Community-Focused Anime Recommendation System

Personalization for (Public) Media

Lachezar Popov

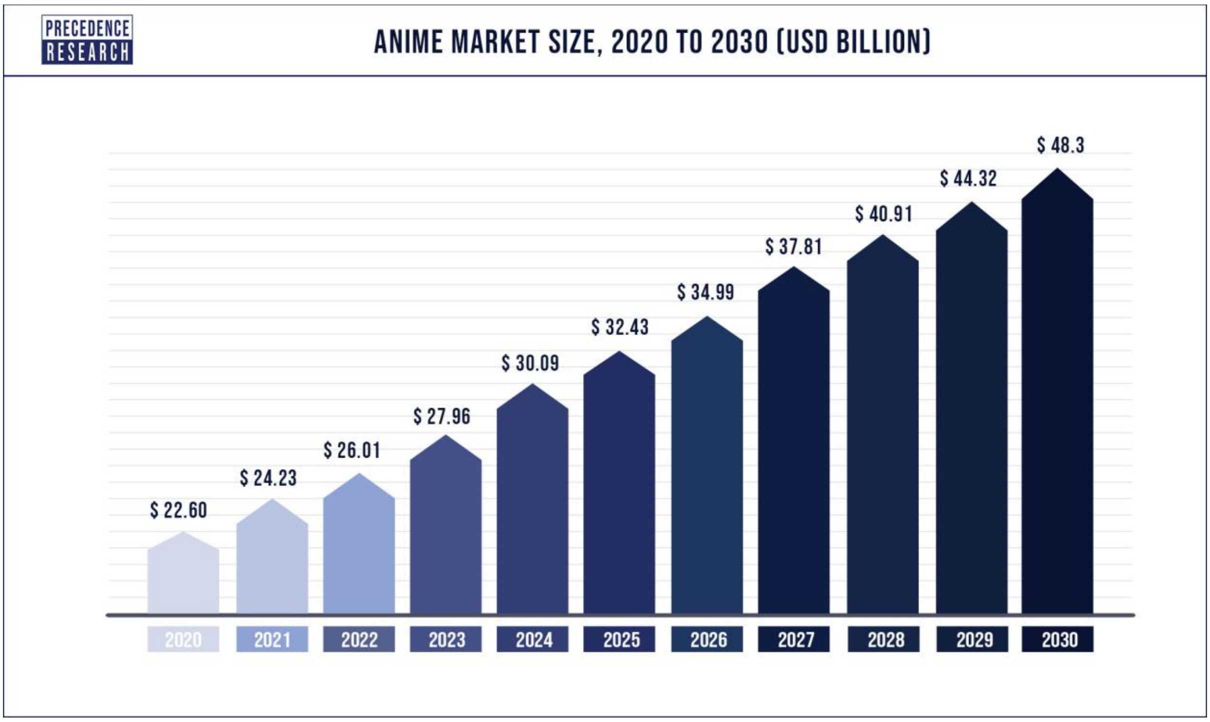
Contents

1. Introduction

1.1. Context and target audience

The chosen medium for my recommendation system is anime. Anime as a medium has benefited from a growing popularity in recent years. Global demand for anime content has grown by 118% in the past two years according to Parrot Analytics (2022). Moreover, the size of the global anime marked is projected to grow from an estimated $22.6 billion in 2020 to $48.3 billion by 2030 at a compound annual growth rate of 7.9% (Precedence Research, 2021).

Figure 1: Projected growth of global anime market (Precedence Research, 2021)



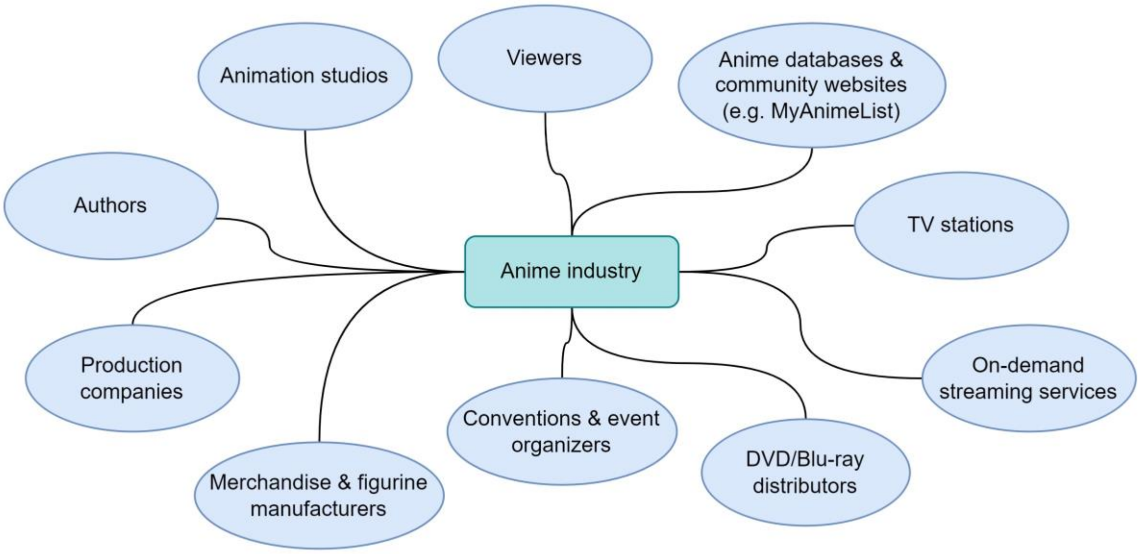
.This all-time-high interest in anime enables the exploration of new kinds of recommendation systems that deviate from the generic format adopted by popular streaming services and emphasize the values of anime fans.

There rest of this report describes such a recommendation system targeted at anime fans and developed through a value-centric approach.

