 10 video tags that have the most total play counts, including 'Front Face,' 'Movie,' and 'Side Face,' and individually tracked across days to unveil patterns in user interaction. Each line on the chart corresponds to a distinct label, showcasing how play counts evolve over time since the videos' initial posting dates and the nuances between the trend for different video tags.

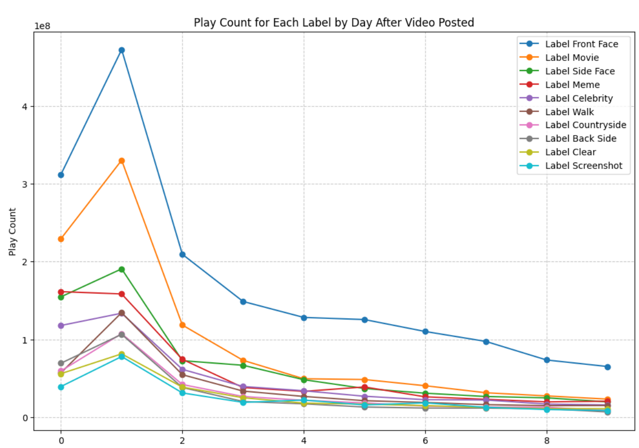

Figure 12: *Daily Play Counts by Tags After Video Posted*

Together, these time-series visualizations offer a nuanced exploration of temporal patterns within the dataset, facilitating a deeper understanding of user preference and recommendation system behavior over time.

MODELING

Model 1: Recommendation of Videos Based on User's Watch Ratio

In the age of digital media, personalized content delivery has become crucial for enhancing user experience. In the context of Kuaishou, recommendation systems are vital in helping users discover content tailored to their interests, which increases their engagement and satisfaction. By leveraging the watch ratio variable, we could identify what kinds of videos a user is interested in and predict and recommend videos that align with the user's preferences.

In our dataset, we chose user ID, video ID, and watching ratio as our features from the small matrix.csv which contains 4,676,570 rows in total. We narrowed our focus to include only the dates between July 5th 2020 and July 15th 2020 for the following analysis. The 3D scatter plot as shown in Figure 13 visualizes the relationship between user ID, video ID, and watch ratio. There are clusters of points at different depths, indicating that certain videos have a wide range of watch ratios among different users. The color gradient shows that most watch ratios are on the lower end of the scale, with fewer points with a high watch ratio. The plot hints at the interactions between users and videos and which user is more likely to watch which types of videos.

Feature Engineering For this recommendation model, we employed the Singular Value Decomposition (SVD) algorithm for its core logic. We constructed a matrix $R_{i,j}$ representing the watch ratios of user i of video j. Then we applied SVD to decompose the utility matrix $R_{i,j}$ into three matrices:

$$R_{i,j} = U_{i,i} \Sigma_{i,j} V_{j,j}^T \tag{1}$$

where U represents the user features matrix, and each row gives a vector of latent features for a user. Σ is a diagonal matrix containing singular values that scale the latent features, and V^T represents the video features matrix, where each column gives a vector of latent features for a video [7].

Recommendation Model We utilized the surprise library to employ the SVD algorithm [8]. For data prepa-

