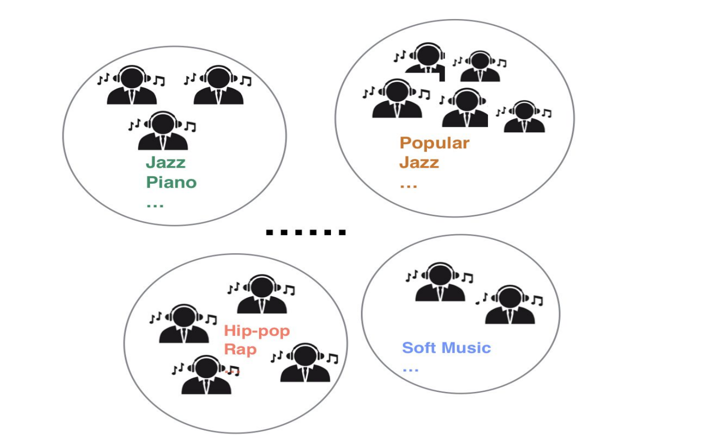
Music Recommendation and Churn Prediction

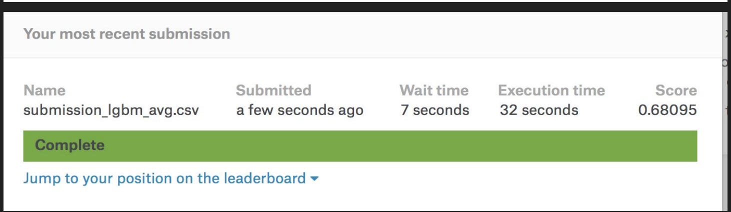
-XJTU squad
Yiming Sun ys3031
Yinan Wang yw2924
Yiwei Chen yc3343

Our Achievements

Music Recommendation

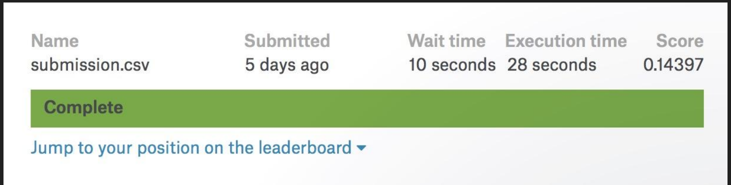
- User Clustering
  - divided huge amount of of users into different clusters
  - dataset is around 30 GB
- Music Preference Prediction
  - Implemented on Lightgbm
  - Precision is higher than 70% competitors on kaggle