1.2. Stakeholders

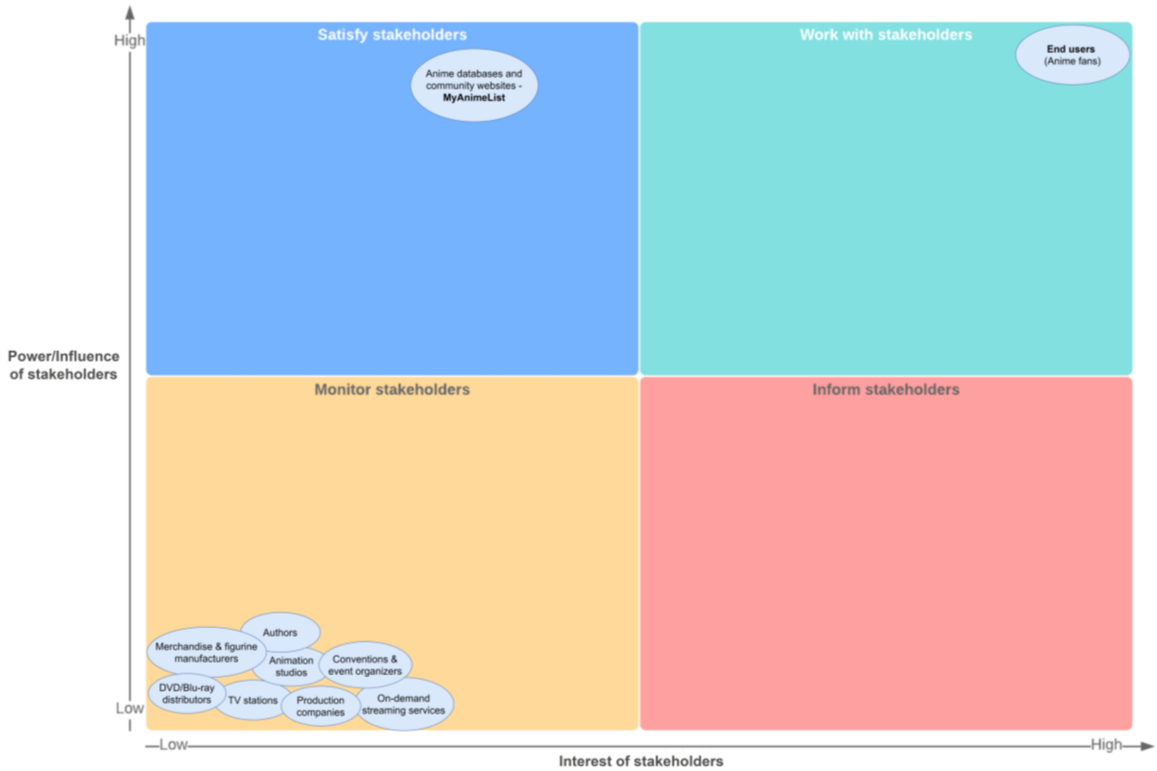
A mind map of the various entities comprising the anime industry is shown in Figure 2.

Figure 2: Anime industry mind map



These entities are then mapped onto a Power-Interest Matrix in relation to their interest in and influence over the recommendation system described in this report (Figure 3).

Figure 3: Stakeholder power-interest matrix



We can see that besides the end users there is 1 other stakeholder of meaningful relevance – MyAnimeList. MyAnimeList (2022) is an anime database and community website from which the data for the recommendation system is retrieved in real-time (see chapter 2.1). This makes them a crucial stakeholder with high influence, as the functioning of the recommendation system is ultimately dependent on access to their API.

The interests of MyAnimeList are straightforward – the terms of use for their API state that usage of the API should not put excessive load on their servers (MyAnimeList, 2019). The value tensions with end users that result from these interests are explored in chapter 4.1.

Although the interests of MyAnimeList are straightforward, the values and interests of end users of the recommendation system require additional research. This research is described in chapters 1.3 and 1.4.

While one might think that streaming services may also be an important stakeholder, they have no meaningful relation to the recommendation system described in this report.

1.3. Literature review

Value-sensitive design is used as the guiding principle for the development of the anime recommendation system. Hence, insights into the values of anime fans are required.

In a paper titled "Psychological Needs Predict Fanship and Fandom in Anime Fans" Ray et. al. (2017) present the results of a survey of 923 anime fans in relation to their motivations to engage with the anime fandom. Table 1 shows summary statistics relating to the importance of these motivations for anime fans.

Table 1: Summary statistics of key motivations of anime fans (Ray et. al., 2017)

Means (Standard Deviation) of Motivations, Fanship, and Fandom by Sex

Variable	Men	Women	F	p	η_p^2
Fanship	4.80 (1.64)	4.69 (1.75)	1.01	.316	.001
Fandom	4.91 (1.54)	4.84 (1.66)	0.41	.523	.000
Self-Esteem	4.08 (1.77)	4.36 (1.80)	5.37	.021	.006
Efficacy	3.93 (1.72)	4.13 (1.84)	2.68	.102	.003
Meaning	3.82 (1.81)	3.95 (1.98)	1.18	.278	.001
Continuity	4.29 (1.84)	4.27 (1.89)	0.03	.854	.000
Belongingness	4.44 (1.72)	4.57 (1.84)	1.20	.273	.001
Distinctiveness	4.20 (1.85)	4.36 (2.03)	1.61	.204	.002
Less Uncertainty	3.41 (1.85)	3.51 (1.96)	0.70	.404	.001
Friends	4.80 (1.71)	4.86 (1.83)	0.31	.575	.000
Social Support	4.09 (1.81)	4.40 (2.03)	6.07	.014	.007
Worldview	4.36 (1.92)	4.55 (1.99)	2.22	.137	.002
Self-Verification	3.64 (1.85)	3.71 (1.94)	0.35	.554	.000

Note. 7-point Likert-type scale, from 1 = *strongly disagree* to 7 = *strongly agree*.

