Predicting Cast and Director Collaboration using Social Network Analysis

Mahsun ALTIN Serkan KIDIR Abdullah Enes CAN

January 2020

Abstract

This paper introduces an experimental study on social network analysis and an approach that combines the graph analysis and machine learning implementations. We show how the graph embedding are useful to analyze and find relations between nodes in the networks via Node2Vec approach. To test the method, we performed several machine learning algorithms over the data gathered from Internet Movie Database (IMDb). Node2Vec was used to have embeddings of nodes and this is used in the models as feature set to predict collaboration between the actors and directors. The best model that is a logistic regression model is resulted in the ROC AUC score of 0.958, precision of 0.954 and recall of 0.956. As a result of the study, it is seen that the actors and directors that worked together in the past, tend to work together again.

Keywords: Link Prediction, Social Network Analysis, Internet Movie Database, Machine Learning

1 Introduction

Movies play an important role in entertainment area. There are many big companies, that are very old ones, that spend millions of dollars for each film that they make. Since billion dollars worth market is even getting bigger, the role of successful directors and casting can be seen more clearly. Film makers generally work with casting agencies, but it is also known that the network of director is important for finding cast for important roles. The aim of this project is to understand the relationship between cast and directors by using social network analysis methods.

Social network analysis is a complement of methods to understand and analyze social networks. In this study, we have the network that consists of actors, actresses and directors that are connected through movies. Therefore, social network analysis is used in this study to understand and analyze the network of actors, actresses and directors.

Social network analysis is applied to variety of problems in years and there are examples of applications that machine learning algorithms are used to predict features of the networks. A data of network can be used as an input to analyze the nodes and predict some meaningful outputs. The issue of Cambridge Analytica can be an example of this. The data of network of people is used to predict which political party they are going to vote for.[1] Link prediction is on ether example in this field, historical data of links between nodes is used to predict possible links.

In this project, since the objective is to understand and predict the relationship between cast and directors, the network of these people may help us to see the relationships. One way was to use link prediction methods to see which of these people worked together in the past and they would work together again. Therefore, this paper focuses on predicting the possible future links between cast by using historical network data. By comparing predicted values with the realized ones, it is aimed to see which of these people are more tend to work with the same people.

2 Literature Reviews & Related Works

The project may be considered under the scope of growing and developing social network analysis and machine learning disciplines. For this reason, there is a possibility to happen upon a variety of associated studies as regards the network analysis field. The project requires movie data that is used in creating a network in order to analyze cooperation between directors and actors/actresses, considering that reviewing the studies which aim for social network analysis and using IMDb data are notable. By the same token, the studies that are done under the scope of machine learning, and network analysis topics were helpful and had a valuable contribution to the project.

Graphs are well-known data types, frequently used in computer science and related disciplines. Many systems can be effectively modelled by graphs such as social networks, ecosystems, protein structures etc. They allow relational knowledge about interacting nodes. When we need predictions over the graphs by using machine learning algorithms, we need an approach to convert graphs into d-dimensional vectors. For that reason, we need representation learning that is done by graph embeddings. We use it because it helps input data to be generalised better. Embeddings may vary such as vertex embeddings, graph embeddings, edge embeddings. In short, we obtain vectorization of each vertex for vertex embeddings. For graph embeddings, we obtain a whole graph as vector representation.

Since our project aims predictions of edges between nodes, Node2Vec was the exact approach that we needed. Nodes and edges are the critical points for the evaluations and metrics to analyze the networks. Because some of the necessary tasks in network analysis include the forecastings of nodes and edges. [4] With the objective of using network components like features and machine learning ingredients, the graph should be embedded. Although graphs are a useful way to represent data, they have some limitations when it comes to implementation in machine learning. “Graph embedding are the transformation of property graphs to a vector or a set of vectors.” [3] By graph, node, and link embedding, we have a powerful tool on our hands. Because such network relationships can use only a particular subset of machine learning, statistics, and math, while vector spaces have a wider variety of approaches. Through that, graph topology and relevant information regarding node to node relationship can be obtained, such as similarities between nodes. Alternative solutions to embedding such as DeepWalk [6], LINE, and SDNE are available in addition to Node2Vec.[7] DeepWalk and Node2Vec are probability-based approaches, which use random walks to get paths from the graph and measure the probability of co-occurrence, which are performed to get embedding for each node through the Skip-gram model.

