# A recommender systems and social networking approach to alleviate the issue of cold start

Research Project
MSc Data Analytics October 2020/21

Mahesh Arjun Matele
Student ID: 19179065

School of Computing
National College of Ireland

Supervisor: Paul Stynes and Pramod Pathak

**National College of Ireland**
**Project Submission Sheet**
**School of Computing**

| | |
|---|---|
| **Student Name:** | Mahesh Arjun Matele |
| **Student ID:** | 19179065 |
| **Programme:** | Msc. Data Analytics |
| **Year:** | September 2020/21 |
| **Module:** | Research Project |
| **Supervisor:** | Paul Stynes and Pramod Pathak |
| **Submission Due Date:** | 16/08/2021 |
| **Project Title:** | A recommender systems and social networking approach to alleviate the issue of cold start |
| **Word Count:** | XXX |
| **Page Count:** | 13 |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

**ALL** internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

| | |
|---|---|
| **Signature:** | |
| **Date:** | 19th September 2021 |

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies). | ☐ |
| **Attach a Moodle submission receipt of the online project submission**, to each project (including multiple copies). | ☐ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | ☐ |

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# A recommender systems and social networking approach to alleviate the issue of cold start

Mahesh Arjun Matele
19179065

**Abstract**

Cold start is the most frequent issue faced by recommender systems (RS). The reason for its happening is because it happens to ever new user that enters the system. A good RS is characterized by its recommendations, which completely depend upon the user's preferences to watch movies of genres it likes. Cold start is when the new user enters the system and none of his preferences are available to RS to make recommendations.As the user is new it hasn't set any preferences in the system or hasn't rated an movies, because of which no history for a user is available with RS.

As part of this research we proposed a classification supervised model to recommend movies to user with the help of online social networks (OSN). The combination of OSN along with the user's demographic data is used to recommend movies to the user. Which also assists the goal of the research to recommend movies with minimal demographic data. The options for fetching the OSN data is Twitter, Amazon or simply MovieLens which can be fed to the RS. This will contribute to the RS community to recommend movies to a new user with a minimal number of demographic variables.

When conducting the experiment we were able to achieve AUC-ROC curve of 0.70 using hyperparameterized KNN algorithm and were able to recommend movies of different genres to the user. The start-of-the-art experiment of J. Herce-Zelaya (2020) uses Random Forest and scored a Mean Absolute Error of 0.298 with 12 demographic variables from the twitter profile of the user.The tuned KNN algorithm was based on finding the optimal K neighbours using silhouette method. The optimized values for leaf node was 28 distance calculation between data points was most efficient using manhattan method with best neighbours values of 28 using brute algorithm.

As mentioned earlier the AUC-ROC curve of 0.69 was achieved as part of this research with a precision of 0.68 and a recall score of 0.76 using KNN which is an improvement over the state-of-the-art experiment by 5% on recall score while keeping the same precision score. Besides, this the state-of-the-art experiment uses 12 demographic variables whereas the score achieved in this experiment uses 2 demographic variables namely gender and age to recommend movies. The infrastructure and other limitations and challenges faced during the experiment are highlighted in the challenges 3.1 section.

For further studies on this research topic for the community of enthusiasts this research has set a benchmark of 2 demographic variables with precision at par with the state-of-the-art experiment. The other research can continue with advanced machine learning models and try to improve the precision with greater number of demographic variables.