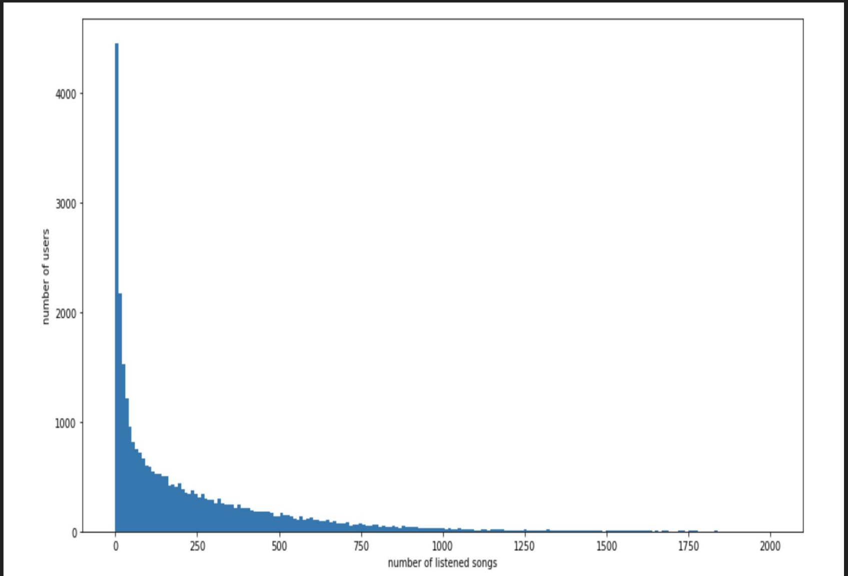
Churn Prediction

- Implemented on xgboost
- Precision is higher than 72% competitors on kaggle

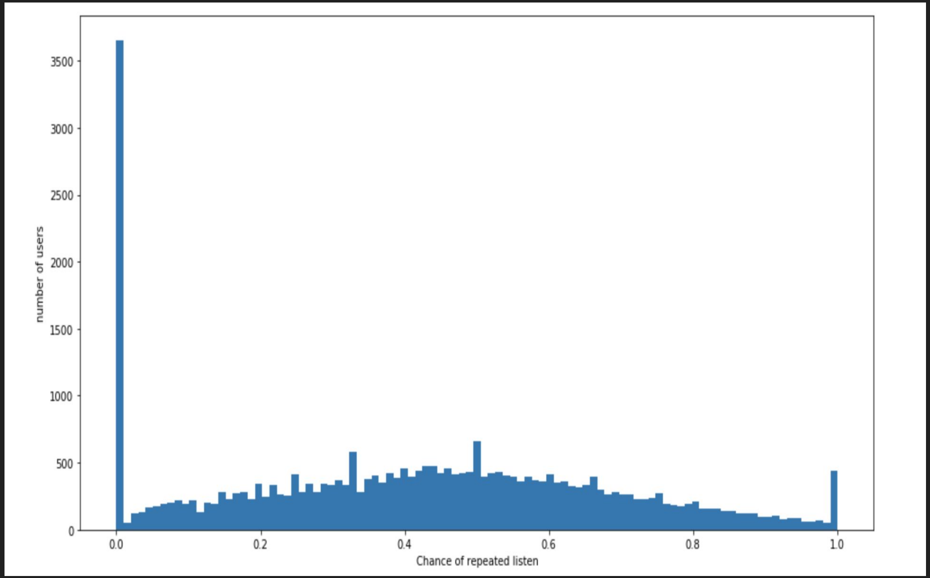


Statistics on our datasets

Number of listened songs / Number of users

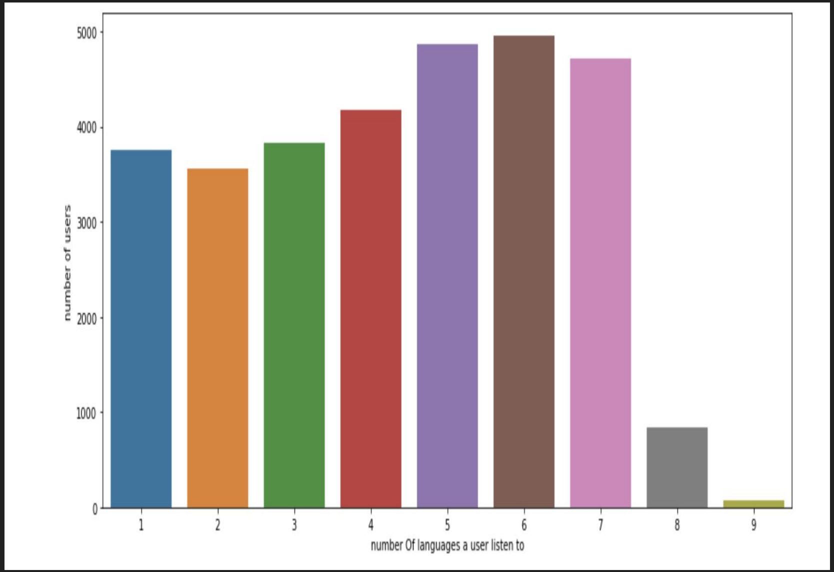


Chance of repeated listen / Number of users

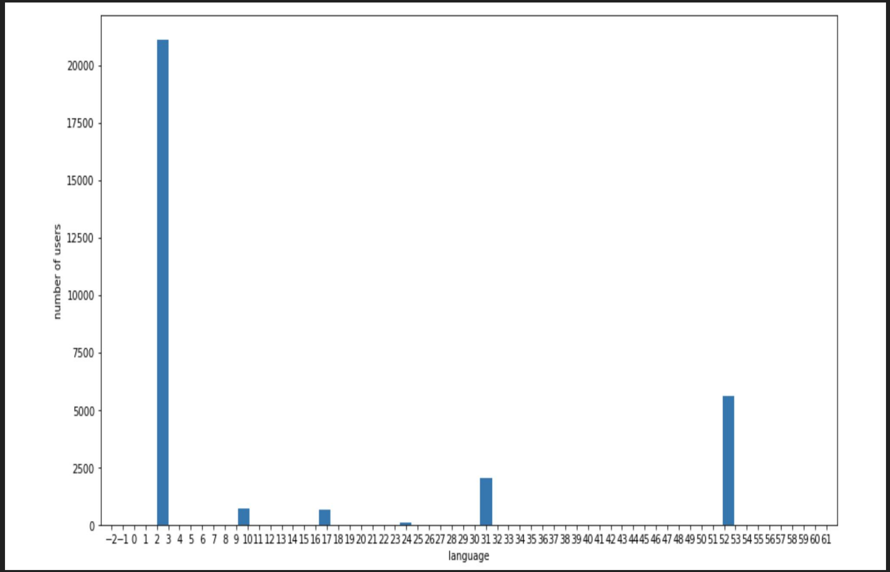


Statistics on our datasets

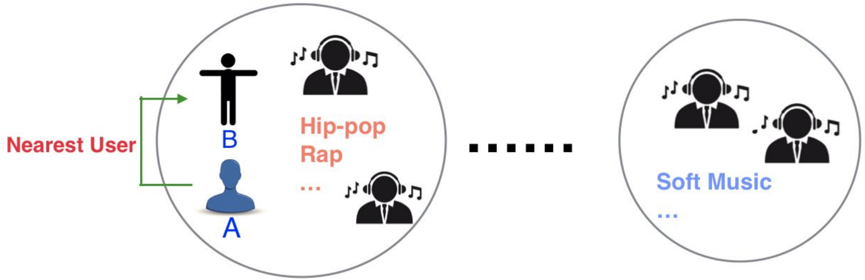
Number of languages / Number of users



Language / Number of users



Music Recommendation



Recommend music for user A:

1. Find the nearest user who share the most similar taste: User B

   (K-Means)

2. Get user B's listening record : song1, song2, song3…

