TABLE OF CONTENTS

ACKNOWLEDGEMENT

Hereby thanking my torchbearers with an immense pleasure who have helped us and stood with us through this project.

Our first and foremost thanking goes to our subject faculty Prof. Niarv Bhatt sir for giving us time to solve our problems and help us collecting the resources of our project.

Also our hearty gratitude to our Head of Department, Dr.Parth Shah for giving us encouragement and technical support on the project.

Akshay Pandya (16IT061)

Madhav Modi (16IT056)

ABSTRACT

Recommender systems being a part of information filtering system are used to forecast the bias or ratings the user tend to give for an item. Among different kinds of recommendation approaches, collaborative filtering technique has a very high popularity because of their effectiveness. These traditional collaborative filtering systems can even work very effectively and can produce standard recommendations, even for wide ranging problems. For item based on their neighbor's preferences Collaborative filtering techniques creates better suggestions than others. The engine will recommend movies to the users as per their interest as well as it will recommend movies rated by other users who are similar to the user.

CHAPTER-1

INTRODUCTION

1.1 Introduction

- A recommendation system is a type of information filtering system which attempts to predict the preferences of a user, and make suggests based on these preferences.
- Movie recommendation systems provide a mechanism to assist users in classifying users with similar interests.
- This makes recommender systems essentially a central part of websites and e-commerce applications.

1.2 Purpose of System

- The purpose of this project is to predict the preferences of a user,finds other users with similar interests and suggests movie based on their preferences and interests.

1.3 Scope of the System

- This project is based on recommendation system that recommends movies to users.
- This system calculates the similarities between different users and then recommend movie to them as per the ratings given by the different users of similar tastes.
- This will provide a precise recommendation to the user.

1.4 Objectives:

- Helps users to find movies of their interests.
- It also finds other users having similar interests.

CHAPTER- 2:

SYSTEM REQUIREMENTS

2.1 User characteristics:

- User must know the basics of python and algorithms and should know how to operate and run program.User will be shown movies and ratings of his similar users as a output.

2.2 Hardware and Software Requirement:

2.2.1 Software Requirement:

- Operating system : Windows XP or any higher Microsoft OS
- Tool: Spyder (Python 3.6)

2.2.2 Hardware Requirement:

- 1GHz Pentium processor or other compatible
- 512MB RAM

2.3 Assumptions and dependencies:

- We have assumed that user has the basic knowledge of python programming and some mathematical algorithms.

2.4. Documentation:

- Microsoft Word 2013for documentation.
- Power point 2013 for presentation

.

CHAPTER-3:

SYSTEM DESIGN

3.1 DESIGN METHODOLOGY

A software design is a meaningful engineering representation of a software product that is to be built. A design can be traced to the customer's requirement and can be assessed for quality against predefined criteria.

In order to evaluate the quality of a design (representation) the criteria for a good design should:

- Exhibit good architectural structure.
- Be modular.
- Contain distinct representation of data, architecture, interfaces and components (modules).
- Lead to components that exhibit independent functional characteristics.
- Lead to an interface that reduces that complexity of connection between modules and whit the external environment.
- Is derived using a reputable method that is drive by information obtained during software requirement analysis?
- To design this software we have used Object Oriented Design Method.

What was the reason to choose this method is as under:

- It provides the feature like reuse, quality, an emphasis on modeling the real world, resistance to change, encapsulation and abstraction etc.
- **Faster development:** this was the requirement of the project.
- **Reuse of previous work:** This was required in the project because it is going to extend further to meet the other requirement of the organization.
- **Increased quality:** this was the non-functional requirement of the organization.
- **Modular Architecture:** this was required to meet the modification in the product.
- **Better Mapping to the problem Domain.**