# Contents



# List of Figures

# 1 Introduction

**Background:** Historically the RS presented recommendations based on the user's stored data over time on preferences. It scanned across users likes, interests, preferences to make a set of suitable recommendations Singh et al. (2020). Usually, as soon as a user logs in to the system it is presented with a set of recommendations which are achieved by the RS after going through several algorithms. Methodically the RS are bifurcated into 3 main branches based on the algorithms used to provide recommendations. These methods are content-based, collaborative and hybrid RS.Hybrid RS are a combination of content-based and collaborative RS Rahul et al. (2021) .These methods were considered as the first generation of RS, which further evolved into matrix factorization, web usage mining based and personality based. Further, the evolved into the third generation which makes use of deep learning techniques to provider recommendations. Content-based RS provides recommendations based on the user's preferences, collaborative as name suggests provides recommendations based on other users with similar preferences TK et al. (n.d.). Despite the evolution, the issue of cold start still remain in RS and as part of this research has looked into it more in order to suggest a solution.
**Motivation:**From the studies we can see there are 2 aspects to a cold start issue, one that emanates from the items and other from the users Natarajan et al. (2020). Cold start from users is seen when a new user enters the system with no preferences where a cold start from the item or movie in this case is seen when a new movie enters the system and because it is not rated and hence never shown as a recommendation thus causing the issue of data sparsity. The base paper on this research of J. Herce-Zelaya (2020) proposed a solution using the Machine Learning algorithm of Random Forest to make recommendations. As part of that research 12 demographic variables from the twitter profile of the new user were used to make a recommendation. As part of this research we have improved this research with a similar dataset but with a lesser number of demographic variables of the user.The recommender system performs its best when it supplied with a quality of demographic variables more than quantity of demographic variables.
**Research objective:** Accordingly, the aim of this research is to:

- Investigate to what extent a recommender system and social networking approach solves cold start. This can be achieved by usage of a social media network which provides user demographics in order to recommend a movie which suits his demographic profile.

- Make relevant recommendations to the end-user with a smaller number of demographic variables.This can be achieved by finding the best possible, least amount of demographic variables that can help achieve the same or better precision as that of the state-of-the-art experiment by J. Herce-Zelaya (2020).

**Structure of paper:** To achieve the above the research paper is structured as section 1 for background and objective of research. Section 2 provided other authors work on the cold start issue. Section 3 provides the details of how the experiment is conducted. Section 4 provides the evaluation of the experiments conducted as part of the research. And finally the section 5 provides the conclusion and scope of future work.

# 2 Literature Review

This section will elaborate the studies done by other authors on the subject of cold start in RS. From the many studies conducted by authors we will be discussing the most latest start-of-the-art solutions which makes use of OSN as part of the research.
The base paper used to improve on is the work done by J. Herce-Zelaya (2020) where Twitter is used as a OSN to fetch user demographic attributes and Random Forest and other decision tree classifier machine learning algorithms are used to create recommendations.Considering this, we have improved upon the research by using classification and clustering algorithms in order to assign the new user to the right cluster for recommendations.

## 2.1 Cold start issue with machine learning algorithms

In the next paragraphs we will be elaborating on the research done by authors with OSN and several other sources used as a database for gaining new user insights about its demographic data and preferences and ratings provided to movies.
User-item matrices, classification and clustering are some of the most common ways to provide recommendations as per the study by Reddy et al. (2019). This study by Reddy et al. (2019) alters the standard approach by adding tag-based and genre as parameters to the RS along with euclidean distance as a measure for calculation of similarity. Study by Ahuja et al. (2019) uses one of the traditional approaches of clustering using the Within-Cluster Sum of Squares (WCSS) approach. WCSS helps in getting the measure of change in the observations within each cluster. In this approach, a cluster with small sum of squares is smaller and compact than the cluster which has a larger sum of squares. So, for our study we have used the concept of user-item matrices from this experiment to get a good RMSE.
Similarly, Mokarrama et al. (2020) proposed a solution of centralized database of demographics so that whenever a new user enters the system a query based approach is used to get the releavant recommendations for the new user. This is similar to another approach of DBPedia. The downside of this approach is before making a query to the database certain preferences of the current user should be supplied to get the recommendations, which does not suit our research as our research proposes a method where no preferences from the user are provided.
Optimization of the algorithm also hold a lot of significance in the experiment as the recommendations should be provided in a least amount of time. Chen et al. (2020) proposes optimization using hyper parameter tuning calling it the Collaborative Filtering Recommendation algorithm based on User Correlation and Evolutionary Clustering (UCEC&CF) which is run against other state-of-the-art approaches like Document Type Description(DTD), Dynamic Particle Swarm Optimization (DPSO) and seems to have provided a better RMSE and ina a lesser amount of time. Besides using the user-item matrix it also optimizes the parameter to find which is the best distance metric to be used for the dataset. Singh et al. (2020) also has conducted a siilar study where the research proposes the best similarity distances amongst - euclidean, pearsons or cosine.
So, using the studies from latest research by Chen and Tang (2019), Natarajan et al. (2020), Singh et al. (2020). we can also base our research on other studies from Eklaspur and Pashupatimath (2015), Rahul et al. (2021), Wibowo et al. (2020), Zarzour et al. (2020) to improve the work done by J. Herce-Zelaya (2020).

