

Overview of Anime voice actor's social network and popularity.

Florenzia Zanollo

Abstract—Although Social Network of actors is a relatively common object of investigation that has been addressed many times; we can say that seiyuu (anime voice actors), come from a very different industry with a distinct way to relate to each other.

In this research we use Wikidata and MyAnimeList to collect seiyuu information and build a Social Network. Topics explored:

- Structure and characteristics of seiyuu Social Network.
- Understanding what properties have a main role describing and predicting popularity of seiyuu.
- Compare prediction performances between different machine learning algorithms (and different models).

I. INTRODUCTION

TODO.

II. ANIME/SEIYUU DATASET

WIKIDATA¹ is a collaboratively edited knowledge base intended to provide a common source of data which can be used by Wikimedia projects such as Wikipedia. The information is stored in RDF format, and can be retrieved in multiple ways, one of them being through a SPARQL endpoint.

Using Wikidata's SPARQL endpoint we retrieved a list of seiyuu. This list contains all persons that have seiyuu as occupation, a total of 6472 entities were obtained². Gender, birthday and birthplace information was also fetched (last two were not used in the end because it was lacking in the majority of entities).

Since Wikidata information about seiyuu's works is really incomplete, MyAnimeList (MAL)³ was used to retrieve voice acting roles and anime information. MAL is a social networking and social cataloging application website with a large database on anime and manga that started in April 6, 2006. Users can make a list of currently watching, watched and/or favorite anime; also score, review, comment and recommend similar ones. They can also put comments and favorite people working on the industry (voice actors, directors, editors, etc).

Since only 59 of Wikidata's seiyuu entities had MyAnimeList ID (MALID) property; a matching between Wikidata and MyAnimeList was done using seiyuu's complete name to retrieve the ID for those who didnt had. Successfully restoring 3033 MALIDs, giving a total of 3092 seiyuus with that property; 2956 of them have at least one work according to MAL so we are using this subset for our experiments.

Advisor: Hideaki Takeda

¹<http://wikidata.org/>

²There's actually 7030 seiyuu in Wikidata but only 6472 of them have an English label (name)

³<https://myanimelist.net/>

Using Jikan API⁴ and MALID, seiyuu data, voice acting roles and more information about each anime was retrieved.

An issue to take into account is whether we unify all anime adaptations of the same intellectual property as one or take a single adaptation as a independent work. We choose the later because a seiyuu could work at one adaptation only, which has its own producer, score, popularity, among other information; it would be incorrect to say a seiyuu worked in a popular work when actually that adaptation didnt have enough fame.

Information that was ultimately used:

- For Seiyuu:
 - Name
 - Debut (this was obtained from oldest work aired date)
 - Gender
 - Popularity (member_favorites information of MAL)
 - Work (anime roles)
- For Works (Anime):
 - Year that began airing
 - Favorites
 - Score (from 0 to 10, MAL user based)
 - Popularity (ranking over all MAL animes)
 - Genres

It's important to notice that data such as popularity and scores are retrieved from MAL, which is user review based only, so it may differ with actual awards winning or professional reviewing of works.

Also, this dataset is biased in favor of more recent anime and seiyuu, since it accounts for more complete data and with better quality. Oldest anime in this dataset is from 1960 having no record about previous ones. Majority of seiyuu's debut are from 1988 which leads us to think information from thereon is more complete.

The data was stored using Virtuoso server to create a local SPARQL endpoint, mongodb was also used as an intermediate storage (before formatting data as RDF).

III. SEIYUU SOCIAL NETWORK

Social networks are a graph like structure with a set of nodes, usually actors or organizations; and a set of edges, defined as a type of relation between actors. This structures are useful to analyze complex social interactions and communities.

A. Node and edge definition

Our social network consists of voice actors (seiyuu) as nodes and co-workship between them as edges. It's important to notice that this social network is time dependant since

⁴<https://jikan.docs.apiary.io/#>

each seiyuu has a debut year and each anime has an aired time; giving us freedom to choose different time frames to observe it.