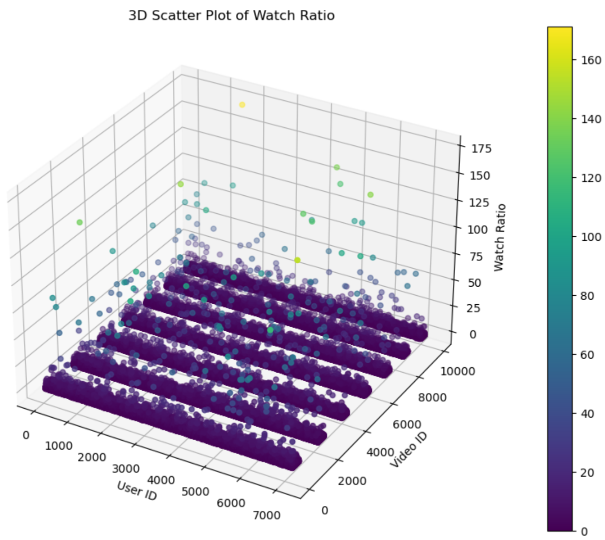

Figure 13: *relationships between user ID, video ID, and watch ratio*

ration, we normalized the watch ratio variable to ensure that the input data conforms to a standard scale. Then we define a Reader object corresponding to the normalized watch ratio, which was then used to load the dataset into the surprise library's data structure. For model training and testing purposes, we split the dataset into training set and test set, with 75% allocated for training and the remaining 25% for testing. We used a grid search to identify the optimal hyperparameters for SVD, and used Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to measure the accuracy of the model. Upon completion of the grid search, the best performing hyperparameters were reported, and the trained model was used to predict the watch ratio for the test set.

To apply SVD to predict the watch ratio for a user-video pair, we multiplied the user's feature vector from U_k, the diagonal scaling matrix Σ_k, and the video's feature vector from V_k^T. The resulting matrix $\hat{R} = U_k \Sigma_k V_k^T$ was the complete matrix with predicted watch ratios for all user-video pairs. The predictions were then assessed using the RMSE and MAE metrics to determine the model's accuracy in a real-world scenario.

After fine-tuning our model's hyperparameters, the optimal settings and performance metrics were as shown in Table 1.

With the optimized hyperparameters, a number of factors at 50, epochs at 30, a learning rate of 0.005 and a regularization term of 0.1, the model demonstrated relatively good results, with a RMSE of 0.0079 and MAE of 0.0022, indicating a relatively high degree of

Parameter	**Value**
Number of Factors ($n\_factors$)	50
Number of Epochs (n_epochs)	30
Learning Rate (lr_all)	0.005
Regularization Term (reg_all)	0.1
Metric	**Score**
Root Mean Square Error (RMSE)	0.0079
Mean Absolute Error (MAE)	0.0022

Table 1: *Optimal Hyperparameters and Performance Metrics*

accuracy in predicting user video preferences.

Model 2: Recommend New Friends to a User

Social network is an essential channel for information transmission. Users are more likely to watch and like video contents that their friends like, and Kuaishou's recommender system has been working on feeding users with contents that has been liked by their friends. This information sharing mechanism promotes the expansion of excellent self-created works.

In our dataset, as illustrated in Figure 14, the Kuaishou friend connection network exhibits predominantly isolated clusters with minimal inter-cluster connections, indicating a fragmented social landscape. This visual underscores the imperative to bolster community cohesion by fostering more friendships. Therefore in this section, we explored a model to recommend more friends to a user based on the similarity of their preferences.

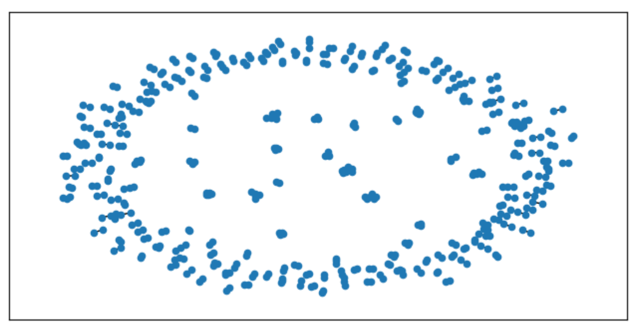

Figure 14: *Current friend network graph*

Feature Engineering For this prediction task, we created features using Bayesian latent factors on user watching history. To store the video watching history of each user, we constructed an $n_{user} \times n_{video}$ sparse matrix R, with the (i,j) entry as the video watch ration of user i at video j.

$$R = \begin{pmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m,1} & r_{m,2} & \cdots & r_{m,n} \end{pmatrix}$$

This matrix is then fitted into Alternative Least Square model, and we extracted user latent factors with $n = 100$ level. Other features representing a user's engagement on Kuaishou, such as "is video author" "active degree" are also included in features. Now the feature vector of user i consists of

$$u_i = [UserLatent_i, ItemLatent_i, 0, 1, 0 \cdots] \quad (2)$$

Having composed user feature vectors, we fitted them with the label about whether a pair of user are friends. Below is our comparison of several models.

Prediction Model Table 2 compares the accuracy of several models after 5-fold cross-validations. The performance, in terms of classification accuracy, is not outstanding compared to model 1. This outcome is partly caused by the incomplete friend network data under the isolated cluster characteristic in the data set.

