

High Level Requirements Document

on

**Content Recommendation Engine**

(Movies/Shows/Video-game recommendation using data science and machine learning)

*(Submitted to* ***Jain (Deemed-to-be-University), Bengaluru*** *as a part of Project Centric Learning**for the partial fulfillment of the degree of* ***Bachelor of Computer Application)***

**BCA Semester - 2**

**Session 2021-22**

*Submitted by*

AMAN KUNDU [21BCAR0182] < DATA ANALYSIS>

FATHIMA FIRDOUSE [21BCAR0026] <GENERAL>

DARSHAN SHAJI [21BCAR0018] <GENERAL>

AJITH CHINNAPPA [21BCAR0097] <CTIS>

*Guided by*

Dr. M N Nachappa

*HEAD- SCHOOL OF CS & IT,*

*JAIN (DEEMED TO BE) UNIVERSITY, BANGALORE*

****

**Table of Contents**

**Introduction**

**Problem Statement**

**Project Objectives**

**Project Scope**

**Study**

**Software Requirements**

**S/W, H/W, Tools & Technology Requirements**

**Limitations of Existing System**

**Advantages of Proposed System**

**Economic Feasibility Assessment**

**References**

**Introduction**

Fast development of internet technology has resulted in explosive growth of available information over the last decade. Recommendation systems (RS), as one of the most successful information filtering applications, have become an efficient way to solve the information overload problem. The aim of Recommendation systems is to automatically generate suggested items (movies, TV shows, Web Series, Video Games) for users according to their historical preferences and save their searching time online by extracting useful data. Movie recommendation is the most widely used application coupled with online multimedia platforms which aims to help customers to access preferred movies intelligently from a huge movie library

**Problem Statement**

For building a recommender system from scratch, we face several different problems. Currently there are a lot of recommender systems based on the user information, so what should we do if the website has not gotten enough users. After that, we will solve the representation of a movie, which is how a system can understand a movie. That is the precondition for comparing similarity between two movies. Movie features such as genre, actor and director is a way that can categorize movies. But for each feature of the movie, there should be different weight for them and each of them plays a different role for recommendation.

Project Objectives

To make a system to curate and recommend content based on collaborative-filtering, content Based Recommendation and user profiling.

To collect segregate data relevant information in a database, to make recommendation system on the listed methods and parameters

To make better predictions than competing/existing system

**Project Scope**

To collect segregate data relevant information in a database, to make recommendation system on the listed methods and parameters

To make better predictions than competing/existing systems

Incorporate user profiling and make an intuitive GUI application

Implement a web scraping program to add tags and information to our dataset.

**Study/Literature Review report**

**Data Science**

Data science is the domain of study that deals with vast volumes of data using modern tools and techniques to find unseen patterns, derive meaningful information, and make business decisions. Data science uses complex machine learning algorithms to build predictive models.

The data used for analysis can come from many different sources and presented in various formats.

Now that you know what data science is, next up let us focus on the data science lifecycle. Data science’s life cycle consists of five distinct stages, each with its own tasks:

1. Capture: Data Acquisition, Data Entry, Signal Reception, Data Extraction. This stage involves gathering raw structured and unstructured data.
2. Maintain: Data Warehousing, Data Cleansing, Data Staging, Data Processing, Data Architecture. This stage covers taking the raw data and putting it in a form that can be used.
3. Process: Data Mining, Clustering/Classification, Data Modeling, Data Summarization. Data scientists take the prepared data and examine its patterns, ranges, and biases to determine how useful it will be in predictive analysis.
4. Analyze: Exploratory/Confirmatory, Predictive Analysis, Regression, Text Mining, Qualitative Analysis. Here is the real meat of the lifecycle. This stage involves performing the various analyses on the data.
5. Communicate: Data Reporting, Data Visualization, Business Intelligence, Decision Making. In this final step, analysts prepare the analyses in easily readable forms such as charts, graphs, and reports.

### Modeling

Mathematical models enable you to make quick calculations and predictions based on what you already know about the data. Modeling is also a part of Machine Learning and involves identifying which algorithm is the most suitable to solve a given problem and how to train these models.

### Statistics

Statistics are at the core of data science. A sturdy handle on statistics can help you extract more intelligence and obtain more meaningful results.

### Programming

Some level of programming is required to execute a successful data science project. The most common programming languages are Python, and R. Python is especially popular because it’s easy to learn, and it supports multiple libraries for data science and ML.