Aside from being time dependant there exists different possible definitions of relationship or co-workship between seiyuu. One could say two actors know each other if they have worked in at least one job together, or maybe it requires more than one. There's also a time frame to define, relationship could take into account all works of both of them or only of a certain time frame.

After observing graphs built with different interpretations of relationship, the criteria for connecting two nodes became: at least 10 works in common, during the time frame between the first debut registered (1960) and the year of observation. The reason behind this decision is that requiring more jobs in common means less amount of edges; this leaves a more understandable graph without changing its structure.

There's also other interesting definitions of relationship, for example we can use only common works from the last x years. This options weren't explored; having into account our limited time we opted to decide on one and put more effort in analyzing the data and social network at hand.

B. Construction

As a first approach Gephi was used to build the network. Since the graph was big enough to bring performance problems and we needed to build the edges dynamically (which couldn't be done in Gephi) NetworkX was used instead.

NetworkX was chosen because it's an easy yet powerful Python library, it doesn't get along with massive graphs but ours was not big enough to present a problem. One can also export the graph and open it on Gephi, for a more visual analysis.

We needed to build the edges dynamically because they depend on the time frame we are looking at. For example if two actors worked together in 9 jobs between 1960 and 1970 we shouldn't see an edge between them; but if they worked together again in 1971 then looking at 1960-1971 they should be connected.

C. Analysis

Is easy to tell at first glance that this social network is really interconnected. With only 2956 nodes it has 395887 edges when only one work in common is required and 13629 edges when asking for 10 or more. It shows a tightly interconnected cluster surrounded by poorly or not connected nodes. This cluster represents 99% of the nodes of one work in common graph and 23% of 10 works in common.

In terms of modularity we can see at least four clear communities in each graph.

Table I shows some metrics about each graph. Requiring more works in common decreases average degree circumstantially but doesn't change a lot modularity or network diameter.

Tables II and III show top 10 nodes, for degree and betweenness centrality. Only for 10 works in common graph since, aside from actual values, they are similar.

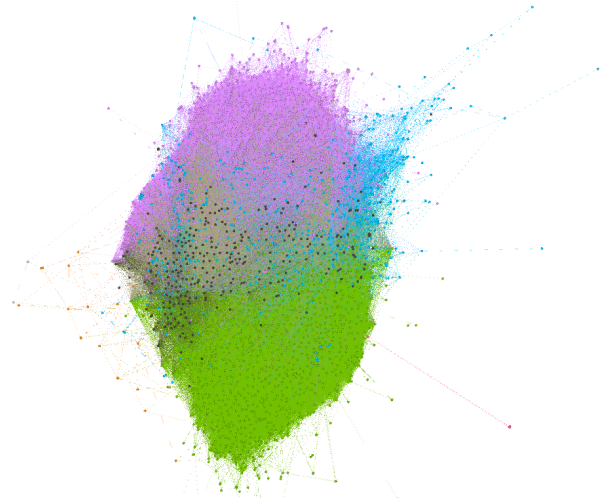


Fig. 1. At least one work in common graph coloured by community.

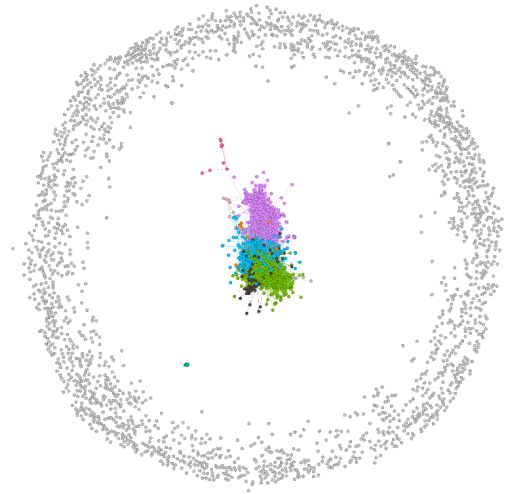


Fig. 2. At least ten works in common graph coloured by community. Big cluster at the center, surrounded by loosely connected nodes.