## 2.2 Ethics

The datasets used in the experiment contains no information that can identify a user or its personal information. The datasets were collected by the GroupLens Research Project at the University of Minnesota and they bear no responsibilities for the impact of the data as per the disclaimer. Also, this dataset as per the disclaimer follows the GDPR rules and can be used for educational purposes and prohibits any endorsements or redistribution.

# 3 Methodology

As per the design of the experiment implemented in python 3.2 below are the steps followed:

- Get movie review given by a user from SNAP database/MovieLens 100K Dataset.
- Match the movie review with user and get user demographics.
- create a user-movie matrix to create a user-movie profile.
- Pass new user demographics to get recommendations based on similar user-product profile.

Below is the high level diagrammatic in figure 1 representing the same:

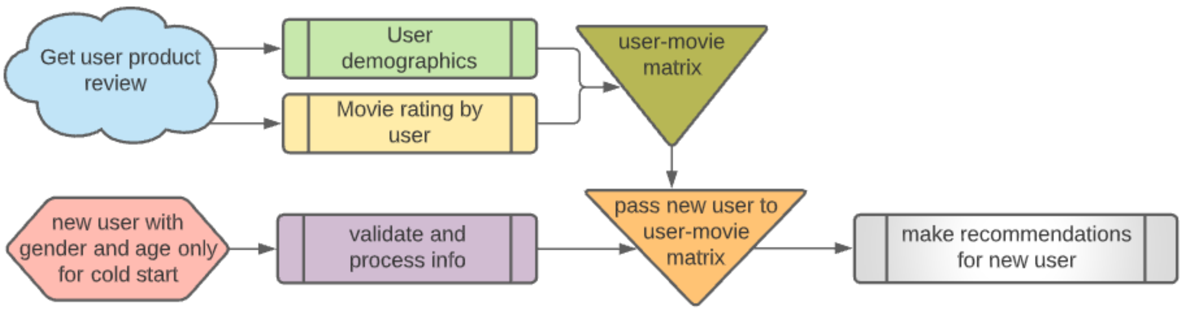


Figure 1: High level diagram of experiment

## 3.1 Challenges

The challenges faced during the experiment are as below: The base paper from J. Herce-Zelaya (2020) uses filmaffinity.com as a OSN and twitter.com to get the movie ratings provided by the user. Because the dataset used by the base user has changed overtime with some user profiles being absent from twitter. Also, a developer account access is required to get access for user-profiles on twitter which match with filmaffinity [1] movies database which takes time for approval. To mitigate this instead of using filmaffinity and twitter as a combination we used amazon datasets to get user demographics and ratings from the Stanford Large Network Dataset Collection [2] website. As the aim of the experiment to address cold start issue still remains unaltered. Next challenge faced

[1]MovieLens: `https://www.filmaffinity.com/ie/main.html`
[2]SNAP: `http://snap.stanford.edu/data/web-Movies.html`

during the experiment is the getting amazon user demographics [3] requires a amazon developer account which is again related to approval of access for a developer account, hence we used the MovieLens [4] dataset which is a dataset for educational purposes with user demographics and movie ratings.Again, the motive of the experiment remains unaltered.

From the infrastructure perspective python was used the programming language. Jupyter Notebook and google colabs was used for improving the processing power. The amazon dataset consisted of 8 million records, of which an odd 50k records were records with valid records with user reviews for more than 50 movies. But, given the size of the dataset google colabs was not able to filter them out. For movielens which was a dataset of 100k records the dataset was filtered as it was in a structured format unlike the case with amazon datasets.