Model	Accuracy
XGBoost	0.530
Logistic Regression	0.508
Random Forest	0.455

Table 2: *Accuracy of friend prediction models*

Though the incomplete data limits the performance upgrade of prediction models, existing features and models can be a temporary option to recommend new friends. Features can be updated and re-trained upon receiving the first-round user feedback.

Model 3: Predict Daily Play Counts of a Video

Predicting the daily play counts of a video is a crucial task with profound implications for content creators, platform administrators, and businesses operating in the digital video space. It helps in refining algorithms for content recommendations, improving search functionality, and tailoring the overall platform experience based on user preferences and video popularity trends. In this section, we target to build a time-series-based model to predict the daily play counts of a video based on its play counts in the past few days.

Feature Engineering and Prediction Model A The data set we used encompasses 60619 entries on 6106 videos posted between July 5, 2020, and August 19, 2020. Each entry represents data collected on a specific day within the period from the video's posting date until August 28, 2020. Our initial prediction task is to predict the next day's play counts based on the play counts of any five days.

The first model we built is an autoregressive feature representation model to uses temporal information. The autoregressive feature function extracts the play counts of a video for the last five days leading up to the target day. This representation aims to encapsulate the recent engagement history of the video, enabling the model to discern trends and patterns.

For each entry of the data set, we processed to get the autoregressive features. These features were then divided into training and testing sets. The training set, constituting 80% of the data, was used to train the model.

For the prediction model, a linear regression model and a random forest regression model are used. The baseline model we used calculates the average play counts of the past 5 days.

Particularly, for the linear regression model, its coefficients (theta) are derived as [0.46223195, -0.01001723, 0.04956376, 0.03349819, 0.0364957], showing that play counts of the most recent day is most useful in predicting the next day play counts. Model performances were assessed using R-squared (R2). The results in Table 3 indicate that the best R-squared value we get is 0.2048, presenting that while the model provides some predictive capability, the relatively high errors and modest R-squared suggest that there may be room for improvement. Possible avenues for refinement could include exploring more sophisticated models, incorporating additional relevant features, or refining the feature engineering process.

In order to identify possible reasons for the limited performance, we deeply analyzed the characteristics of the dataset and the distribution of the play counts and found the randomness of the distribution as a possible reason. The autoregressive feature representation provides temporal context but assumes that historical play counts are sufficient for prediction. However, after a video is published, its daily plays do not vary in a linear or regular manner. The number of plays may show a huge increase or a sudden decrease on a certain day long after the release, which leads to the difficulty of building an accurate prediction model to predict the number of plays on any given day. However, we found that the change is more regular within the first 10 days of a video being posted. Therefore, we proposed the second prediction model for play counts.

Model	R-squared (R2)
Baseline Model	-0.0058
Linear Regression Model	0.2274
Random Forest Model	0.2048

Table 3: *Prediction Model Performances*

Feature Engineering and Prediction Model B Our refined prediction task is to predict the play counts on day 6 and 7 based on daily plays on days 1-5 after the publication of the video. The features we used are the daily play counts of the first 7 days of each video. The features were then divided into training and testing sets. The training set, constituting 80% of the data, was used to train the model.

Similarly, We built a linear regression model and a random forest regression model, and the baseline model calculates the average play counts of the first 5 days.

For the linear regression model, its coefficients (theta) for predicting day 6 are [0.00999439, 0.01252573, 0.00837846, -0.04405089, 0.81881996], and day 7 is [0.01292557, 0.0199063, 0.01823883, -0.00342771, 0.55917815], indicating that the play counts of day 5 are most insightful for predicting both day 6 and day 7 play counts. Model performances evaluated by R-squared (R2) are shown in Table 4.

Model	R-squared (R2)
Baseline Model	-4.8062
Linear Regression Model	0.7784
Random Forest Model	0.8149

Table 4: *Prediction Model Performances*

The Baseline Model yields a notably negative R-squared, indicating its limited ability to capture the variance in data. In contrast, both the Linear Regression and Random Forest Models exhibit positive R-squared values of 0.7784 and 0.8149, respectively. These positive values suggest a more meaningful ability of the latter two models to explain and predict the daily play counts, with the Random Forest Model demonstrating slightly superior predictive performance. The positive R-squared values signify that the chosen models are capable of explaining a significant proportion of the observed variance in the daily play counts, demonstrating the effectiveness of the model in predicting the plays for the next two days based on the first 5 days' data.