TABLE I
GRAPH ANALYSIS

Graph	Avg Degree	Graph Density	Modularity
One work in common	267	0.09	0.2
Ten works in common	9	0.003	0.29

Graph	Network Diameter	Connected Components
One work in common	6	18
Ten works in common	7	2261

TABLE II
TOP 10 DEGREE

Name	Degree
Takehito Koyasu	311
Akira Ishida	273
Mamiko Noto	258
Daisuke Namikawa	232
Katsuyuki Konishi	229
Keiji Fujiwara	220
Junichi Suwabe	216
Toshiyuki Morikawa	215
Rie Kugimiya	213
Nobuyuki Hiyama	201

TABLE III
TOP 10 BETWEENNESS CENTRALITY

Name	Betweenness Centrality
Takehito Koyasu	18489.440724
Mamiko Noto	10988.962717
Daisuke Namikawa	9570.480619
Akira Ishida	8299.187308
Rie Kugimiya	7560.155871
Katsuyuki Konishi	7413.715782
Kenichi Ogata	7160.536666
Harumi Sakurai	6775.762776
Keiji Fujiwara	5980.58333
Yoshimasa Hosoya	5607.954551

TODO

- GRAPHIC AND EXPLANATION OF HOW NODES AND EDGES GROW, THE FACT THAT'S SIMILAR FOR BOTH GRAPHS

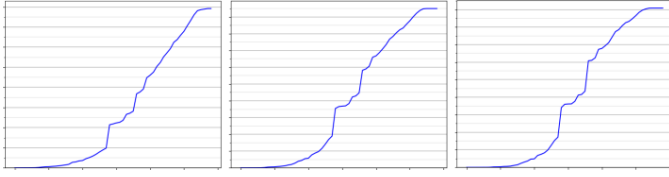


Fig. 3. Grow of edges over time. For 1, 5 and 10 works in common graphs

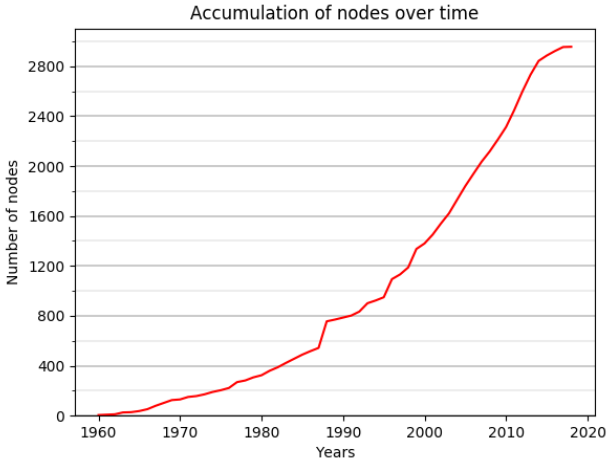


Fig. 4. Grow of nodes over time.

IV. ANALYSIS AND PREDICTION OF SEIYUU POPULARITY

POPULARITY is an abstract criterion that must be defined as a numerical metric in order to be used for analysis and prediction. Since we are using MAL database for seiyuu and anime information and it has a social component; seems logic to use member_favorites as a representation of popularity. We can also get popularity and score for works from opinions of the same set of users.

In terms of distribution *popularity* is highly unequal –as we can observe in Fig. 5– having a lot of seiyuu which are no member favourites and only a few who are favorite of more than 10000 members. It's good to keep in mind that users can favorite multiple seiyuu.

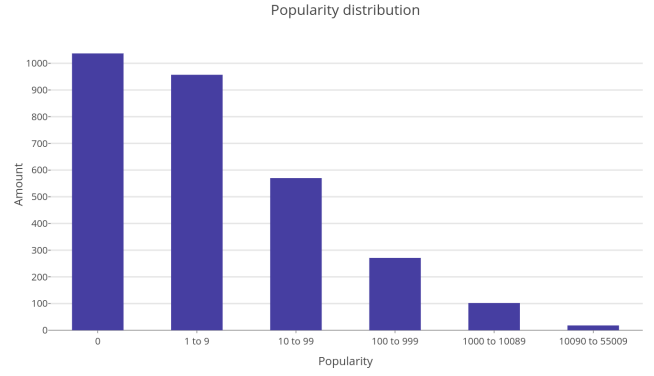


Fig. 5. Amount of seiyuu with that popularity, divided into groups for better visualization.

