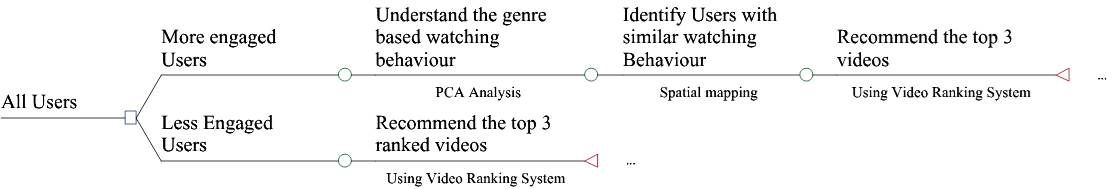
# Rakuten-Viki Global TV Recommender Challenge

**Team name: WhoamI**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Team member(s)** | **Team member(s)’ email(s)** | **Team member(s)’ mobile number(s)** |
| **1** | Wesley Goi | wesley@bic.nus.edu.sg | +65-81002713 |
| **2** | Ritika Kapoor | Ritika.kapoor22@gmail.com | +65-93950226 |

**Introduction and Overview of Solution**

* **Brief description of your algorithm**



All the videos were ranked based on a ranking system (explained later)

* **Detailed description of your algorithm, including a walkthrough of the thought process that guided you to the solution**

**Step1:** **Segmentation of User Base and Ranking of Videos**

1. **Merged user behave and video attributes datasets**
2. **Updated the New Scoring System:** We decided to further segment the scores into 4 categories

The new scoring system was made to be able to give high weightage to users who watched more than 50% of the video as it indicates a higher interest of the users as compared to the whole range of 20% - 80%



**3. Identify the highly engaged User Base based on the initial scoring scheme**

We segmented users into high score users (top 20 percentile) and low score users (rest 80 percentile).

The top 20 percentile watch 60% of the videos and hence were selected as a high priority subgroup.

**Note: Intermediary steps** - We analyzed the behavior pattern of users in terms of the videos they watched compared with the total score they had (Fig1). We observed there was an increase in frequency around the 80th percentile and a very sharp increase after the 95th percentile. As it was also in sync with the famous 20-80 rule, we considered the users in the top 20 percentile as high Score Users who form the important user base for viki.

Analysis was done on High-Score-High-Frequency (hiSchifreq) user groups (top 5 percentile). However, we found that the hiSchifreq group did not have any difference in behavior in terms of the videos watched as the entire highScore user group. We also analysed the user base behavior in terms of the new scoring system but did not find any major difference, hence based it on the initial scoring system only.

**Fig1:** Pattern of user frequency vs. User cumulative score

1. **Developed Ranking Scheme for Videos**

The new ranking basis of videos = f(frequency of visits, score). We gave higher weights to videos with mv\_ratio >=50

**Logic:** This is to overcome the dilemma where a highly clicked video with very low score may rank equal to a moderately viewed video with highScore. We definitely wanted to recommend users which other users had shown interest in and if multiple users did not like the same video, it was not worthy to recommend.

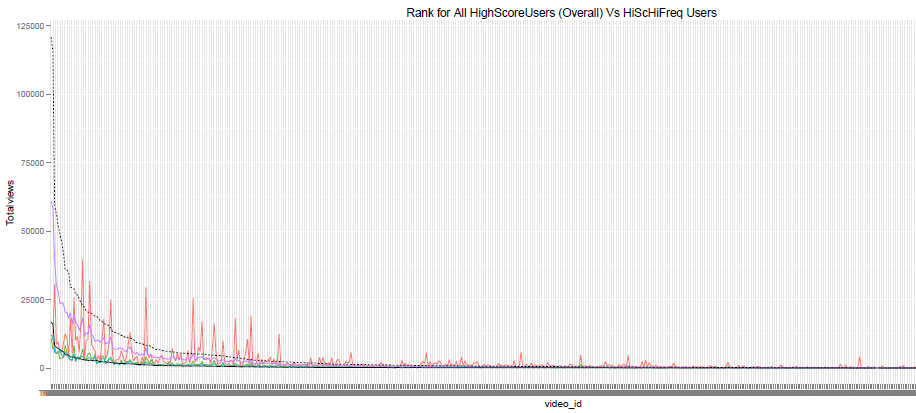
Formulae for Ranking (based on new scoring system):

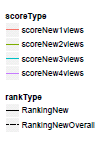
***Rank = (score1views\*1 + score2views\*2 + score3views\*3 + score4views\*4)\* (score3views + score4views)\* total views).***

* + Score1/2/3/4views were taken as a percentage of the total views for that user
  + Multiplication with total views gave a higher ranks to videos which were watched more with a score of 3 & 4.