Some experimental results, however, indicate that Node2Vec is the one that gives better results, especially in the prediction of links. [4]

As we searched through the studies, there are also a variety of experimental works done over IMDB data. Yet, most of them are related to movie recommendation systems [5] or financial predictions [2] of movies that aim to infer the trend of movies via social networks. In addition to that, some research introduced an analysis of the relationships movie producing teams and their success. [8]

3 Methodology

This paper aims to predict link between a director and a cast who have acted in the same future movie. To be able to predict collaborations, we needed to collect a dataset, which should include "train and test" . The IMDb dataset and The Movie DB were used as data sources. The gathered data was transformed into node embeddings by using Node2Vec. These embeddings were later used for predicting collaborations.

3.1 Data Explanation

There are several sources which include movie information such as cast, director, writer, production year, grades of movies and etc. As several examples for these sources, IMDb, The Movie DB and Movie Lens database can be given.

3.1.1 IMDb

IMDb has several datasets for non-commercial projects, we analysed mainly 3 of the datasets and their information to get clear and understandable data. Details of the datasets are shared below. The used data was gathered by merging these datasets.

- title.basics.tsv.gz:
  - id of title
  - the type of the title (i.e. movie, tv series, tv shows etc.)
  - popular name of the title
  - the release year of the title
  - boolean value if the title is adult
  - runtime of the title
  - genres of the title
- name.basics.tsv.gz:
  - id of the person
  - name of the person who has take charge in the movie
  - birth year and death year of the person
  - the top-3 professions of the person
  - titles the person is known for (see. the maximum number of titles are 8)
- title.principals.tsv.gz:
  - id of movies
  - id of name
  - the category of job that person was in

title.principals.tsv.gz is the data of cast of movies. title.basics.tsv.gz was used to eliminate movies according to years(e.g. movies that were played before 2000 were eliminated) and to eliminate different types of movies(e.g. TV Episodes, Short Movies, Animations etc were eliminated). name.basics.tsv.gz was used to learn the directors of the movies.

3.1.2 The Movie DB

The Movie DB has a powerful API service, which has rate limit requests to 40 requests every 10 seconds. Therefore, we easily could be able to get very large scale datas from The Movie DB. The Movie DB API has the feature which allows to query for external id's like IMDb. Therefore, firstly we prepared a movie list by analysing data from IMDb, then, we pulled the cast and directors of movies which the API service of The Movie DB has. However, after processing data, when we examine the dataset, we saw that there were problems with lacks of data. At the end we had less edges, which are 937579 edges, between directors and casts than edges, which are 1492505 edges, after processing of the IMDb. Therefore, we continue with IMDb dataset.

3.2 Data Preprocessing

In data preprocessing part, we eliminate the "title.principals.tsv.gz" dataset like below:

1. The titles which have the type of "movie"
2. We gave 0 to production year for movies which has no data about production year
3. Movies in IMDb dataset which were produced before 2000, since we try to predict future edges.
4. Death people who have take charge in the movies.
5. We rename the director and cast data with adding prefix as;

- nm0158032 with d_nm0158032
- nm0070239 with c_nm0070239

After these eliminations, we created edges between cast and directors who has take charge in the same movie. Then, we count these edges to find weighted edges. At the end, we totally have 591,723 weighted edges between 91,425 director and 311,686 actors.