This is something to take into account when trying to predict popularity of actors or explain it using other features. TODO
ADD WHY

- Mean: 289.55
- Median: 2.0
- Max: 55018
- Min: 0 (1037 values equal to zero)
- Only 120 values bigger than 1000

A. Correlation with only one feature

As shown in Fig. 5 our first approach to explaining popularity was using Pearson correlation.

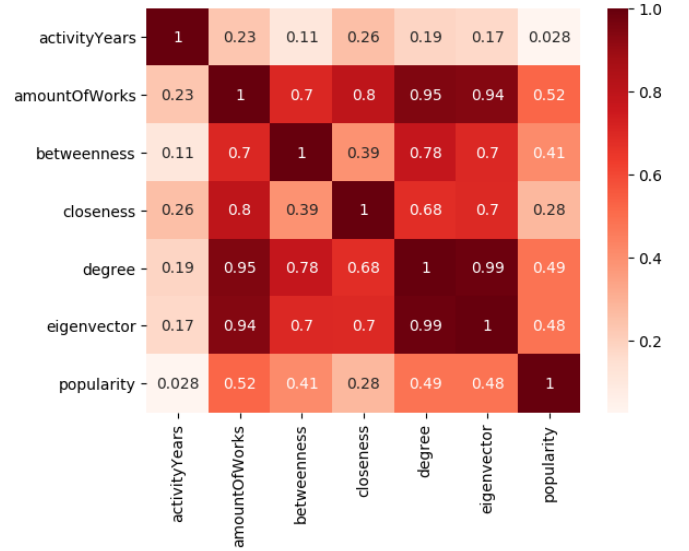


Fig. 6. Pearson correlation between some features of nodes.

A fairly big correlation can be seen between popularity and amount of works. Since this data is biased to more modern anime we thought of trying to correlate with more recent works only. But, how recent? Last 5, 10 or 20 years? Thus correlation between popularity and works from different data frames was analyzed, Fig. 7.

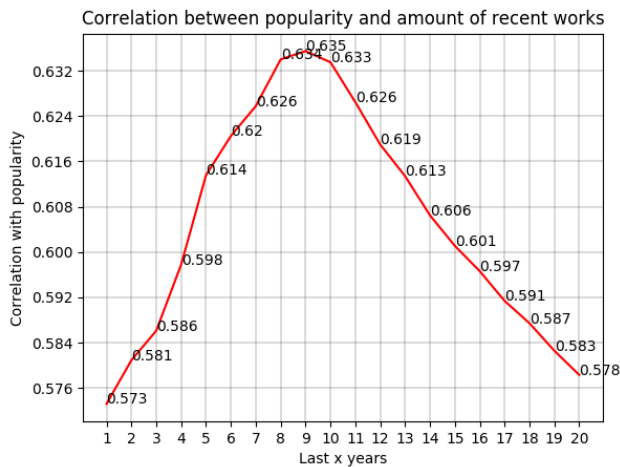


Fig. 7. Last X years means works from 2018-X to present.

The best result was given by recent works from last 9 years. Therefore, this definition of recent works was used from there on.

Graphics of some characteristics of works divided by years were made, trying to shed some light over why works from last 9 years were more "important". Fig. 8 shows an improvement in average of scores and favorites, with the biggest peak in 2005. Popularity goes down but we need to consider that this metrics are from MAL and it could mean, for example, the parameter is in disuse, instead of representing how people feel about new anime.

As we can see on Fig. 9 there's more anime recently, industry is growing bigger each year, in an exponential way; not only that but MAL should have info of every adaptation of last year but maybe not for anime from 1980.