We can see that, not counting Fanship, the top three motivations of anime fans are Fandom, Belongingness and Friends for both men and women. The common denominator between these motivations is the social/community aspect that underpins them. We can think of these motivations in a similar vein as values, and group them under the value of "community". Hence, to delimit the scope of my recommendation system, I will focus on "community" as the value that my recommendation system should represent.

Chih et. al. (2017) explore the notion of community in relation to social influence between members. They argue that a sense of community has an effect on social influence. Hence, we can think of communities in terms of spheres of influence. Moreover, Blanchard & Markus (2002) argue that this sense of community is not unique to physical communities, but also occurs in virtual settings.

1.4. Field research

Three interviews were conducted with anime fans recruited through purposive sampling of the author's group of friends. In these interviews "community" was validated a strongly held value among anime fans and the manifestations of this value were explored.

While there are many ways in which this value manifests, three important ways relevant to the recommendation system are:

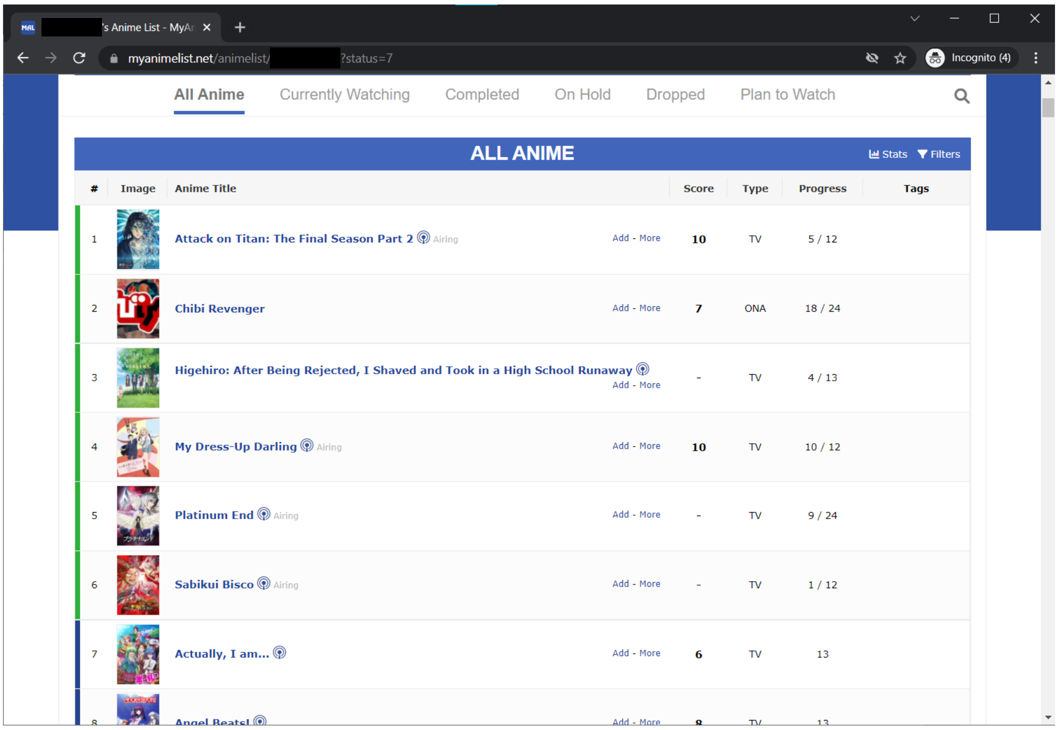
- Anime fans are more likely to watch an anime if it is popular in their community (or friend group)
- Anime fans are more likely to watch an anime if it is highly rated/regarded in their community (or friend group)
- Anime fans are more likely to be influenced by friends with similar tastes

2. Data & methods

2.1. Data

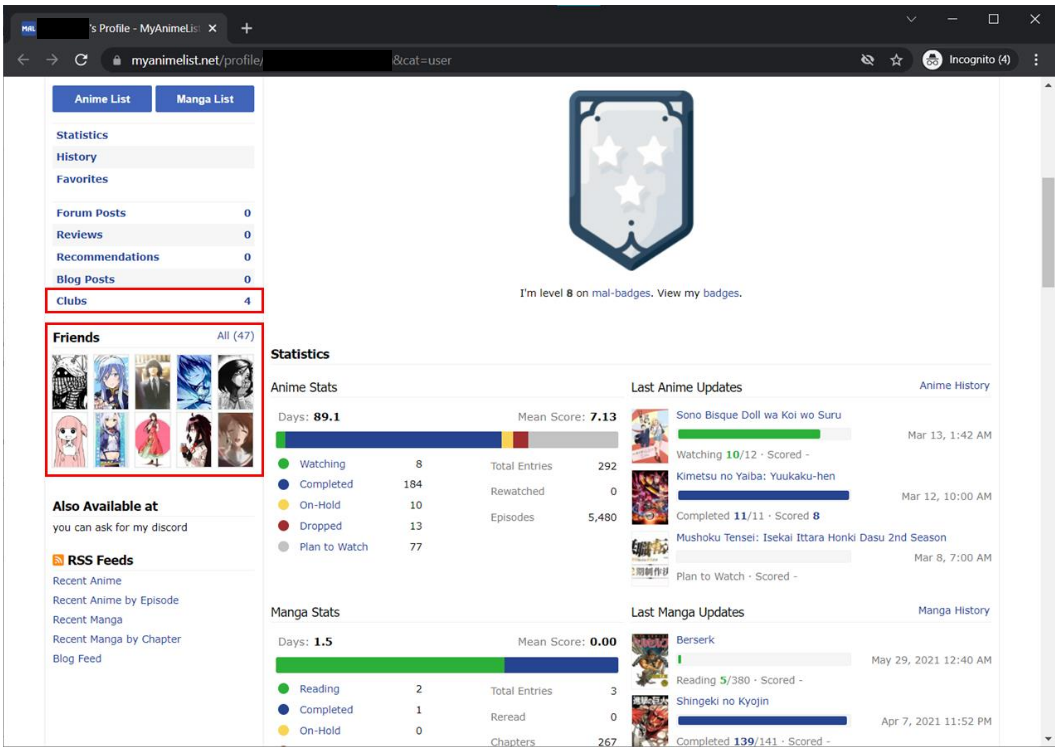
As mentioned in chapter 1.3, all data is retrieved in real-time from MyAnimeList. On MyAnimeList, each user has a list of anime they've watched along with associated ratings (Figure 5).