3.3 Network Graph & Node2Vec

Network of actors and directors is used in representation learning. The data is prepared according to years. In order to get embedding vectors of nodes, we used Node2Vec. We used a data that consists of past collaborations between actors and directors to gather embedding vectors of nodes.

We get embeddings of nodes with the data of 2000 to 2009. For these years, we checked if actors and directors played in any of movies, together or not, between the years 2010 – 2014. Therefore, the training data consists of distance vectors of two nodes that is gained from the data of 2000 to 2009 and a dependent variable that shows whether these two nodes worked together in the past, or not. We used the data of 2000 – 2009 to get embedding vectors of actors and directors. Therefore, if they did not play in any of movie between these years, they will not have embedding vectors and they will be eliminated from the training data.

We gathered edge embeddings from the data of 2000 – 2009 and for the collaborations between years 2010 – 2014, we used the difference of embedding vectors of collaborated actors and directors to get their distance vector in multidimensional space. This data was used as independent variables training data, which has 16 dimensions that means 16 independent variables in our case.

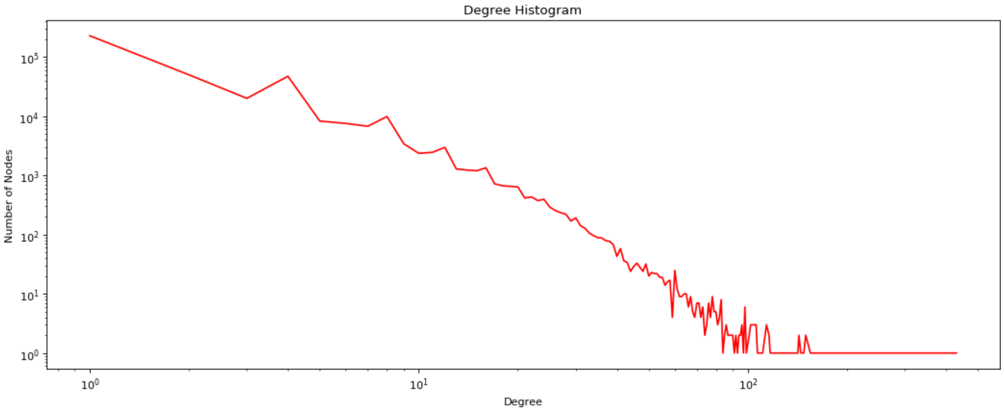
In a similar way, we used the data of 2005 – 2014 to get embedding vectors of actors and directors. These embedding vectors are used with the data of 2015 – 2019 collaborations. We have embedding vector of an actor or director if he/she played in any of films between 2005 – 2014. Therefore, we used embedding vectors as independent variables' test data and used the data of collaborations between 2015 – 2019 as dependent variable's data. This constructs the test data.

We populated false samples for the training data randomly. We matched actors and directors that did not work together between the years 2010 and 2014. Node embeddings of these actors and directors were gathered from Node2Vec output of the 2000 - 2009 data. The data with 12 dimensions includes 29411 true samples and 5063 false samples in the training data and 42130 true samples and 7313 false samples in the test data. The data with 16 dimensions includes 29166 true samples and 24782 false samples in the training data and 41921 true samples and 7247 false samples in the test data.

