2021-2022
DA-516 Term Project

Binary Node Classification for Twitch Social Networks Data

Ayda Gizem Kumkumoğlu
Esra Akarçay
Gizem Güneş

Abstract

In the study, the user-to-user Twitch network is held as a network analysis topic. Dataset consists of six different countries as six separate network structures. Spain (ES) network is selected for the analysis to overcome the computation problems. Community detection may be important for social network analysis to discover communities and get insides from the network. Therefore, for the community detection of Twitch dataset, two different approaches have been applied. Firstly, the Louvain method is used and 6 communities are found. Secondly, the Girvan-Newman method is applied. The Louvain method yields a better modularity result for the community detection of the network. For the binary node classification task, 3 different feature datasets are utilized. Firstly, node embedding vectors are calculated and features extracted from node2vec approach are used as a dataset for the prediction task. Second, Rolx algorithm is applied and extracted features are used for the prediction task. Third, node2vec and Rolx features are combined with the own properties of Twitch network data and this merged feature dataset is used for the prediction task. In the binary node classification problem, the Logistic Regression and Random Forest Classifier algorithms are used. The best model is obtained with the node2vec features and Random Forest algorithm with 96% F1 score.

Introduction

Twitch is a live-streaming platform for gamers and other lifestyle casters that launched in 2011. It supports building communities around a shared and streamable interest. Twitch streamers "broadcast" their gameplay or activity by sharing their screen with fans and subscribers who can hear and watch them live.

Users can watch other people playing games, interact with other viewers, or live stream their own gameplay to the world. Lots of different games are streamed, with popular titles such as Fortnite, Teamfight Tactics, League of Legends, and Grand Theft Auto V being among the most watched.

Twitch offers gamers — or anyone interested in lifestyle casting about other subjects like food or music — the ability to stream their activity and let others watch in real-time. Streams can last anywhere from a minute to eight hours and beyond. One can find a stream by browsing various categories, including specific games. If users find a streamer they can like, you can follow their channel and get activity updates and notifications.

The dataset consists of Twitch user-user networks of gamers who stream in a certain language. Nodes are the users themselves and the links are mutual friendships between them. Vertex features are extracted based on the games played and liked, location and streaming habits. Datasets share the same set of node features. These social networks were collected in May 2018.

Dataset Statistics						
	DE	**EN**	**ES**	**FR**	**PT**	**RU**
Nodes	9,498	7,126	4,648	6,549	1,912	4,385
Edges	153,138	35,324	59,382	112,666	31,299	37,304

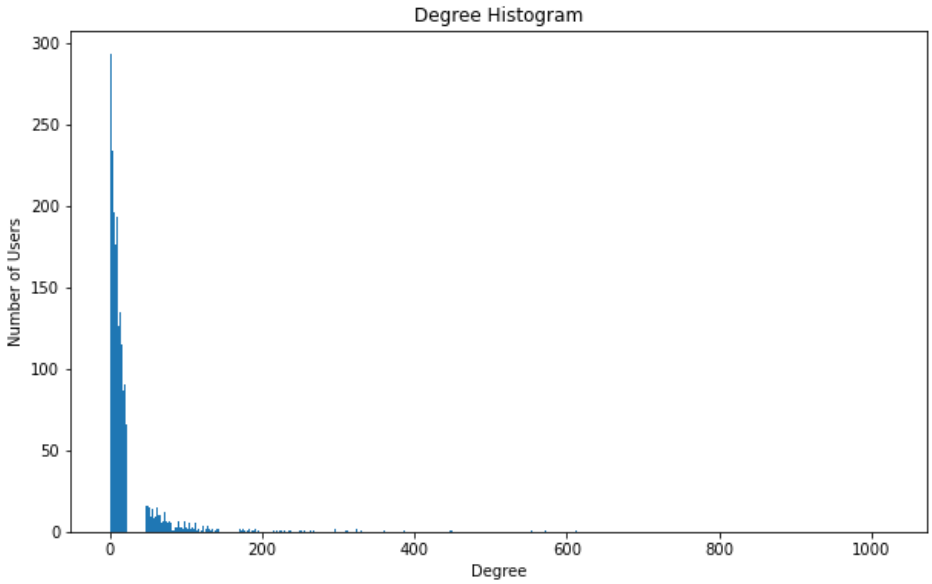
Density	0.003	0.002	0.006	0.005	0.017	0.004
Transitivity	0.047	0.042	0.084	0.054	0.131	0.049

The detailed information about the network of Spain is given below since it is used for the model building purposes.