Figure 4: User anime list on MyAnimeLists



Moreover, users can have friends on the website and be members of various clubs (Figure 6). These can all be regarded as separate communities. I.e., each club can be viewed as its own community and a user's group of friends can also be viewed as a distinct community.

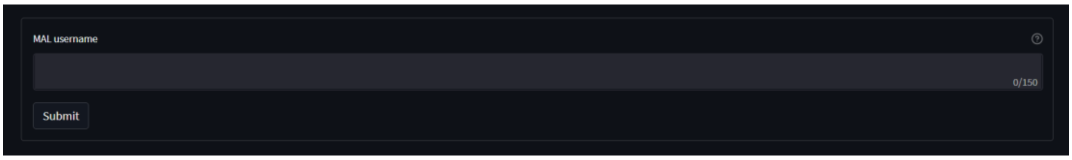
Figure 5: MyAnimeList interface - Friends & Clubs



The vast majority of users on the website have their privacy set to "public", meaning that their data (including anime list, friend list, club list, etc.) can be fetched through the official MyAnimeList API (2021) and the unofficial Jikan MyAnimeList API (2022). My prototype utilizes these two APIs to collect real-time data for generating recommendations.

Data collection is triggered through 2 distinct interactions with the interface:

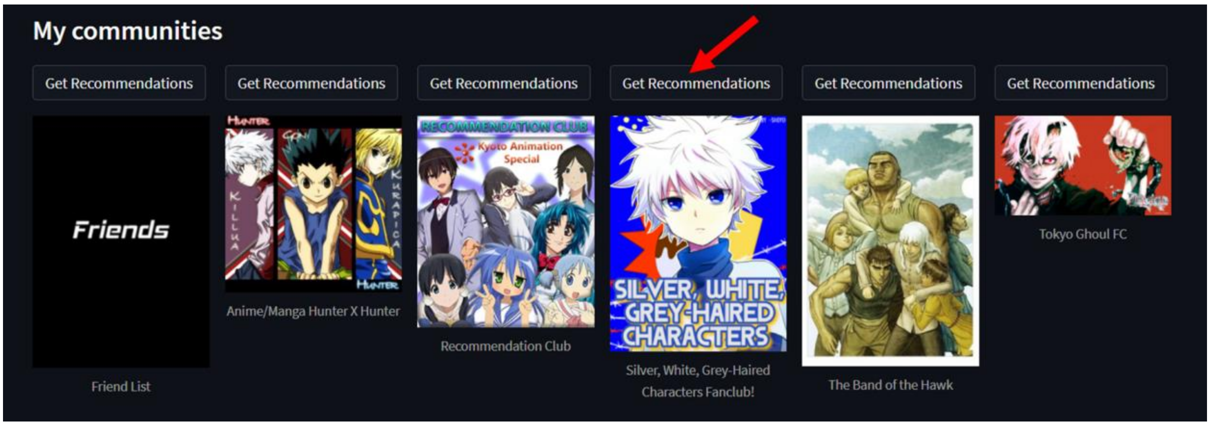
1) Submission of MyAnimeList username:



Upon submission of the user's MAL username, the following data is fetched:

- User profile picture
- User anime list (with ratings)
- For each club that the user is a member of:
  - Club name, image and id

2) Selection of target community



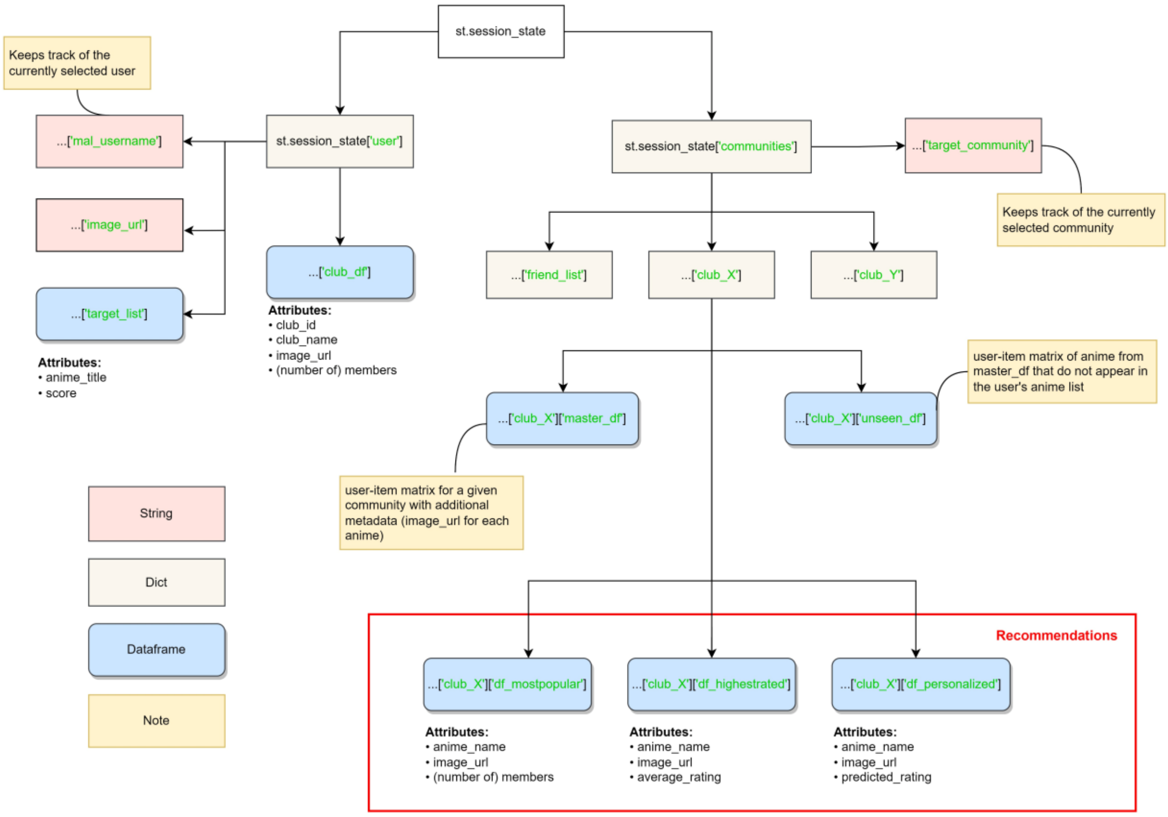
When the user selects a target community by clicking on the "Get Recommendations" button, the following data is fetched:

- List of community members
- Each community member's anime list (with ratings)

Hence the "community" value is directly incorporated into the data used, as the user's own communities (and the anime lists of their members) are used as the context for recommendations.

Upon fetching, the data is processed and recommendations are calculated as described in section 2.2. The data are then stored in the session state as illustrated in Figure 7.

Figure 6: session_state structure



2.2. Methods

Following the insights from field research, I have operationalized the community value of anime fans into the following types of recommendations:

- Most popular in X community
- Highest rated in X community
- Personalized recommendations based on similar users in X community

These three categories of recommendations directly correspond to the manifestations of the community value described in section 1.3.

As all data is collected in real-time, this means that nothing is pre-trained and all recommendations are calculated in real time. I first describe the common steps in generating all three recommendation categories, followed by the steps unique to each category.

Common steps

- A user-item matrix (master_df) is constructed from the anime lists of users in the selected community
- Recommended anime are always ones that do not appear in the user's anime list. To achieve this, I create unseen_df from the master user-item matrix by filtering out the anime that appear in the user's anime list.

- **Most Popular**

Simply counts the number of users from the target community who have watched each anime unseen to the user and return the top 10.

```
def get_most_popular(unseen_df, master_df):
    recs = unseen_df.transpose().count().sort_values(ascending=False).head(10)
    recs.name = 'members'
    recs = master_df[['image_url']].merge(recs, how='right', on='title')
    return recs
```

- **Highest rated**

First, I filter out the anime that appear in less than the square root of the number of users in unseen_df user-item matrix. This avoids recommending anime that have received a 10 by only 1 or 2 users. I use n_members^0.5 as the threshold as I found that it scales nicely with the number of members.

```
def get_highest_rated(unseen_df, master_df, n_members):
    # the anime that appear in more than n_members**0.5 number of user anime lists
    unseen_min_n = unseen_df[unseen_df.transpose().count() >= n_members**0.5]
    recs = unseen_min_n.transpose().mean().sort_values(ascending=False).head(10)
    recs = recs.round(2)  # rounding the average rating
    recs.name = 'rating'
    recs = master_df[['image_url']].merge(recs, how='right', on='title')
    return recs
```

where n_members = len(unseen_df)

- **Personalized recommendations**

This set of recommendations utilizes user-user collaborative filtering. Firstly, the similarity between the user and every community member in user-item matrix is calculated. The similarity measure that is used is the Pearson correlation between two users' sets of ratings for anime that have been rated by both users as described by Falk (2019).

Afterwards the neighborhood of similar users is retrieved, which is defined as community members with a correlation larger than 0.1. While 0.1 might seem like a low threshold, some users will only have 2 or 3 friends and some clubs will only have under 10 members – if a higher threshold is selected, we run the risk of ending up with an empty neighborhood. Moreover, as the ratings are later weighted by users' similarity, less-similar users will have a smaller contribution to the predicted rating regardless.

I then create another user-item matrix from this neighborhood of similar users and anime unseen to the target user, and weight the ratings by the given user's similarity to the target user. Finally, we sum each anime's ratings and divide this sum by the sum of weights (similarities) of users that have rated said anime. This provides us with a similarity-weighted average rating for each anime.