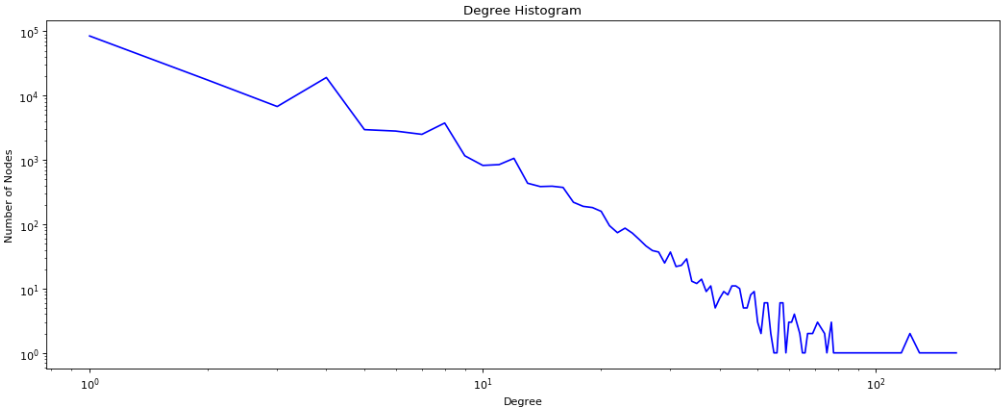
As the result of this process, we created training and test data which is needed to predict future links. In our case, we will predict the collaborations between years 2015 – 2019 and compare the predicted links with the realized ones.

We created 4 bipartite and weighted graphs, since we didn't want to predict edges between cast and cast or director and director. We splitted data into 4 parts while getting node embedding vectors. These parts were 2000 to 2009, 2010 to 2014 training data and 2005 to 2014, 2015 to 2019. It is considered to have network of these sets after analyzing the whole set together. Descriptive statistics of these networks are shared and illustrated below.

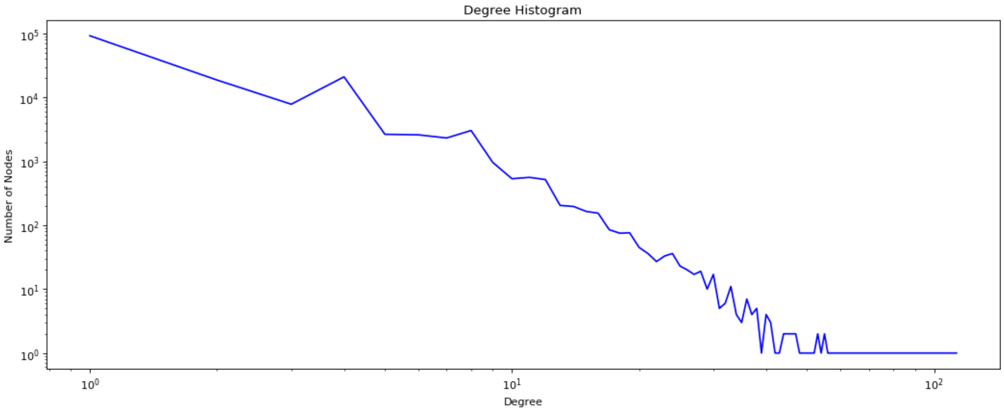
1 - Whole data, 2000 to 2019
Director: 91425 and Cast: 311686
Number of nodes: 403111
Number of edges: 591723
Average degree: 2.9358



2 - Training data, 2000 to 2009
Director: 107533 and Cast: 32639
Number of nodes: 140172
Number of edges: 185614
Average degree: 2.6484



3 - Training data, 2010 to 2014
Director: 116993 and Cast: 36731
Number of nodes: 153724
Number of edges: 178193
Average degree: 2.3183