## 3.2 Data Collection

As mentioned in the challenges section, the final dataset which is chosen for the experiment is MovieLens 100K dataset collected at GroupLens Research Project at the University of Minnesota. These datasets consists of all the required varibles including the user demographics and movies informations and ratings provided by the users.This information can be used for simulation of he cold start issue and ways to improve the base experiment. From the 2 ways in which the data can be made available to experiment namely - fetching the dataset from the website at runtime using webscraping and secondly making the data available locally by downloading to local disk and using it for the experiment; of these we chose the latter option as that reduces the processing time and helps in keeping the results consistent.

The datasets fetched from MovieLens has the below dataset statistics:

- 100K ratings between 1 and 5 from 943 users on 1682 movies
- The ratings provided are for more than 20 movies per user.
- Demographic information of the user has age, gender, occupation, zipcode.
- Number of users - 943
- Number of movie items - 1682
- Number of ratings for movie items altogether - 100K

Ratings is a comma separated values dataset providing the user and ratings provided by a user to movies.

- userId - unique identifier for a user.
- movieId - is a unique identifier for movies
- rating - rating provided by user to movie.
- timestamp - timestamp when the ratings were provided

[3] Amazon user info: `https://docs.aws.amazon.com/connect/latest/APIReference/API_DescribeUser.html`

[4] MovieLens 100K Dataset: `https://grouplens.org/datasets/movielens/100k/`

user's demographic information is as below:

- userId - unique identifier for a user.
- age
- gender
- occupation
- zip code

Movie information:

- movieId - is a unique identifier for movies
- title - is a title for movies
- genres - is an genre for movies, pipe delimited in case of multiple genres.

### 3.3 Data Preparation

While data processing, once the data is available locally on the machine it is not fetched again at run time so maintain the consistency of the results. As mentioned in the 3.2 the data is available in the local system in csv format. Once the data is available it is made available in the coding environment a statistical summary of the dataset is derived to find the count and standard deviation of the ratings for movies of different genres. As the absence or imbalance of dataset for various genres can affect the recommendations for the users. For this purpose a class distribution was attempted to find the size of the number of records for ratings of different genres. This was further elaborated using univariate plots to find the outliers and imbalance of data. From the univariate and multivariate plots it was determined that all the genres were uniformly distributed and no change was required except for the data sparsity caused due to null records where a movie was not rated by any user. These missing values contributed to sparsity and is replaced as 0 for numerical values.Further a test harness was designed with 10 fold cross validation as shown in the figure 7 . This split was of 10 parts with train on 9 and test on 1 to repeat for all combinations of train-test splits.

## 4 Evaluation

The state-of-the-art experiment makes use of twitter profiles of user's present on filmaffinity website to check their ratings. Consequently, profiles are created for recommending movies using similar set of demographic profiles thus tackling the cold start issue. This is conducted using Random Forest algorithm to get an average MAE of 0.298 which can be a precision of 0.70 .We have tried to simulate the same experiment but as mentioned in the challenges section 3.1 we have used a similar set of data demographics. Based on this, we have applied Random Forest to predict the accuracy of the model which scored a precision of 0.68 (+/- 0.03), Recall of 0.80 (+/- 0.05) and roc-auc of 0.70 (+/- 0.02) which is near to the state-of-the-art experiment and on similar lines. The aim of the experiments conducted is to be successfully answer the questions posed int 1 research objectives. Of this the first research objective of improving the score of state-of-the-art experiment is achieved by 4.2 whereas the second objective of keeping the precision at par with state-of-the-art experiment and accomplishing it with lesser number of demographic variables is achieved by 4.3.

Below is a diagrammatic representation 2 of the data collection and execution of the experiment if the challenges are mitigated. This methodology is common for all the experiments conducted, only the model is changed to improve the accuracy and output.