3. For each of those songs, predict the probability that the user A would like it

   (Prediction Model: Dropouts meet Multiple Additive Regression Trees, Gradient Boosting Decision Tree)

4. Pick 10 songs with the highest probability.

Acquired song record fragment:

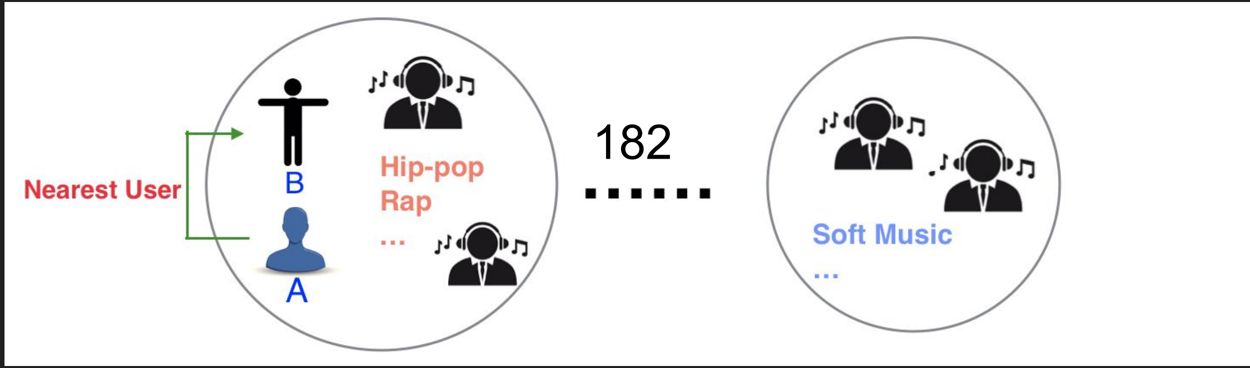
user_id	song_id
HouRZ5St5Wid8UQi64CuMZTClZvTGrMdHSnHzDR/bwY=	59Xo1+K0GkZFiVeqf9sKsOrWzjVOrklEieYUwk7TDlc=
HouRZ5St5Wid8UQi64CuMZTClZvTGrMdHSnHzDR/bwY=	d3UXx7h2qbFwxUzK1bH2hfvPHebcuQ+kAo86YyFlfl0=
HouRZ5St5Wid8UQi64CuMZTClZvTGrMdHSnHzDR/bwY=	t4Vlf/YT6j2Uo5XfhKCrBEsYjlya5xW31NKgDSDhErU=
HouRZ5St5Wid8UQi64CuMZTClZvTGrMdHSnHzDR/bwY=	sakda7kZsGCFMA9S+J+bl8BmqrU3ROZl+Yz2o+ITcQU=
HouRZ5St5Wid8UQi64CuMZTClZvTGrMdHSnHzDR/bwY=	LJrt6EKMcLkOgfyNT17WEg1i/XoNklJeGgADJSEB30M=