3.2 Project Flow:

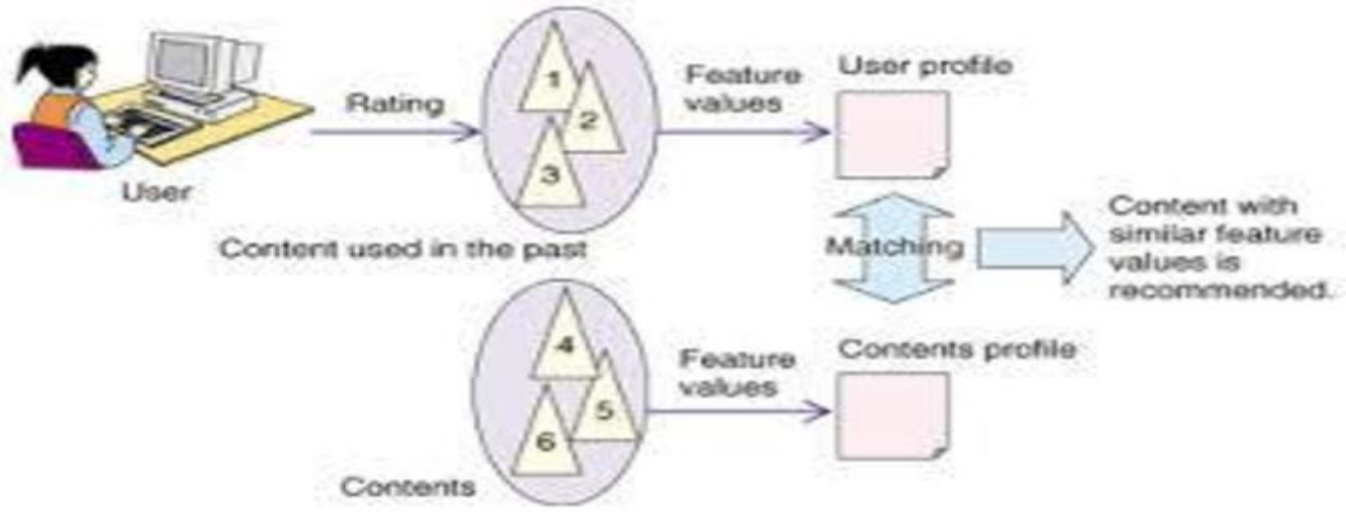

Fig. 3.1

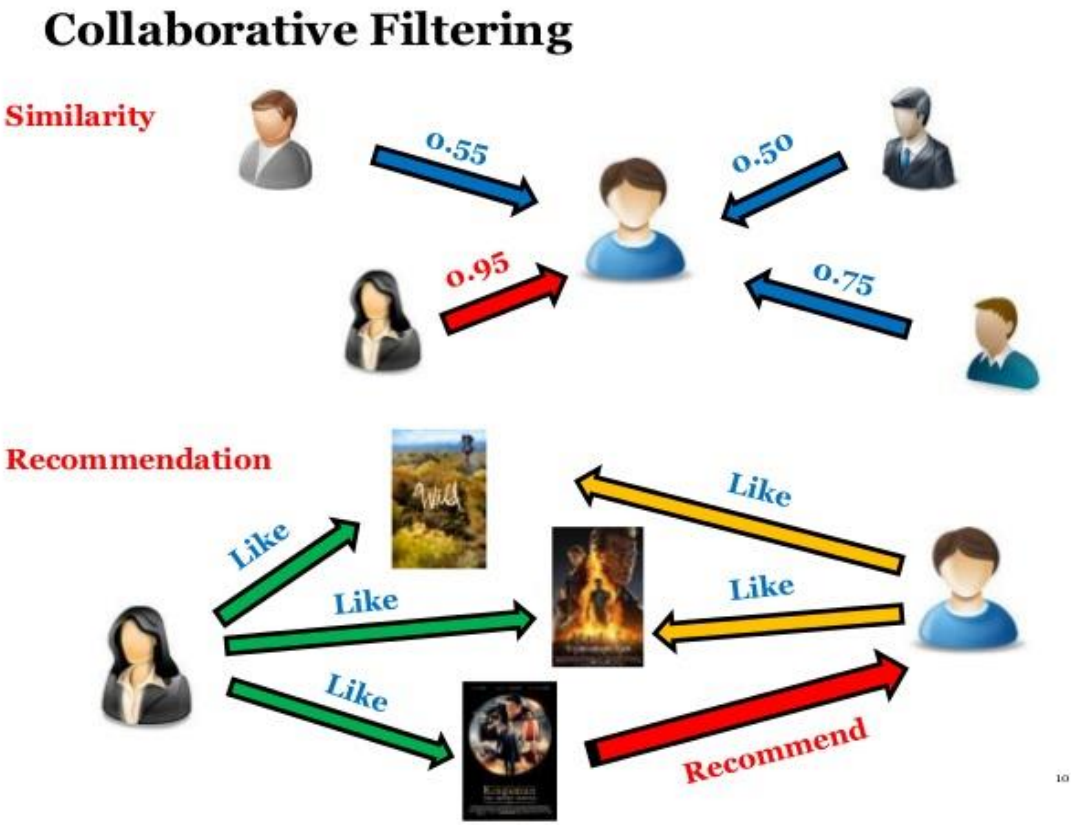

Fig. 3.2

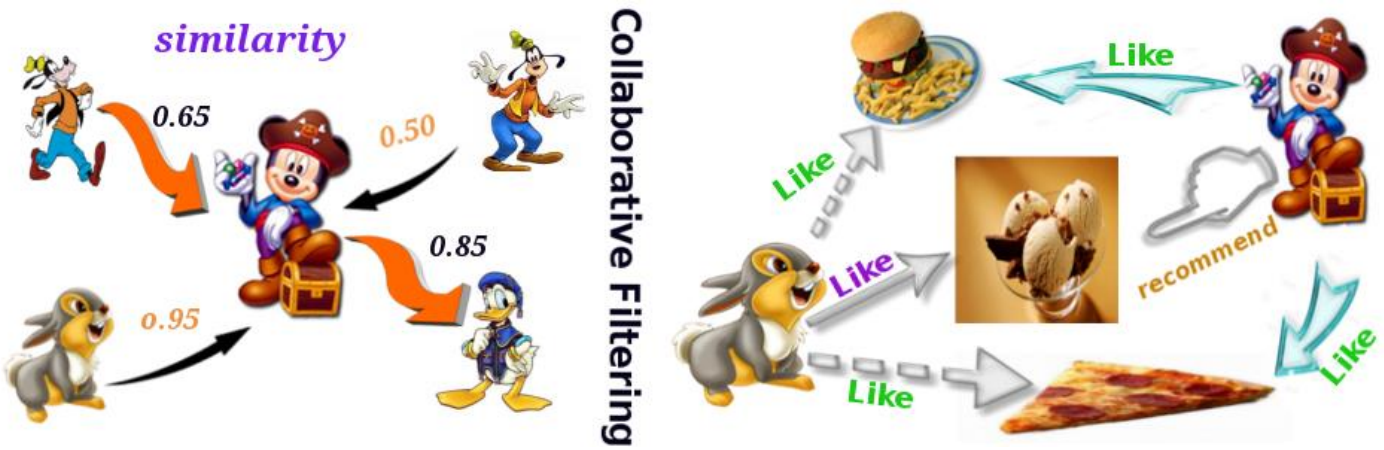
similarity
0.65
0.50
0.85
0.95
Collaborative Filtering
Like
Like
Like
Like
Like
recommend

Fig. 3.3

CHAPTER- 4:

IMPLEMENTATION PLANNING

4.1 Software Process Model:

1 In prototyping model initially the requirement gathering is done.

Develop and customer defines overall objectives; identify areas needing more requirement gathering.