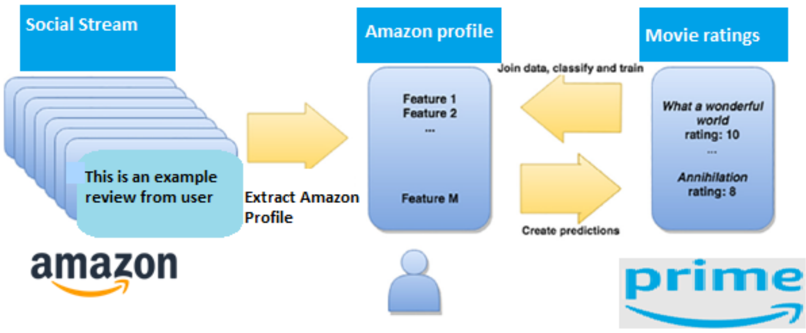


Figure 2: High level diagram of experiment if challenges are mitigated

## 4.1 Experiment 1 : state-of-the-art steps to achieve a result that is near to it

This section provides the metrics achieved by state-of-the-art experiment by J. Herce-Zelaya (2020). As shown in the data collection 3.2 section combination of the datasets is used to create a profile of preferences chosen by a set of user's belonging to the same demographics. This information is used by machine learning algorithm Random Forest to recommend new user's movies. These new users have not registered any movie preferences in the system thus concerning the system with a cold start issue. For the experiment, the demographics provided by the user to the system for recommendation is age and gender present in the dataset.

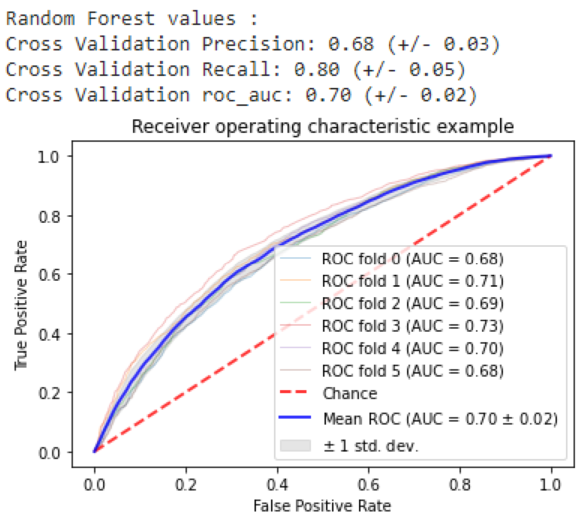


Figure 3: Random Forest experiment

The prowess of a recommendation system lies in the relevance of recommendations made by the system. Below figure shows the recommendations made by Random Forest algorithm for a user who is male and of age 25 years.It shows a mix of Thriller, Adventure and Children's movies as depicted in the movies.

| Random Forest movie recommendations | | | | | |
|---|---|---|---|---|---|
| Movie Name | Year release | Genre | | | Rating(1 more than 3 & 0 is less than 4) |
| From Russia with Love | 1963 | Action | | | 1 |
| Live and Let Die | 1973 | Action | | | 0 |
| Star Wars: Episode V - The Empire Strikes Back | 1980 | Action | Adventure | Drama | 1 |
| Mad Max | 1979 | Action | Sci-Fi | | 1 |
| Peter Pan | 1953 | Animation | Children's | Musical | 0 |
| Reality Bites | 1994 | Comedy | Drama | | 0 |
| Big | 1988 | Comedy | Fantasy | | 1 |
| Enchanted April | 1991 | Drama | | | 0 |
| Them! | 1954 | Sci-Fi | Thriller | War | 1 |
| Duel in the Sun | 1946 | Western | | | 0 |

Figure 4: Random Forest movie recommendations

To improve on the state-of-the-art experiment we executed the same dataset with gradient booster with hyper-parameter tuning. But that didn't improve the results achieved by Random Forest. The result achieved by Gradient Booster showed a lesser ROC-AUC area under curve and also a lesser precision as compared to Random Forest algorithm. The table in figure 5 shows the comparison of the precision, recall and area under the curve values for state-of-the-art experiment and an attempt at improving the scores.

```
Gradient booster values :
Cross Validation Precision: 0.64 (+/- 0.01)
Cross Validation Recall: 0.84 (+/- 0.04)
Cross Validation roc_auc: 0.67 (+/- 0.02)
```