Model 4: Predict User Retention

In the dynamic world of digital platforms, accurately predicting user retention is key, as it significantly influences long-term user engagement and satisfaction. This task entails estimating the probability of users revisiting the platform after interaction with certain content types, which is essential for customizing the user experience and maintaining continuous engagement.

Feature Engineering To effectively analyze user behavior and preferences on the platform, we started by focusing on the aggregation and transformation of our datasets. In the 'user_related_features' dataset, we aggregated data by 'user_id' and 'date', summed up 'play_duration' and calculated the average 'watch_ratio'. This approach is essential since it accounts for users who may watch multiple videos in a single day, providing a consolidated view of their daily activity. Similarly, in the 'content_related_features' dataset, we summed up metrics like 'show_user_num', 'like_user_num', 'comment_user_num', and others for each video. Then, these datasets were merged on the 'date' field to align user activities with content interactions on the same day.

To further refine our analysis, we computed ratios such as 'like_to_show_ratio' by dividing each engagement metric by 'show_user_num'. This normalization helps in comparing videos with differing levels of popularity by adjusting for the number of users who have viewed the video. Additionally, we calculated an 'engagement_score', assigning equal weight (0.25) to 'like_user_num', 'comment_user_num', 'share_user_num', and 'follow_user_num'. This score is a composite metric reflecting overall user engagement with the contents.

Our final feature matrix, including 'engagement_score', 'watch_ratio', 'like_to_show_ratio', 'comment_to_show_ratio', 'share_to_show_ratio', 'follow_to_show_ratio', 'report_to_show_ratio', and 'reduce_similar_to_show_ratio', is designed to capture a holistic view of user engagement and interactions with the platform's contents. Each feature provides unique insights into user behaviors and preferences, vital for understanding and predicting user retention. Our prediction label is binary: '1' denotes users who have watched any video between August 20 and August 23, 2020, while '0' represents those who did not engage in this period.

Prediction Model The accuracy of the baseline model using a Decision Tree Classifier was not very high, which is likely due to several factors inherent in the approach and dataset. Initially, the work involved a synthetic dataset where 'play_duration' and 'watch_ratio' were randomly generated, and 'retention' was determined based solely on whether the 'watch_ratio' exceeded a threshold of 10. This simplistic criterion for defining retention might not have accurately reflected real-world user behavior, ignoring the complexity and nuances of actual user engagement. Furthermore, since all ratios in the feature matrix were also randomly generated, they may not have represented meaningful relationships or patterns that a Decision Tree could learn from. Real user interaction data often contains intricate and subtle patterns that random data cannot replicate. Considering these factors, it wasn't surprising that the accuracy was around 0.5048, which is only slightly better than random guessing in a binary classification task.

The improved model selection, utilizing a Gradient Boosting Classifier, demonstrated a significant increase in accuracy, reaching 0.874. The dataset was split into training and testing sets, with 80% of the data used for training and the remaining 20% for testing. The same features as those in the baseline model were used, but the Gradient Boosting algorithm, with its ensemble method, is typically better at handling variances and complexities in the data. Additionally, the chosen parameters for the Gradient Boosting Classifier, such as n_estimators=100, learning_rate=0.1, and max_depth=3, contributed to its performance. These settings helped the model balance between learning from the training data and generalizing to new data, thus avoiding overfitting.

Feature	**Importance**
watch_ratio	0.998075
engagement_score	0.001142
like_to_show_ratio	0.000252
follow_to_show_ratio	0.000242
comment_to_show_ratio	0.000189
share_to_show_ratio	0.000101
report_to_show_ratio	0.000000
reduce_similar_to_show_ratio	0.000000

Table 5: *Feature Importance Rankings for GBM*

Table 5 presents the feature importance scores from the Gradient Boosting Model. It reveals that 'watch_ratio' is by far the most influential feature, with an importance score of 0.998075, indicating it is the primary driver in predicting user retention. Other features like 'engagement_score' and 'like_to_show_ratio' hold some significance but have considerably less impact. Features such as 'report_to_show_ratio' and 'reduce_similar_to_show_ratio' have zero importance in this model, suggesting they do not contribute to the prediction of user retention in the current setup. This table helps in understanding which aspects of user behavior are most critical in determining their likelihood of returning to the platform.