- Number of nodes : 4,648
- Number of edges : 59,382
- Average degree : 25.5516
- Is directed : False
- Is weighted : False
- Is bipartite : False
- Density : 0.005

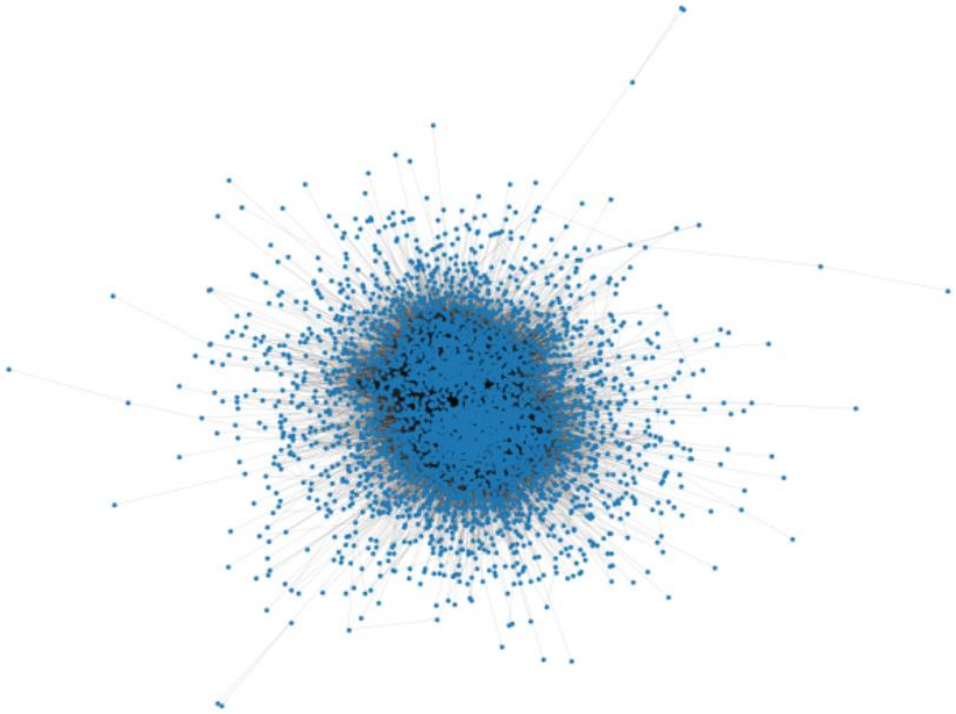
The histogram below shows the degree distribution of the network. Most of the nodes in the network has very few links while a small minority has many links. It can be inferred from the degree distribution that the network follows a Power-Law distribution.

Figure 1. Degree Distribution of Twitch Network



In the graph visualization below, Twitch network has a sparse structure which most of the nodes are connected with just a few of other nodes.

Figure 2. Twitch Network Structure



Community Detection

A community is defined as a subset of nodes within the network such that connections between the nodes are denser than connections with the rest of the network. There can be any number of communities in a given network and they can be of varying sizes. These characteristics make the detection procedure of communities very hard. However, there are many different techniques proposed in the domain of community detection.

There are several algorithms for detection and analysis of community structure. The Louvain community detection algorithm was proposed in this study as a fast community unfolding method for large networks. This approach is based on modularity, which tries to maximize the difference between the actual number of edges in a community and the expected number of edges in the community.

Modularity is a score between -0.5 and 1 which indicates the density of edges within communities with respect to edges outside communities. The closer the modularity is to -0.5 implies non modular clustering and the closer it is to 1 implies fully modular clustering.

The Louvain method is a simple, efficient, and easy-to-implement method for identifying communities in large networks. The method has been used with success for networks of many different types and for sizes up to 100 million nodes and billions of links. The analysis of a typical network of 2 million nodes takes 2 minutes on a standard PC. The method unveils hierarchies of communities and allows zooming within communities to discover sub-communities, sub-sub-communities, etc. It is today one of the most widely used methods for detecting communities in large networks.

The Louvain method is a greedy optimization method that attempts to optimize the "modularity" of a partition of the network. The optimization is performed in two steps. First, the method looks for "small" communities by optimizing modularity locally. Second, it aggregates nodes belonging to the same community and builds a new network whose nodes are the communities. These steps are repeated iteratively until a maximum of modularity is attained

and a hierarchy of communities is produced. Although the exact computational complexity of the method is not known, the method seems to run in time O(N log N) with most of the computational effort spent on the optimization at the first level. Exact modularity optimization is known to be NP-hard.