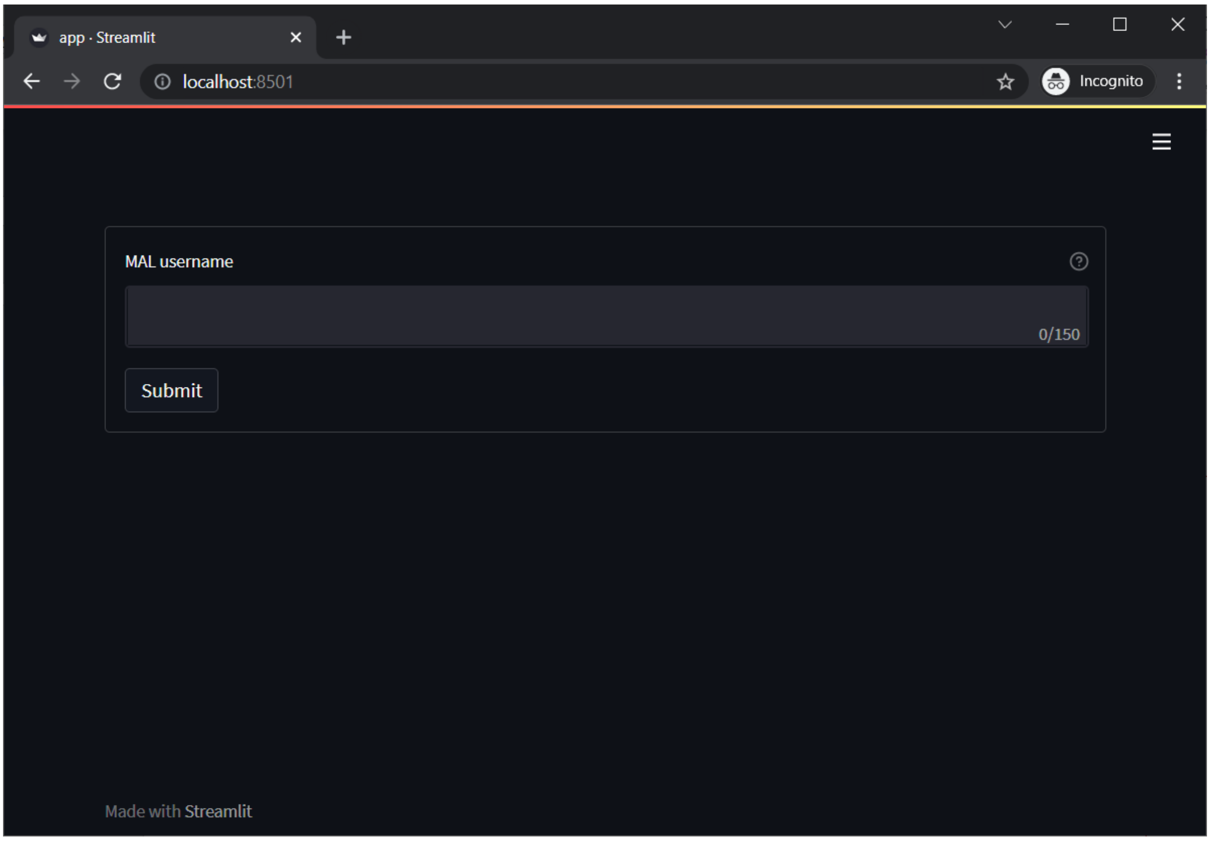
Again, we use n_members^0.5 as the minimum number of neighborhood members that must have rated an anime for it to appear in the recommendations.

```python
def get_personalized_recs(target_list, master_df, mal_username, n_members):

    master_clean = master_df.iloc[:, 1:]

    # Merging target df and master_clean
    sim_df = target_list.merge(master_clean, how='outer', on='title')

    # Calculating similarity between friends/club members and target user
    user_similarities = master_clean.apply(
        target_list[mal_username].corr, axis=0)
    neighbourhood_similarities = user_similarities[user_similarities > 0.10]

    # Filtering master_clean for similar users only
    similar_users = master_clean.loc[:, user_similarities > 0.10]

    # Creating neighbourhood_unseen df
    # the anime unseen by the target user that appear in the neightbourhood of similar users
    neighbourhood_unseen = similar_users[~similar_users.index.isin(
        target_list.index)]

    # Applying weights (based on similarity) to ratings
    def apply_weights(x): return x * neighbourhood_similarities[x.name]
    weighted_scores = neighbourhood_unseen.apply(apply_weights)

    # Filtering out anime that appear in less than n_friends**0.5 number of user lists
    unseen_min_neighbourhood = weighted_scores[weighted_scores.transpose(
    ).count() >= n_members**0.5]

    # Calculating similarity-weighted recommendations
    def weighted_rating(anime): return anime.sum() / \
        neighbourhood_similarities[anime.notna()].sum()
    personalized_recommendations = unseen_min_neighbourhood.apply(
        weighted_rating, axis=1).sort_values(ascending=False).head(10)

    personalized_recommendations = personalized_recommendations.round(
        2)  # rounding the predicted rating
    personalized_recommendations.name = 'predicted_rating'
    personalized_recommendations = master_df[['image_url']].merge(
        personalized_recommendations, how='right', on='title')
    return personalized_recommendations
```

3. Interface design

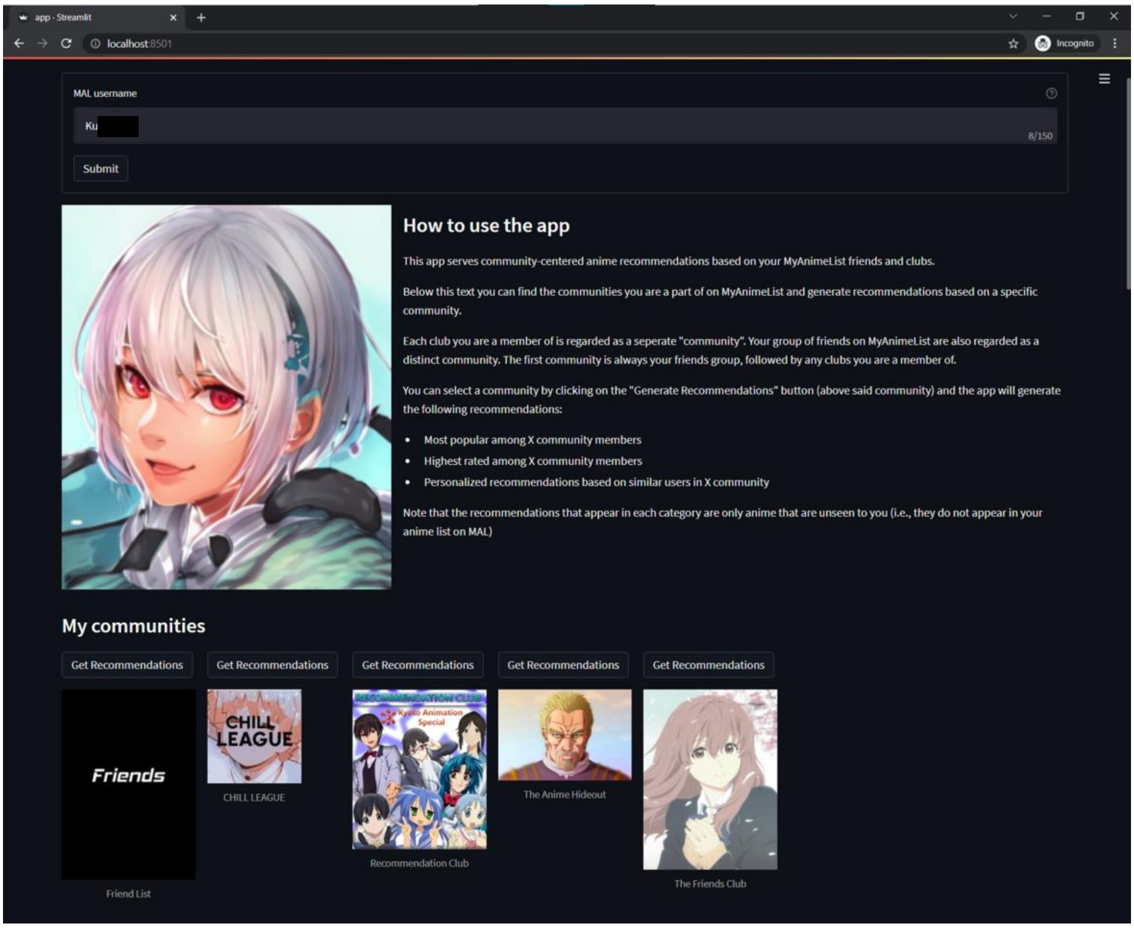
Upon starting the session, the user is met with a clean interface prompting them to enter their MyAnimeList username (Figure 8).