Model 5: Temporal Information-based Video Recommendation

In order to explore how time series can be utilized in video recommendation, we present a video recommendation model leveraging temporal information, specifically the Factorized Personalized Markov Chains (FPMC) algorithm. Our primary objective is to enhance the accuracy of video recommendations by incorporating temporal dynamics and user behavior sequences.

Feature Engineering The dataset utilized comprises 600,000 entries, each representing information on user interactions with videos between July 5, 2020, and August 19, 2020. Key features include user and video IDs, timestamps, and watch ratios. The data pre-processing procedures involved creating dictionaries for users, videos, and interactions. The historical interaction features were split into training and testing set in a ratio of 8:2.

Recommendation Model There are several reasons for selecting FPMC model for time-series-based video recommendation. Firstly, FPMC takes into account the user's personalized characteristics, such as implicit factors, to make recommendations more personalized and accurate. It can recommend items that may be of interest to a user based on their historical behavior and personalized preferences. What's more, FPMC uses Markov chains to model user behavioral sequences, making it able to capture dynamic changes in user interests. This means that the model can adapt to changes in user interests, providing more real-time and accurate recommendations. We implemented the FPMC model with TensorFlow and optimized the parameter using the Adam optimizer. To examine the model's ability to capture temporal dynamics in user-item interactions and evaluate whether the inclusion of the term associated with the previous interaction improves the model's predictive power over time, we analyzed the temporal patterns in the learned latent features. Figure 15 is an example of visualizing the temporal patterns in terms of latent features between users and videos.

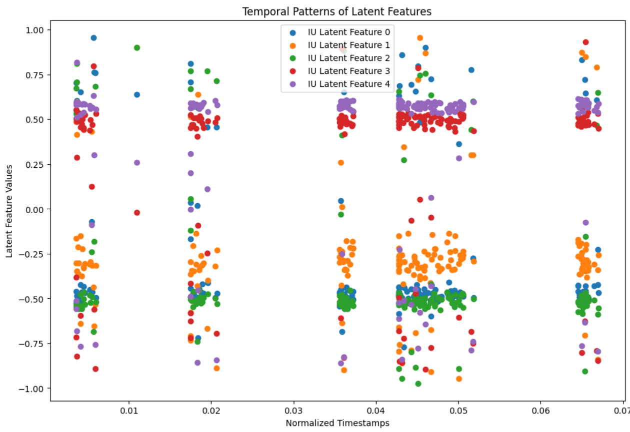

Figure 15: *Temporal Patterns of Latent Features*

Model Evaluation and Results To evaluate the predictive task, we employed a training process with 100 iterations, tracking the objective function's progress. After the training, our model has a loss of 0.1908 on the testing set. Given a user and a video, the model is able to calculate predicted scores for different videos, which are used to rank videos for recommendation. Videos with higher predicted scores are considered more likely to be of interest to the user. We used traditional user-based collaborative filtering as the baseline model. We assessed the validity of our model's predictions by checking that if any of the top 10 recommendations appeals in our dataset after the current video and has a watching ratio over 50% (Table 6). The strengths of FPMC include its ability to capture temporal dynamics, incorporate latent features, and handle sequential user-item interactions. However, limitations may arise in scenarios with sparse data or when the temporal aspect is less critical. Additionally, the model's performance heavily relies on the quality of the temporal information and the effectiveness of latent feature representations.

Model	**Accuracy**
Collaborative Filtering Baseline Model	0.53
FPMC Model	0.65

Table 6: *Model Performances*

RESULTS AND CONCLUSION

In our study, we analyzed data from kuairec.com, focusing on a large dataset with 7,176 users and 76,984,128 interactions, and a subset of this - a 'small matrix' consisting of fully observed data involving 1,411 users and 4,694,397 interactions. Our exploratory analysis delved into video topic preferences and user engagement metrics. We investigated the popularity of video tags, the relationship between play counts and like counts for tags, identified the top 10 video tags, and examined the registration duration distribution among authors and non-authors. We also analyzed the daily distribution of play counts and conducted a time series analysis on the daily total play counts and like counts following video postings. We also modeled Kuaishou's vibrant self-edited video community by enhancing both recommendations and predictions, affecting both video authors and audiences.