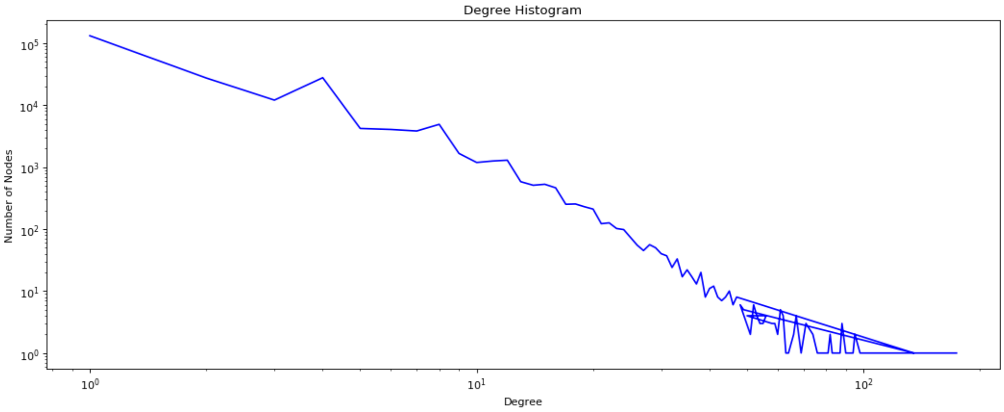
4 - Test data, 2005 to 2014

Director: 172172 and Cast: 52528

Number of nodes: 224700

Number of edges: 290599

Average degree: 2.5866



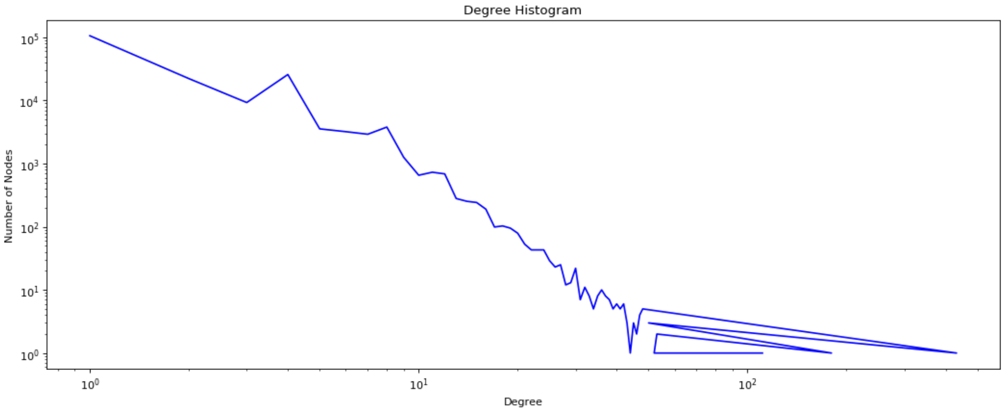
5 - Test data, 2015 to 2019

Director: 138131 and Cast: 43184

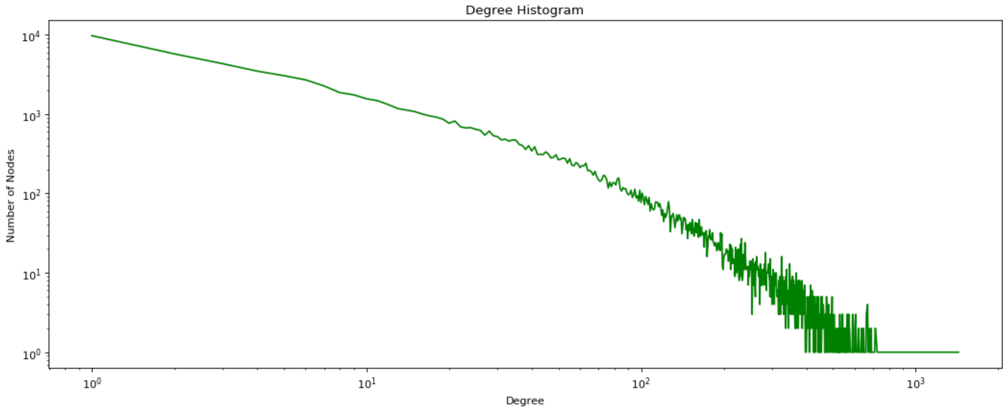
Number of nodes: 181315

Number of edges: 218102

Average degree: 2.4058



6 - Director Projection of Whole Data, 2000 to 2019



Number of nodes: 75352 Number of edges: 1315442 Average degree: 34.9146

In order to analyze the network further, we projected the network onto the directors. We gathered the pagerank centrality scores of the directors. The gathered data of the projected network is shared below.