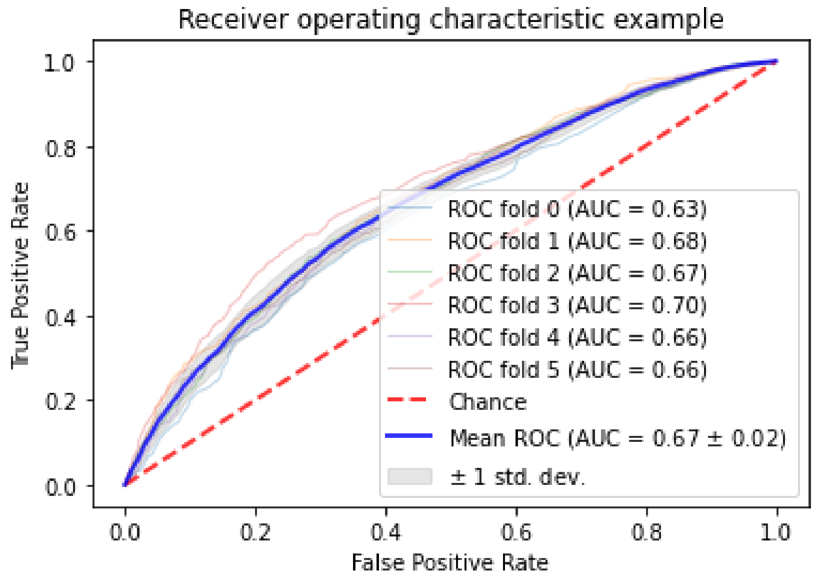


Figure 5: Gradient Booster experiment

An attempt to improve the score using TF-IDF VECTOR and cosine similarity for an existing user was also experimented to check the content based recommendations for a user. This yielded good results, but it worked for users who already had preferences set in the system and didn't assist the cold start issue where there are no preferences supplied by the user.

## 4.2 Experiment 2 : experiment conducted using KNN

As part of the second section of the experiment we repeated the same steps of experiment 1 4.1, with K Nearest Neighbour(KNN) algorithm. KNN is a classification algorithm which is easier to interpret, consumes less calculation time and greater precision power as compared to Random Forest. Random Forest is a tree structure algorithm, where KNN is a classification approach which suits the RS approach with larger amount of data.Hence, in this experiment we used KNN as an alternative approach.
As per the experiment, the classification of the user's profile based on the combination of age, gender, year, genre1, genre2, genre3 is determined. When a new user enters the system with certain and age and gender the KNN model is used for determining the choice of the movie the new user would like to watch. The assignment of the new user to a particular classification group is determined based on the value of K which is part of the hyper-parameter tuning of the experiment 4.3 .
Below is the result achieved by the experiment without hyper-parameter tuning where a randomly selected K value is used. The figure 6 below depicts the KNN metrics without finding an optimal K for the dataset.