10 songs with top probability

YN4T/yvvXtYrBVN8KTnieiQohHL3T9fnzUkbLWcgLro=	讓我留在你身邊	0.85719
DLBDZhOoW7zd7GBV99bi92ZXYUS26lzV+jJKbHshP5c=	演員	0.81359
icz1X14EpEuV+j2SsoUEODyk3XOM9KEs5YumyEGhBko=	醜八怪	0.79963
jHMqK5wu5C0txl0QCxpkSM1xbIev60ii+dc99LXu8EI=	嗯哼 (Uh-Huh)	0.79132
PgRtmmESVNtWjoZHO5a1r21vIz9sVZmcJJpFCbRa1LI=	謝謝妳愛我 (Thanks For Your Love)	0.77507
wBTWuHbjdjxnG1lQcbqnK4FddV24rUhuyrYLd9c/hmk=	小幸運 (A little happiness)	0.76364
QZBm8SOwnEjNfCpgsKBBGPMGET6y6XaQgnJiirspW7I=	年輪説	0.73799
cy10N2j2sdY/X4BDUcMu2Iumfz7pV3tqE5iEaup2yGI=	派對動物 (Party Animal)	0.70148
750RprmFfLV0bymtDH88g24pLZGVi5VpBAI300P6UOA=	FLY OUT	0.70082
W+SDJG+ZtQvSYeAJyIcTxlrpYGRJu791VgVucTlPqM8=	異類 (ALIENS)	0.69451

Music Recommendation
- User Clustering

- Count different user's listening event on music belonging to different genres
- Make 182 clusters based on the similarity of music tastes among users



- For a target user A, we can recommend his/her nearest neighbor user B as his/her potential friends. On the one hand, we can achieve friend recommendation, and after that, we can make music recommendation for A based on the listening history of B.
- Model selection: K-means clustering model based on EM algorithm.

Music Recommendation
- User Clustering (example)

- Input: Users' listening event on different music genres
- Output: 182 user clusters
- Usage: Recommend a friend user B who is A's nearest neighbor to target user A, and output B's listening records

User A

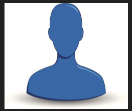
user_id: 2ByH1Vd7BiB8nYGClG0juSnBzZ3sJ4W4a2WjedWSOWM=

User B

user_id	song_id
HouRZ5St5Wid8UQi64CuMZTCIZvTGrMdHSnHzDR/bwY=	59Xo1+K0GkZFiVeqf9sKsOrWzjVOrklEieYUwk7TDlc=
HouRZ5St5Wid8UQi64CuMZTCIZvTGrMdHSnHzDR/bwY=	d3UXx7h2qbFwxUzK1bH2hfvPHebcuQ+kAo86YyFlfl0=
HouRZ5St5Wid8UQi64CuMZTCIZvTGrMdHSnHzDR/bwY=	t4Vlf/YT6j2Uo5XfhKCrBEsYjlya5xW31NKgDSDhErU=
HouRZ5St5Wid8UQi64CuMZTCIZvTGrMdHSnHzDR/bwY=	sakda7kZsGCFMA9S+J+bl8BmqrU3ROZl+Yz2o+lTcQU=
HouRZ5St5Wid8UQi64CuMZTCIZvTGrMdHSnHzDR/bwY=	LJrt6EKMcLkOgfyNT17WEg1i/XoNklJeGgADJSEB30M=

Music Recommendation
- Preference Prediction

- We have the listening records of A’s nearest neighbor, we want to know whether A will like these songs or not.
- Predict the preference of A based on GBDT and DART models.
- GBDT and DART models are trained based on the listening records of all users and the output of this model is the probability of a certain user whether will listen to a song again within one month, which could reflect the preference of a user to a song.



music1: 0.99
(A will like it)

music2: 0.7
(A would probably like)

music3: 0.2
(A would not like it)

- Evaluation: accuracy is measured by the fact whether the user listened to a song or not

Music Recommendation
- Preference Prediction (example)