Pagerank centrality scores of the directors are gathered and it is seen that the directors with the highest pagerank centrality scores are;

1- Jing Wong - 0.000248
2- Uwe Boll - 0.000246
3- Herman Yau - 0.000229
4- Takashi Miike - 0.000227
5- Priyadarshan - 0.000218

Two different Node2Vec models were created with different parameters. The first model was created with 8 walks, the walk length of 50 and dimension of 16. The second model was created with 6 walks, the walk length of 40 and dimension of 12. Weighted graphs of the data of 2000 to 2009 and 2005 to 2014 in the training and test sets were used in Node2Vec.

3.4 Link Prediction

In this section, we apply some prediction techniques with data gathered from node embedding. These techniques are logistic regression and neural network which has a simple structure.

3.4.1 Logistic Regression

Logistic regression is a technique that is used in order to predict the future edges. The reason of using logistic regression is that it is an appropriate technique if the dependent variable is binary. As we discussed in Node2Vec part above, we have embedding vectors of size 16 and 12 as the independent variable and a binary variable that explains the collaboration between the actor and director (e.g. 1 if they worked together in past, 0 otherwise.).

4 Results & Experimental Results & Findings

As it is explained in methodology part, we created logistic regression model model to predict the collaboration between actors, actresses and directors. The data was split into training and test sets; embedding vectors are gathered from the data of 2000 to 2009 and the dependent variable's data is gathered from the data of 2010 to 2014 in the training data, embedding vectors are gathered from the data of 2005 to 2014 and the dependent variable's data is gathered from the data of 2014 to 2019 in the test data. The training data includes 929770 samples of links. The test data includes 444909 samples of links.

4.1 Logistic Regression

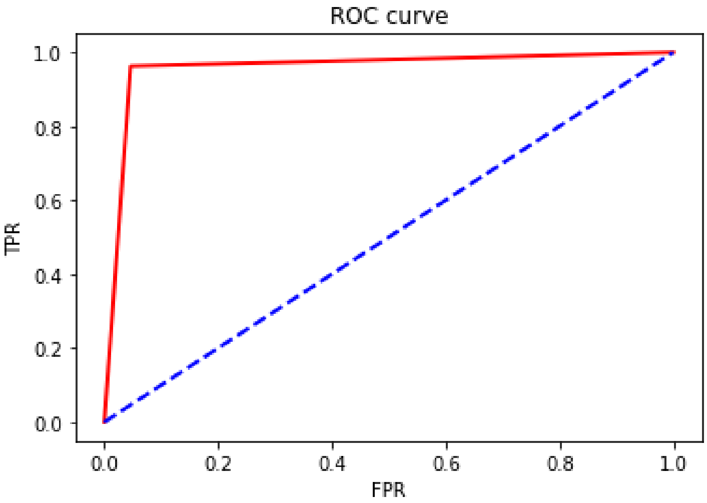
Two logistic regression models were built with 16 and 12 independent variables that show the distance vector of an actor and director. Dependent variable is whether they have worked together in the same film or not, which means it is a binary variable.

In logistic regression, since the dependent variable is binary, we are interested in true negative and true positive predictions. As it can be seen below, according to the predicted and realised values, we get the number of false negative and false positive predictions from the model that has 16 independent variables. Proportion of true positive and true negative predictions to all gives the score of the model as a performance metric.

	Actual	
	Positive	**Negative**
Predicted Positive	23634	1148
Predicted Negative	1093	28073

We gathered the related information from our model and it is shared below. As it can be seen, there are 53,948 predictions, 23,634 of these predictions are true positive and 28,073 of them are true negative. Proportion of these to the total number of predictions is 0.958. This can be seen also from the ROC Curve above.