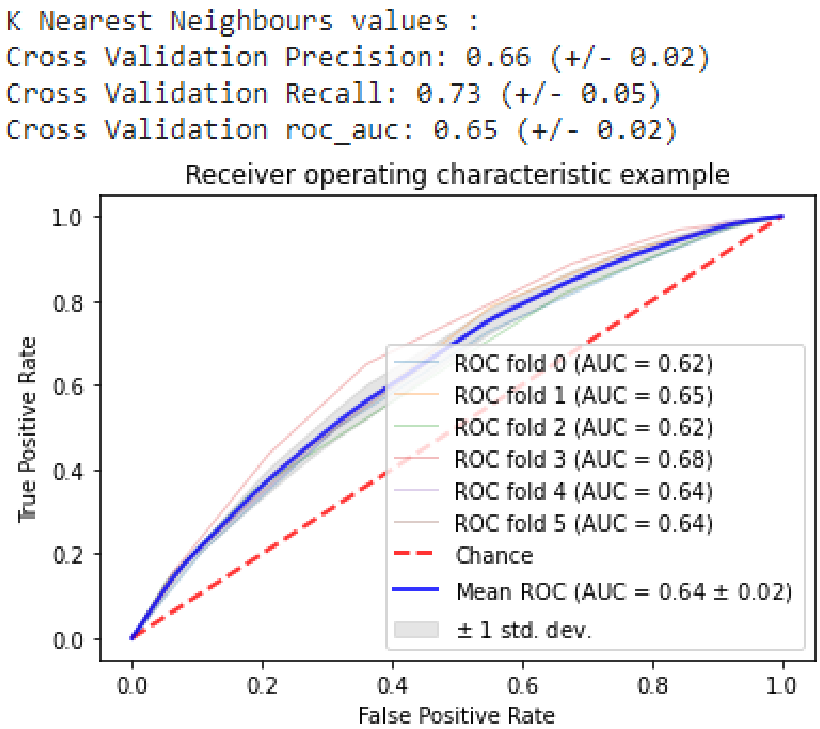


Figure 6: KNN before optimization

## 4.3 Experiment 3 : experiment conducted using KNN and hyper parameter tuning

This section optimizes the steps conducted in the section 4.2.As part of the feature engineering process we normalized the dataset used in the experiment using several nor-

malization techniques, of which MinMaxScaler seemed to give the most optimal result. After normalization of the dataset GridSearchCV algorithm was applied to accrue the optimal value for leaf_size, distance metric to be used and number of neighbours for voting to be used to assign a new member to a category. The leaf size of value determined using this technique was 28. Practically the larger the leaf size, closer the neighbours are to the data point.Other parameters which were tuned in order to assign the user to the right group were - leaf size which was determined to be 28, distance for similarity was determined to be Manhattan, algorithm was determined as brute, neighbours value was determined as 28.
Using these metrics we were able to match the metrics earlier achieved by the Random Forest experiment, with Gradient Booster improving on the recall score whereas KNN almost matching similar metrics as Random Forest.Below figure 7 shows those values.

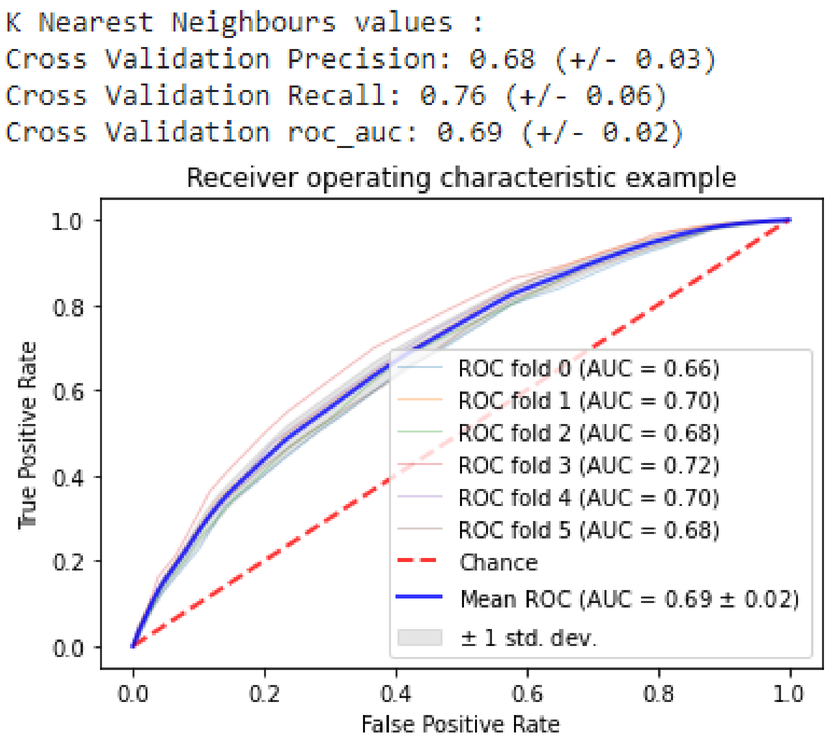


Figure 7: KNN optimized

## 4.4 Discussion

Below is a summarized table of the metrics achieved using the above experiments:

| Summary | | | | | |
|---|---|---|---|---|---|
| **Type** | **Algorithm** | **Precision** | **Recall** | **ROC-AUC** | **No. of attributes** |
| State-of-the-art | Random Forest | 0.68 (+/- 0.03) | 0.80 (+/- 0.07) | 0.70 (+/- 0.02) | 12 |
| Exp 1 | Gradient Booster | 0.64 (+/- 0.01) | 0.84 (+/- 0.04) | 0.67 (+/- 0.02) | 2 |
| Exp 2 | KNN | 0.66 (+/- 0.02) | 0.73 (+/- 0.05) | 0.65 (+/- 0.02) | 2 |
| Exp 3 | Tuned KNN | 0.68 (+/- 0.03) | 0.76 (+/- 0.06) | 0.69 (+/- 0.02) | 2 |