- Input: test data set which contains the listening habits of user A and listening history of user B
- Output: the preference of A to each song in B’s listening records, which is quantified by the probability that whether A will listen to the same song again within a month.
- We can recommend the songs with highest probability to user A

174	YN4T/yvvXtYrBVN8KTnieiQohHL3T9fnzUkbLWcgLro=	讓我留在你身邊	0.85719
29	DLBDZhOoW7zd7GBV99bi92ZXYUS26lzV+jJKbHshP5c=	演員	0.81359
34	icz1X14EpEuV+j2SsoUEODyk3XOM9KEs5YumyEGhBko=	醜八怪	0.79963
18	jHMqK5wu5C0txl0QCxpkSM1xbIev60ii+dc99LXu8EI=	嗯哼 (Uh-Huh)	0.79132
166	PgRtmmESVNtWjoZHO5a1r21vIz9sVZmcJJpFCbRa1LI=	謝謝妳愛我 (Thanks For Your Love)	0.77507
70	wBTWuHbjdjxnG1lQcbqnK4FddV24rUhuyrYLd9c/hmk=	小幸運 (A little happiness)	0.76364
28	QZBm8SOwnEjNfCpgsKBBGPMGET6y6XaQgnJiirspW7I=	年輪說	0.73799
5	cy10N2j2sdY/X4BDUcMu2Iumfz7pV3tqE5iEaup2yGI=	派對動物 (Party Animal)	0.70148
114	750RprmFfLV0bymtDH88g24pLZGVi5VpBAI300P6UOA=	FLY OUT	0.70082
30	W+SDJG+ZtQvSYeAJyIcTxlrpYGRJu791VgVucTlPqM8=	異類 (ALIENS)	0.69451

Churn Prediction

Goal: predict whether a user will pay for his/her membership next month

Training data:

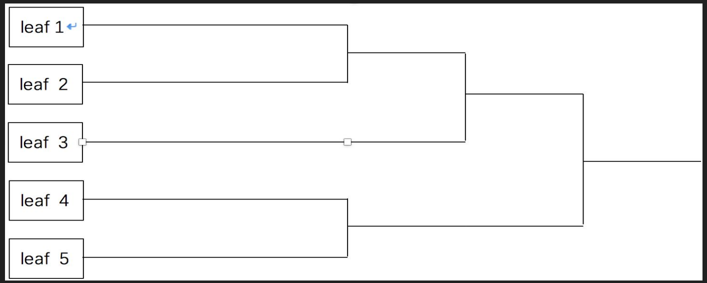
- features: user’s personal information, user’s payment history, user’s listening history
- label: “0” represents the user does not pay for his/her membership while “1” represents yes.

Model & Algorithm: Xgboost

Result & Application:

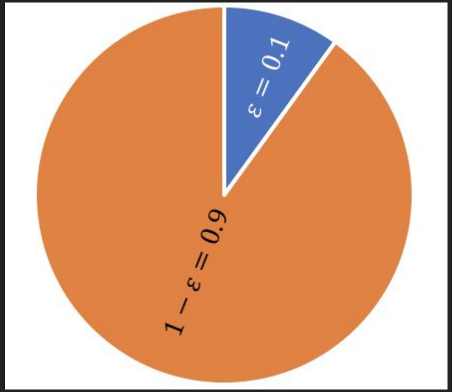
msno	is_churn
0Xc9eSMI3ECDQn1hAqzrJ3QHPpCj7lR+tqZNAmmqWM0=	0.025032
c9cFu574DMJXVgqC4aEK3RKV7vgR8vmafEnQrfv91WQ=	0.030573
TxgLd88Ophk+l6HWogjY7/MRnUtl8eVwV7IdOcxvBMk=	0.030573
fjAo2Ja5hSUNg0WDmgUOrj0RGq+Xo+6GnI8TZbgZpzw=	0.107579
HXBKLi/aPa802h0hXpc23vl33TU83eTlMoG727z/3Mw=	0.875791

Future Work



Hierarchically Clustering

- dealing with even larger user data - TBs of data
- makes millions of clusters
- improvement of efficiency
- when a target user is fixed, we can easily locate the corresponding cluster and recommend similar users to him/her



epsilon-greedy recommendation

- for possibility 1-epsilon, recommend the best music
- for possibility epsilon, recommend new music randomly

Thank You

Q&A