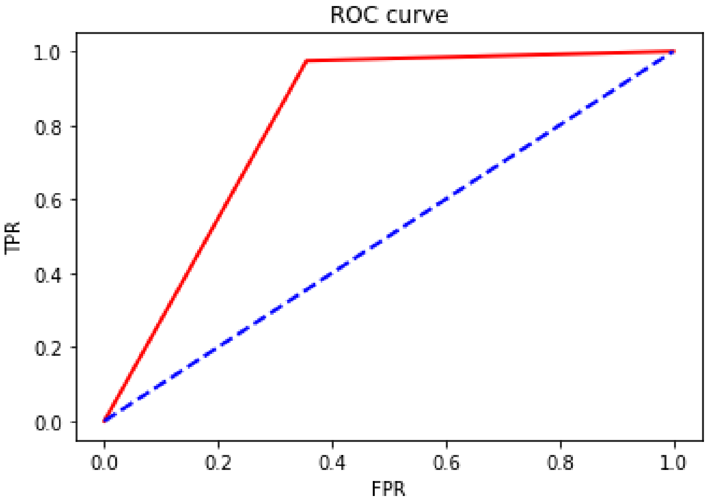
Receiver Operating Characteristic (ROC) curve summarizes the model's performance by illustrating the trade-off between false positive and false negative. The ROC curve of the model is shared below.



The area under the curve is 0.958 that gives ROC AUC Score of the model. Additionally, it is seen that precision of the model is 0.954 and recall of the model is 0.956.

The model with 12 independent variables was built and results of the model is shared below.

	Actual	
	Positive	**Negative**
Predicted Positive	3263	1800
Predicted Negative	743	28668



The area under the curve is 0.877 that gives ROC AUC Score of the model. Additionally, it is seen that precision of the model is 0.644 and recall of the model is 0.814.

It is seen that the model was resulted better with the first Node2Vec model that has the dimension of 16. The comparison of the models is shared below.

5 Conclusion

This study was conducted with the objective of understanding the collaboration between cast and directors. For this objective, we are decided to have link predictions of cast and directors and see whether they are generally working together or not. The data is gathered from IMDb dataset and prepossessed. The information of cast and directors that worked together between 2000 and 2009 is used to gather embeddings and these embeddings are used with data of 2010 to 2014 as the training data, the data of 2005 to 2014 is used to gather embeddings and these embeddings are used with the data of 2015 to 2019 as the test data. In order to predict links, we worked with Node2Vec to get a embedding vectors of the nodes. These embedding vectors are used as features in the prediction models. Since we are predicting the links between them, the dependent variable is considered to be a binary variable that is 1 if they are going to work together in the past, 0 otherwise. Therefore, logistic regression is thought to be proper option as a model. We constructed logistic regression model and it is seen that logistic regression model resulted in the ROC AUC score of 0.958, precision of 0.954 and recall of 0.956. As a result of the analysis, it is seen that there are some directors that tend to work with the same people most of the time. Similarly, it is seen that there are many cast that tend to work with the same director most of the time.

References

[1] Juan Pablo Alperin. Politicians the public: The analysis of political communication in social media, Feb 2019.

[2] Lyric Doshi, Jonas Krauss, Stefan Nann, and Peter Gloor. Predicting movie prices through dynamic social network analysis. *Procedia - Social and Behavioral Sciences*, 2(4):6423–6433, 2010.

[3] Primož Godec. Graph embeddings-the summary, Jun 2019.

[4] Aditya Grover and Jure Leskovec. node2vec. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD 16*, 2016.

[5] Jelena Grujić. Movies recommendation networks as bipartite graphs. *Computational Science – ICCS 2008 Lecture Notes in Computer Science*, page 576–583, 2008.

[6] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD 14*, 2014.

[7] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077. International World Wide Web Conferences Steering Committee, 2015.

[8] Wladston Viana, Pedro Onofre Santos, Ana Paula Couto Da Silva, and Mirella M. Moro. A network analysis on movie producing teams and their success. *2014 9th Latin American Web Congress*, 2014.