Figure 7: Interface at session start



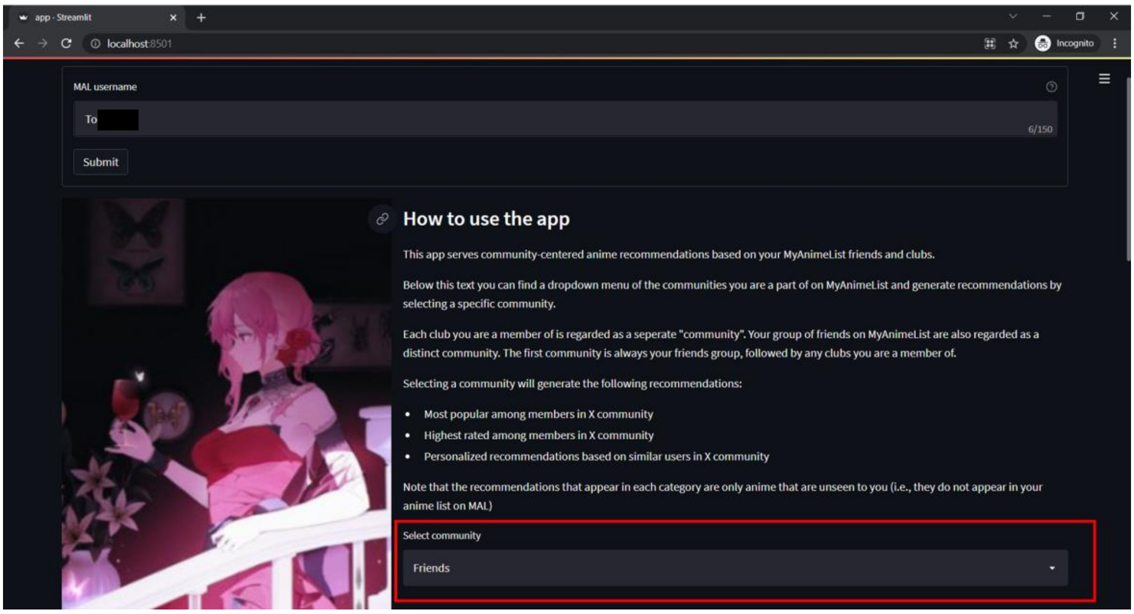
Once the user has entered their MAL username, their profile image is displayed next to a "how-to guide" for the interface (Figure 9). Below this, the user sees the communities they're a member of (i.e., their friend list and clubs). They can select a target community by clicking on the "Get Recommendations" button, after which recommendations are generated as described in section 2.2. The motivation behind this interface design was guided directly by a focus on the community value of the target audience. The design directly incorporates this value by enabling the user to specify the target community and, by extension, the context for recommendations.

Figure 8: Interface - profile image & communities



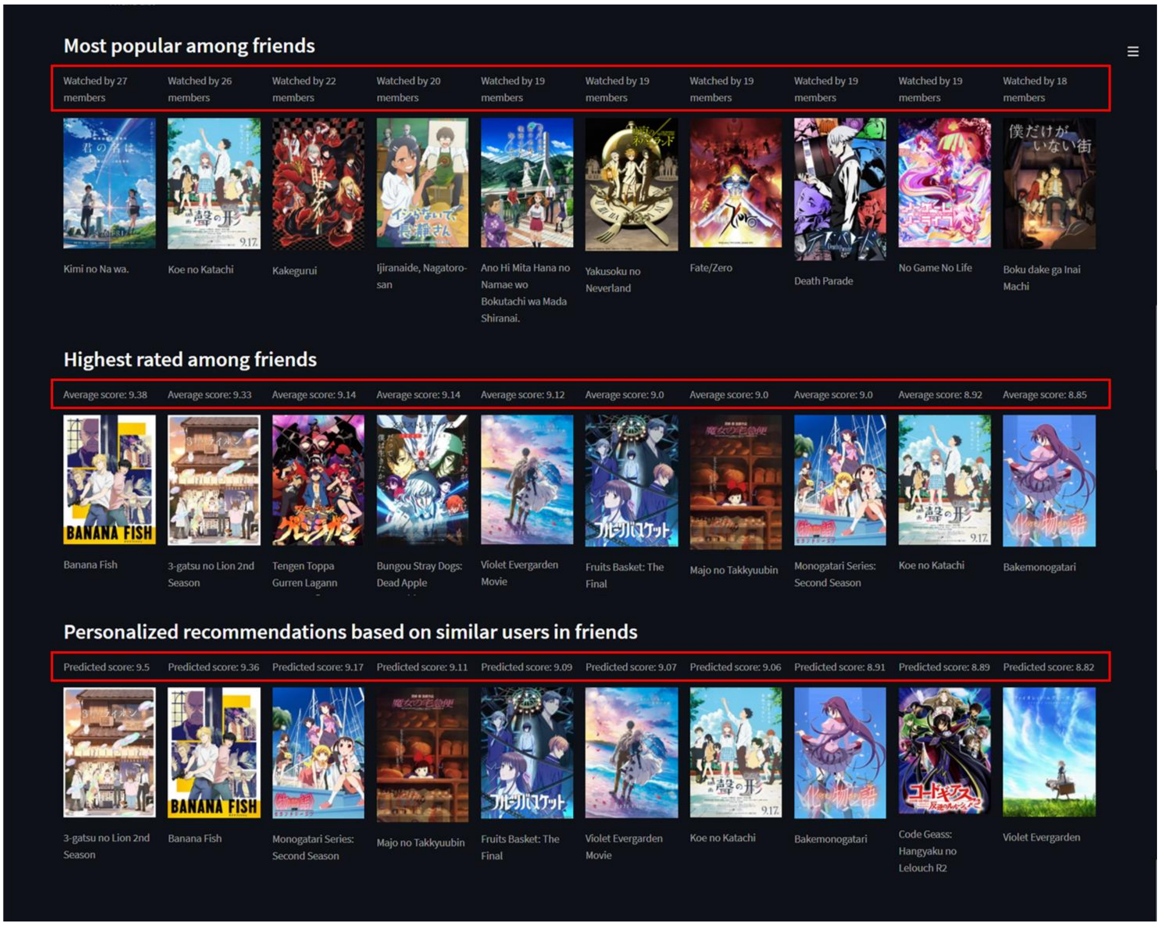
However, if a user is a member of more than 7 communities, a dropdown menu is displayed instead of the community tiles (Figure 10).