Intermediary steps:

We compared the performance of the algorithm of ranking the videos among the segment of all high score users to hiSchifreq users. A similar analysis was also done in the hiSchifreq user group on the initial scoring system and the new scoring system (Fig2). Similar analysis was also done comparing (Fig3). These analysis made us conclude that videos ranked among the high score user group and based on the new scoring scheme were a better strategy.



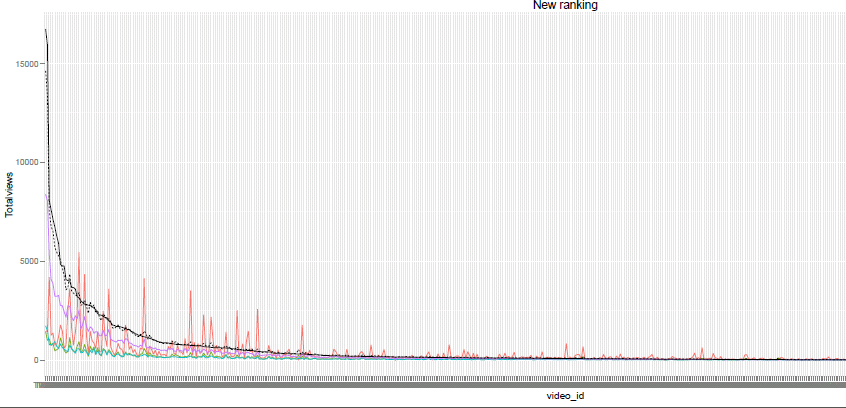


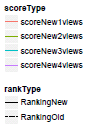
**Fig2:** Comparison of video Ranks for hiScore vs. hiSchifreq User segment.

**Legends:** Ranking New: Ranking among hiSchifreq user segment based on the new scoring scheme

Ranking New Overall: Ranking among high score user segment based on the new scoring scheme

Ranking scale has been readjusted to view the behavior in the same graph.





**Fig3:** Comparison of video Ranks for hiSchifreq user segment based on the two scoring systems

**Legends:** Ranking New: Ranking based on the new scoring scheme (score levels-1,2,3,4)

Ranking New Overall: Ranking based on the old scoring scheme (score levels- 1,2,3)

* + ReadMe of Step2: Wesley’s document (ReadMe for recommendR package)

**Insights from data**

* Can you find the main attributes or factors, which have the most influence on how long a given user watched a particular movie?
* Can you find any other insights from the data?
* The top 5 percentile of the users have a very high frequency with good scores. Hence, viki must try to develop a more precise recommender system for such customers.
* Also, the gender does not successfully classify the watching pattern. Thus drama (specifically Korean drama) is the most watched genres.
* For the low score users, viki can try to recommend different genre videos as they are experimenting with their video types.
* 90% of the hiScorehifreq users of viki are females thus they can actually introduce some features in their website which would appeal more to females.

**Implementation of model/algorithm from the fields selected**

* Can the model/algorithm be implemented? Please explain how.
* How easily can it be implemented?
* Any risk in the implementation? If so, how to overcome it?
* Complexity level of your algorithm?
* Ease of portability and ease of understanding and improvability?
* What was the experimental framework you used to decide between several solutions and error analysis process, if any?
* State assumptions which you made to implement the solution.

**Ideas**

* Ideas to improve current data collection and how it should be done.
* Ideas for external and/or additional data to complement your model.

**Conclusions**

* Explain how this solution will be useful and beneficial to Viki business and its users
* Provide any conclusions you can make from this challenge

# *Algorithm Summary Template*

|  |  |
| --- | --- |
| Algorithm specifications | Particulars |
| Language of your code | R, perl |
| Packages you like to highlight |  |
| Algorithm | Collaborative filtering, content-based, etc |
| Features utilised | User\_id, score, genres, |
| Number of Features |  |
| Feature selection process | How do you select above features from all features? |
| Is the algorithm Parallelizable | Yes/No? |
| CPU | i5 2.6 GHz? i7 2.5 GHz? … |
| RAM | 4G, 8G? … |
| Training time | How long will it take to train your algorithm? |
| Running time | How long will it take to generate the prediction, negligible? |
| Public score | The public score of this particular algorithm. |
| Private score | Will be filled by Dextra team for this particular algorithm you choose. |
| Comments | Your comments on the algorithm |

**Appendix**

* Provide any supporting ideas, suggestions, illustrations, articles and charts.
* Provide a separate .zip file of the complete codes and the codes documentation.