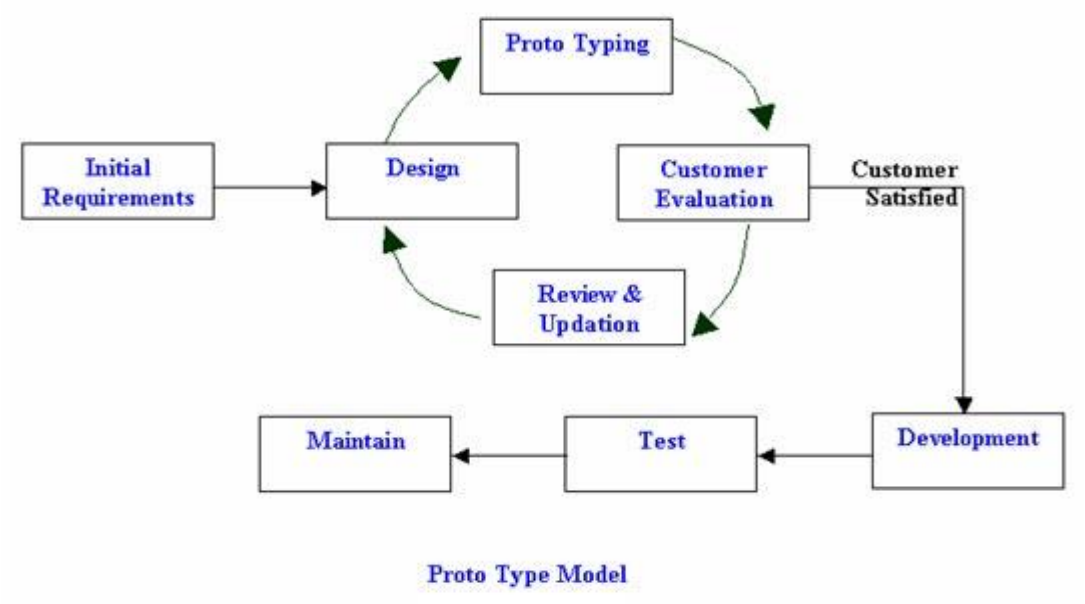

Fig. 4.1

2 Then a quick design is prepared. This design represents what will be visible to user in input and output format.

3 From the quick design a prototype is prepared. Customer or user evaluates the prototype in order to refine the requirements. Thus prototype is tuned for satisfying customer requirements. Thus prototype is important to identify the software requirements.

4 When working prototype is built, developer use existing program fragments or program generators to throw away the prototype and rebuild the system to high quality.

5 Certain classes of mathematical algorithms, subset of command driven systems and other application where results can be easily examined without real time interaction can be developed using prototyping paradigm.

4.1.1 Advantage of Prototyping

1. Software applications that are relatively easy to prototype almost always involve human-machine Interaction (HCL) the prototyping model is suggested.
2. A general objective of software is defined but not detailed input, processing or output requirements. Then in such a case prototyping model is useful.
3. When the developer is unsure of the efficiency of an algorithm or the adaptability of an system then prototype serves as a better choice.

4.2 Project planning

A **project plan** is "A formal, approved document used to guide both project execution and project control. The primary uses of the project plan are to document planning assumptions and decision, facilitate communication among stakeholders, and document approved scope, cost, and schedule baselines. A project plan may be summary or detailed."

"A statement of how and when a project's objectives are to be achieved, by showing the major products, milestone, activities and resource required on the project.

4.3 Implementation environment:

Python user and machine learning and data science aspirant**.**

4.4 Coding Standards

```
def user_reommendations(person):

    # Gets recommendations for a person by using a weighted average of every other user's rankings
    totals = {}
    simSums = {}
    rankings_list =[]
    for other in dataset:
        # don't compare me to myself
        if other == person:
            continue
        sim = pearson_correlation(person,other)
```

```
            #print ">>>>>>>",sim

            # ignore scores of zero or lower
            if sim <=0:
                continue
            for item in dataset[other]:

                # only score movies i haven't seen yet
                if item not in dataset[person] or dataset[person][item] == 0:

                # Similrity * score
                    totals.setdefault(item,0)
                    totals[item] += dataset[other][item]* sim
                    # sum of similarities
                    simSums.setdefault(item,0)
                    simSums[item]+= sim

            # Create the normalized list

        rankings = [(total/simSums[item],item) for item,total in totals.items()]
        rankings.sort()
        rankings.reverse()
        # returns the recommended items
        recommendataions_list = [recommend_item for score,recommend_item in rankings]
        return recommendataions_list


print("Recommendations for Madhav")
a=user_reommendations('Madhav')
print(a)

print("Recommendations for Akshay")
b=user_reommendations('Akshay')
print(b)
```

```
print("Recommendations for Parth")
b=user_reommendations('Parth')
print(b)
```