As part of the three experiments conducted using Gradient Booster and KNN algorithm to improve upon the state-of-the-art Random Forest experiment of J. Herce-Zelaya (2020) we could see that optimized KNN was able to improve the score of recall from 0.80 to 0.76 and was nearly able to match the score derived from random forest with an ROC-AUC score of 0.70 with a score of 0.69. As per the other aim of the experiment we were able to achieve these metrics with only 2 attributes age and gender, whereas the existing start-of-the-art experiment involves more than 10 attributes from twitter to improve the recommendation.

# 5 Conclusion and Future Work

From the discussion 4.4 section we can see that KNN perform better that Random Forest despite the lesser number of variables supplied to it. The dataset used for this experiment was not the same as used for the state-of-the-art experiment as the dataset is based upon the user's availability on the OSN with the same ratings, hence as a workaround we have used MovieLens dataset instead of twitter as source of profiles. We also conducted additional experiments using K-Means clustering, but as part of finding the best K-value for clustering, the resourcing power was not ample.Hence as part of the future work, if enough resources are available we should be able to improve this score using clustering methods similar to the steps taken for classification procedure of KNN algorithm. Besides this, RS is a dynamic system and current work for RS is based on emotion analysis of the user watching the movies. Hence, a factor which can be added for further research is that it can be add user-item-emotion matrix to the current experiment.

# References


Ahuja, R., Solanki, A. and Nayyar, A. (2019). Movie recommender system using k-means clustering and k-nearest neighbor, pp. 263–268.

Chen, J., Zhao, C., Chen, L. et al. (2020). Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering, *Complex & Intelligent Systems* **6**(1): 147–156.

Chen, V. X. and Tang, T. Y. (2019). Incorporating singular value decomposition in user-based collaborative filtering technique for a movie recommendation system: A comparative study, pp. 12–15.

Eklaspur, N. M. and Pashupatimath, A. S. (2015). A friend recommender system for social networks by life style extraction using probabilistic method-friendtome, *International Journal of Computer Science Trends and Technology (IJCST)* **3**(3).

J. Herce-Zelaya, C. Porcel, J. B.-M. A. T.-L. E. H.-V. (2020). New technique to alleviate the cold start problem in recommender systems using information from social media and random decision forests, *Information Sciences* **536**(1): 156–170.

Mokarrama, M. J., Khatun, S. and Arefin, M. S. (2020). A content-based recommender system for choosing universities, *Turkish Journal of Electrical Engineering & Computer Sciences* **28**(4): 2128–2142.

Natarajan, S., Vairavasundaram, S., Natarajan, S. and Gandomi, A. H. (2020). Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data, *Expert Systems with Applications* **149**: 113248.

Rahul, M., Kumar, V., Yadav, V. et al. (2021). Movie recommender system using single value decomposition and k-means clustering, **1022**(1): 012100.

Reddy, S., Nalluri, S., Kunisetti, S., Ashok, S. and Venkatesh, B. (2019). Content-based movie recommendation system using genre correlation, pp. 391–397.

Singh, R. H., Maurya, S., Tripathi, T., Narula, T. and Srivastav, G. (2020). Movie recommendation system using cosine similarity and knn, *International Journal of Engineering and Advanced Technology* **9**(5): 556–559.

TK, N., Surendiran, B. and Rajagopalan, M. M. R. D. N. (n.d.). Analysis of sub-clustering in group recommender system.

Wibowo, A. T. et al. (2020). Leveraging side information to anime recommender system using deep learning, pp. 62–67.

Zarzour, H., Al-Ayyoub, M., Jararweh, Y. et al. (2020). A convolutional neural network-based reviews classification method for explainable recommendations, pp. 1–5.