- To incentivize video content creators, we increased the exposure of their videos to potentially interested audiences using the SVD algorithm.We further enhanced our video recommender by involving a temporary information-based algorithm. Daily play counts of videos are predicted by linear regression and random forest models.
- To enhance the cohesion of the user community, we recommended new friends based on common preferences and predict user retention or churn rate.

For each task, we developed corresponding models and compared them to the baselines. Most models demonstrate improved performances, except for the friend recommendation model due to insufficient available samples in our dataset. Performances of our best models are summarized in Table 7.

Model	**Performance**
Video Recommender	RMSE = 0.0079
Friend Recommender	Accuracy = 53.0%
Video Play Counts Predictor	R^2 = 0.8149
User Retention Predictor	Accuracy = 87.4%
Temporal Video Recommender	Accuracy = 65.0%

Table 7: *Best Performances Summary*

Discussion

While our models demonstrated effectiveness in enhancing user engagement and content personalization, there exist several areas where additional actions can be taken to potentially achieve further improvements.

- Expanding Data Sources: Due to the vastness of our dataset, we currently only incorporate data within a specific timestamp range. To enhance the accuracy and adaptability of our models, future investigations could involve the inclusion of a broader range of data. This expansion could not only provide a more comprehensive understanding of user behavior over extended periods but also help identify long-term trends, yielding more robust results for our models.
- Incorporating User Feedback: Integrating mechanisms to actively capture and respond to user interactions and feedback in real time could assist our models in adjusting future recommendations to better align with user preferences. For short-video platforms like Kuaishou, dynamic adjustments are crucial as user preferences and content trends constantly evolve.
- Enhancing Algorithmic Complexity: The combination of multiple models, such as Random Forests or Gradient Boosting Machines, may improve prediction accuracy by mitigating variance and bias. Experimenting with hybrid models or exploring more sophisticated deep-learning algorithms could potentially lead to enhanced performance.

References

[1] SA Transcripts. *Kuaishou Technology (KUASF) Q3 2023 Earnings Call Transcript*. Accessed: November 29, 2023. 2023. URL: `https://seekingalpha.com/article/4653566-kuaishou-technology-kuasf-q3-2023-earnings-call-transcript`.

[2] Chongming Gao et al. "KuaiRec: A Fully-Observed Dataset and Insights for Evaluating Recommender Systems". In: *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. Atlanta, GA, USA, 2022. DOI: `10.1145/3511808.3557220`.

[3] Kerui Xu et al. "Adapting User Preference to Online Feedback in Multi-Round Conversational Recommendation". In: Virtual Event, Israel: Association for Computing Machinery, 2021, pp. 364–372. ISBN: 9781450382977. DOI: `10.1145/3437963.3441791`.

[4] Julian McAuley. *Personalized Machine Learning*. Cambridge University Press, in press.

[5] Jin Huang et al. "Keeping Dataset Biases out of the Simulation: A Debiased Simulator for Reinforcement Learning Based Recommender Systems". In: *Proceedings of the 14th ACM Conference on Recommender Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 190–199. DOI: `10.1145/3383313.3412252`.

[6] Qingpeng Cai et al. "Constrained reinforcement learning for short video recommendation". In: *arXiv preprint arXiv:2205.13248* (2022).

[7] Michael E Wall, Andreas Rechtsteiner, and Luis M Rocha. "Singular value decomposition and principal component analysis". In: *A Practical Approach to Microarray Data Analysis*. Ed. by D.P. Berrar, W. Dubitzky, and M. Granzow. Norwell, MA: Kluwer, 2003, pp. 91–109. URL: `https://public.lanl.gov/mewall/kluwer2002.html`.

[8] Nicolas Hug. "Surprise: A Python library for recommender systems". In: *Journal of Open Source Software* 6.61 (2021), p. 3067.