The Girvan-Newman method is also one of the classic community clustering techniques. The algorithm relies on the iterative elimination of edges that have the highest number of shortest paths between nodes passing through them. By removing edges from the graph one-by-one, the network breaks down into smaller pieces, so-called communities. The algorithm was introduced by Michelle Girvan and Mark Newman.

The Girvan-Newman algorithm is a computationally expensive algorithm since it requires the repeated evaluation for each edge in the system. The highest modularity score of the Girvan-Newman algorithm obtained is 0.00044.

In this study, the Louvain method is selected as the best community detection algorithm due to its convenience and highest modularity score. There are 6 communities detected in the network. The modularity score of the Louvain method is calculated as 0.406.

Node Embedding

The node2vec framework learns low-dimensional representations for nodes in a graph by optimizing a neighborhood preserving objective. Besides reducing the engineering effort, these representations can lead to greater predictive power. In this project, parameters are assigned as;

- Dimensions=20
- walk_length=20
- num_walks=50
- p=1
- q=1
- weight_key=None
- workers=4

It is suggested to use dimensions between the cube root and the square root of the number of nodes. Since the number of nodes in the network is 4,648, the dimension number between 17 and 68 is meaningful. Due to computation issues, 20 is selected as the number of dimensions. Walk length of 20 with the window of 10 and 50 number of walks is found appropriate considering the large dataset. p and q are left at their default values. Since the edges do not have weights there is no weight stated in the node embeddings as well.

Prediction

There is an additional data set which includes attributes of nodes as follows:

- Days: number of days active using
- Views: total number of streamings
- New_id: Id
- Lang: streaming language
- Mature: content of the streaming
- Partner: undefined

For the prediction problem, “views” are used to generate labels. Average total number of views is calculated and this number is set as threshold for binary classification. Nodes receiving views more than this threshold are labeled as 1, others are labeled as 0. This method enables the network for the use of prediction modeling.

- Avg. number of views (threshold): 103,236

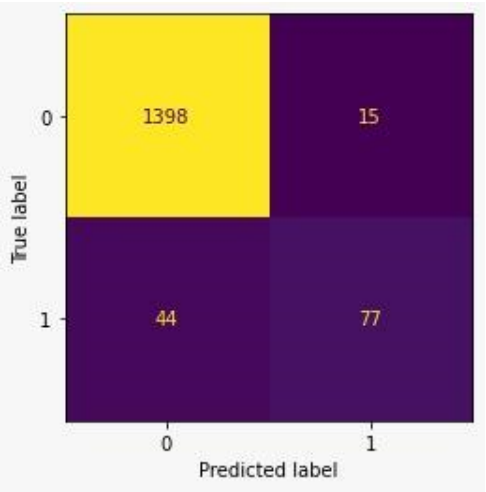
Model Preparation

- **Train-Test-Split:** Dataset is split into train and test data by using train test splitter in scikit-learn, given parameters are:
  Stratify parameter is used to make more robust predictions because of the imbalances of the dataset.
- **Model Selection:**

Datasets	ML Algorithm	Hyperparameters	Accuracy Score	F1 Score
Node2vec features	Logistic Regression	Default	0.9211	0.8833
	Random Forest Classifier	max_depth=20, n_estimators=300, min_samples_leaf=2, min_samples_split=2, criterion='gini'	0.9642	0.9648
Rolx features	Logistic Regression	Default	0.9518	0.9481
	Random Forest Classifier	max_depth=20, n_estimators=300, min_samples_leaf=2, min_samples_split=2, criterion='gini'	0.9577	0.9596
All features (Node2vec, Rolx, network features)	Logistic Regression	Default	0.9518	0.9481
	Random Forest Classifier	max_depth=20, n_estimators=300, min_samples_leaf=2, min_samples_split=2, criterion='gini'	0.9592	0.9615

Random Forest model with all features has shown a better result with 0.9648 F1 score. Result from Random Forest is shown below:

Figure 4. Confusion Matrix for Spain Dataset



	precision	recall	f1-score	support
0	0.97	0.99	0.98	1413
1	0.84	0.64	0.72	121
accuracy			0.96	1534
macro avg	0.90	0.81	0.85	1534
weighted avg	0.96	0.96	0.96	1534

References

1. https://towardsdatascience.com/community-detection-algorithms-9bd8951e7dae
2. https://perso.uclouvain.be/vincent.blondel/research/louvain.html#:~:text=The%20method%20is%20a%20greedy,communities%20by%20optimizing%20modularity%20locally.
3. https://www.pnas.org/doi/10.1073/pnas.0400054101#fig1
4. https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.centrality.girvan_newman.html
5. http://snap.stanford.edu/data/twitch-social-networks.html