Figure 9: Interface - dropdown menu



Some design choices were made as a direct consequence of feedback from user testing. An example of this was displaying the average score, predicted score and “watched by X members” values for the recommendations as shown in Figure 11. These were included as multiple testers said that they would prefer to see concrete numbers behind the recommendations.

Figure 10: Interface - Recommendations



4. Discussion & conclusion

4.1. Value Tensions & Limitations

As mentioned in chapter 1.2, the terms of use for MyAnimeList’s API state that usage of the API should not put excessive burden on the website’s servers. Hence, the number of API calls sent to MyAnimeList’s servers should not exceed a reasonable amount. The same conditions apply to the Jikan API. This creates value tensions between the interests of MyAnimeList and users of the app in two ways:

1) **Quality of recommendations**: More data generally equals betters recommendations. Some clubs on MyAnimeList have thousands of members, however, we cannot send thousands of API calls at once as this would place excessive load on MyAnimeList's servers. This creates a value tension between end users who would like higher quality recommendations based on more data and MyAnimeList. This value tension is balanced by taking a sample of community members. This sample is 50 for a user's friend list and 36 for a given club's members (as there are 36 members per club page and the Jikan API requires specifying a page)

2) **Speed of the interface**: Sending multiple API requests at once creates an increase in server load for MyAnimeList. Moreover, the Jikan API specifies that a maximum of 3 requests can be sent per second. This necessitates a delay between consecutive API calls. This creates a value tension with the end users who would like to have a fast interface. Using a delay of between 0.2 and 0.4 seconds between successive API calls strikes a balance between the interests of these stakeholders.

Hence, the main limitations of the app in terms of speed and scope/quality of recommendations are a direct result of the value tensions between stakeholders. Feedback from user testing has indicated that users have little issue with having to wait an extra 15-30 seconds for recommendations, however, the sampling of community members is somewhat more displeasing.

Another limitation is that users of the recommendation system must have a MyAnimeList account. This limitation may be overcome by including support for other anime-tracking websites (anilist.co, Kitsu, aniDB, etc.)

4.2. Conclusion & future steps

Through implementation of value-sensitive design and an iterative development processed based on user testing, I was able to create a prototype that incorporates the "community" value of anime fans into all three main aspects of the recommendation system – the data, the algorithms and the interface. While there are certain limitations that arise from value tensions between stakeholders, these do not inhibit the core functionality of the app.

Future steps:

- More extensive user testing
- Research into potential integration of other anime-tracking websites (anilist.co, Kitsu, aniDB, etc.)
- Developing suitable performance measurements/metrics. I.e. - can we quantify how well our recommendations implement the "community" value?

Bibliography

Blanchard, A., & Markus, M. (2002). Sense of virtual community - maintaining the experience of belonging. *Proceedings of the 35th Annual Hawaii International Conference on System Sciences.* Big Island: IEEE.

Chih, W.-H., Hsu, L.-C., & Liou, D.-K. (2017). Understanding virtual community members' relationships from individual, group, and social influence perspectives. *Industrial Management & Data Systems*.

Falk, K. (2019). *Practical Recommender Systems.* Shelter Island: Manning Publications.

Jikan. (2022). *Jikan - Unofficial MyAnimeList API*. From https://jikan.moe/

MyAnimeList. (2019). *API License and Developer Agreement*. From https://myanimelist.net/static/apiagreement.html

MyAnimeList. (2021). *MyAnimeList API (beta ver.) (2)*. From https://myanimelist.net/apiconfig/references/api/v2

MyAnimeList. (2022). *About*. From https://myanimelist.net/about.php

Parrot Analytics. (2022). *Anime and Asian series dominate 4th Annual Global TV Demand Awards, highlighting industry and consumer trends towards international content*. From https://www.parrotanalytics.com/announcements/anime-and-asian-series-dominate-4th-annual-global-tv-demand-awards/

Precedence Research. (2021). *Anime Market (By Type: TV, Video, Movie, Internet Distribution, Music, Live Entertainment, and Others) - Global Industry Analysis, Size, Share, Growth, Trends, Regional Outlook, and Forecast 2021 - 2030.* Ottawa.

Ray, A., Plante, C. N., Reysen, S., Roberts, S. E., & Gerbasi, K. C. (2017). Psychological Needs Predict Fanship and . *The Phoenix Papers*.