The majority of works are from 1990 to 2018 and half of them are distributed over the last 14 years (2014 to 2018) but as far as we can tell there isn't anything particular over the last 9 years nor on year 2009. Judging by amount of favorites per year it appears that users have been more active in recent years so this could be one of the reasons.

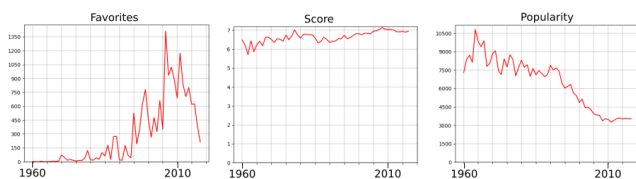


Fig. 8. Average favorites, score and popularity per year.

Some interesting enough correlations are shown next TODO
ADD SCATTER PLOTS AND EXPLAIN A LITTLE MORE

B. Correlation with multiple features

For this section Scikit-learn, a free software machine learning Python library, was used. The node attributes were divided into categories, leaving four distinct types:

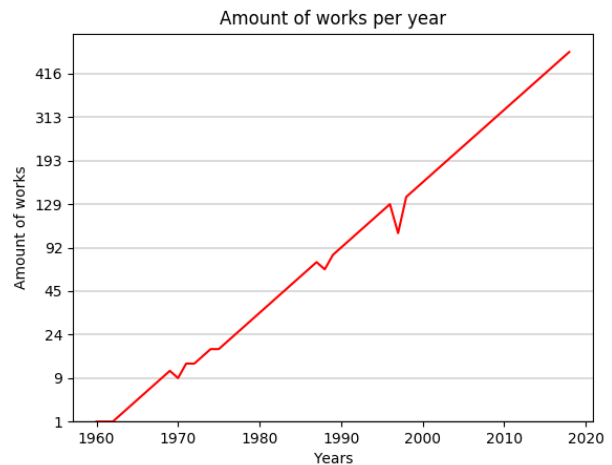


Fig. 9. Amount of works divided by years which were aired for the first time.

- Personal data:
 - Debut
 - Gender
 - Activity years (2018-debut)
- Works data:
 - Amount
 - Top 5 genre
 - Favorites
 - Score
 - Popularity
- Recent works data:
 - Same as works but for only last 9 years
- Graph data:
 - Degree
 - Betweenness centrality
 - Closeness

Fitting and prediction experiments were run for each category, each combination of 2, 3 and all of them together; using 80% as train data and the rest as test. This was done for all following models:

- DecisionTreeRegressor
- DecisionTreeClassifier
- LinearRegression
- KNeighborsClassifier
- LinearDiscriminantAnalysis
- GaussianNB
- SVM

TODO, WRITE THIS AGAIN: To compare prediction performance mean and median absolute error were used. Unfortunately since popularity variance is really high we observed good results in terms of absolute error but particular predictions were a loof. That's why we ended up using r2_score for accuracy comparison.

TODO SHOW TABLE WITH R2 SCORE RESULTS FOR EACH CATEGORY / MODEL AND GROUP OF CATEGORIES

V. CONCLUSION

This section summarizes the paper.

REFERENCES

- [1] J. Hagenauer, E. Offer, and L. Papke. Iterative decoding of binary block and convolutional codes. *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429-445, Mar. 1996.
- [2] T. Mayer, H. Jenkac, and J. Hagenauer. Turbo base-station cooperation for intercell interference cancellation. *IEEE Int. Conf. Commun. (ICC)*, Istanbul, Turkey, pp. 356-361, June 2006.
- [3] J. G. Proakis. *Digital Communications*. McGraw-Hill Book Co., New York, USA, 3rd edition, 1995.
- [4] F. R. Kschischang. Giving a talk: Guidelines for the Preparation and Presentation of Technical Seminars. <http://www.comm.toronto.edu/frank/guide/guide.pdf>.
- [5] IEEE Transactions \LaTeX and Microsoft Word Style Files. <http://www.ieee.org/web/publications/authors/transjnl/index.html>