4.5 Snapshots of project

```
Euclidean distance score:
0.4721359549995794
Similarity between 2 person with Pearson Correltion is:
0.7470178808339965
Person who is most similar to a given person:
[(0.9912407071619299, 'Madhav'), (0.7470178808339965, 'Harsh'), (0.5940885257860044, 'Jeet')]
Recommendations for Madhav
['Gold', 'Raees', 'Sanju']
Recommendations for Akshay
[]
Recommendations for Parth
['Sanju', 'Bahubali']
```

Fig. 4.2

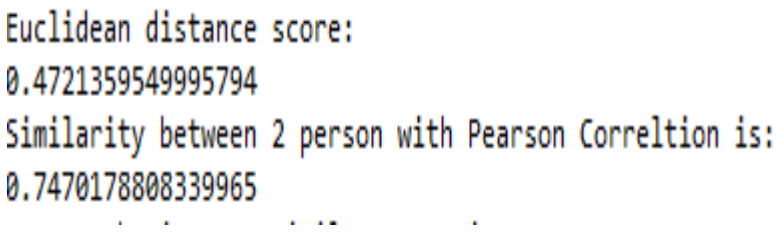

Fig. 4.3

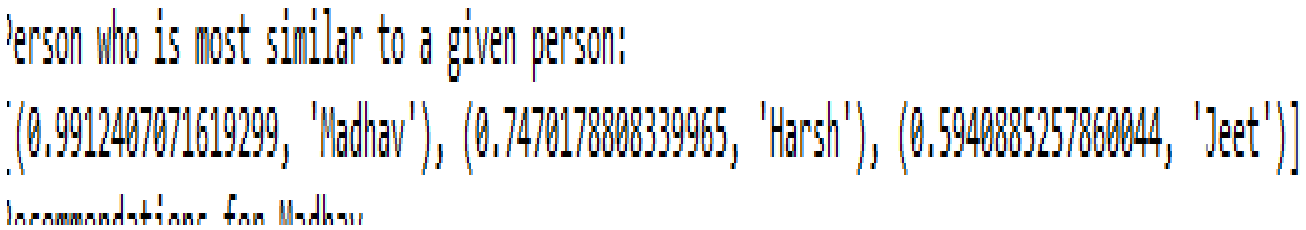

Fig. 4.4

CHAPTER-5

LIMITATIONS AND FUTURE ENHANCEMENT

5.1: Limitations

- This project has limitation that it is static project. The new user cannot come into the project during run time.
- Also more suitable algorithms like KNN algorithm is not used.
- It is based only on collaborative filtering.

5.2: Future Enhancements

- In future, we will add some new algorithms like KNN, clustering,movielens dynamic dataset that will give our project more enhancement.
- We will make it dynamic by taking run time users and adding new users and finding their similar users and recommend movies to them.
- We would also like to add the genres of the movie which will find the users liking similar category of movies.

Chapter 6

CONCLUSION

A recommendation system has been implemented based on hybrid approach of collaborative filtering engine and context based engine. We have tried to combine the existing algorithms for recommendation to come up with a hybrid one. It improves the performance by overcoming the drawbacks of traditional recommendation systems. We have used collaborative filtering which will recommend movies to the users as per their interest as well as it will recommend movies rated by other users who are similar to the user.

Chapter 7

REFERENCES & BIBLIOGRAPHY

BIBLOGRAPHY

- During the development of my project, we have taken help of many books, and websites for reference, which we would like to humbly mention in this section. We hereby sincerely express our gratitude to all authors, publishers, designers, software developers of these books and websites which we used as materials.
- These books, web sites acted as our tutor cum guide during the project Development.

REFERENCES:

- **www.analyticsvidhya.com**
- **www.datacamp.com**
- **www.pythonforbeginners.com**
- **Collective Intelligence book**
- **www.udemy.com**