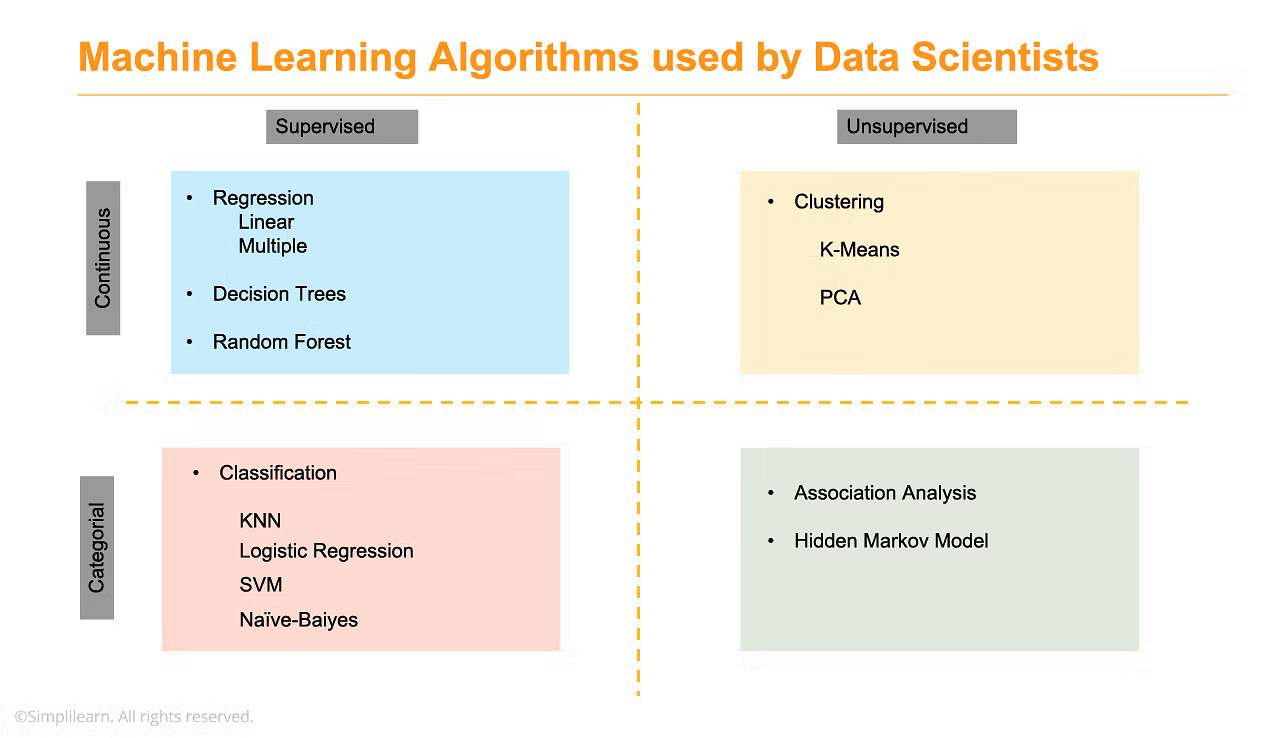
### Databases

A capable data scientist needs to understand how databases work, how to manage them, and how to extract data from them.

## Data Science Use Cases

Here are some brief overviews of a couple of use cases, showing data science’s versatility.

* Law Enforcement: In this scenario, data science is used to help police in Belgium to better understand where and when to deploy personnel to prevent crime. With only limited resources and a large area to cover data science used dashboards and reports to increase the officers’ situational awareness, allowing a police force that’s spread thin to maintain order and anticipate criminal activity.
* Pandemic Fighting: The state of Rhode Island wanted to reopen schools, but was naturally cautious, considering the ongoing COVID-19 pandemic. The state used data science to expedite case investigations and contact tracing, enabling a small staff to handle an overwhelming number of concerned calls from citizens. This information helped the state set up a call center and coordinate preventative measures.
* Driverless Vehicles: Lumewave, a sensor manufacturing company, was looking for a way to make sensor technology more cost-effective and accurate. They turned to data science and machine learning to train their sensors to be safer and more reliable, as well as using data to improve their 3D-printed sensor manufacturing process.

**Machine Learning Techniques**

### What’s the difference between data science, artificial intelligence, and machine learning?

Artificial Intelligence makes a computer act/think like a human. Data science is an AI subset that deals with data methods, scientific analysis, and statistics, all used to gain insight and meaning from data. Machine learning is a subset of AI that teaches computers to learn things from provided data.

**R studio**

RStudio is an integrated development environment for R, a programming language for statistical computing and graphics. It is available in two formats: RStudio Desktop is a regular desktop application while RStudio Server runs on a remote server and allows accessing RStudio using a web browser.

R is not a programming language like C or Java. It was not created by software engineers for software development. Instead, it was developed by statisticians as an interactive environment for data analysis. You can read the full history in the paper A Brief History of S[5](https://rafalab.github.io/dsbook/getting-started.html#fn5). The interactivity is an indispensable feature in data science because, as you will soon learn, the ability to quickly explore data is a necessity for success in this field. However, like in other programming languages, you can save your work as scripts that can be easily executed at any moment. These scripts serve as a record of the analysis you performed, a key feature that facilitates reproducible work.

1. R is free and open source.
2. It runs on all major platforms: Windows, Mac Os, UNIX/Linux.
3. Scripts and data objects can be shared seamlessly across platforms.
4. There is a large, growing, and active community of R users and, as a result, there are numerous resources for learning and asking questions.
5. It is easy for others to contribute add-ons which enables developers to share software implementations of new data science methodologies

**MACHINE LEARNING, PYTHON**

It is the field of study that gives computers the capability to learn without being explicitly programmed.

The ability to learn.

**INTRODUCTION:**

The term Machine Learning was coined by Arthur Samuel in 1959, an American pioneer in the field of computer gaming and artificial intelligence, and stated that “it gives computers the ability to learn without being explicitly programmed”.

And in 1997, Tom Mitchell gave a “well-posed” mathematical and relational definition that “A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

Within the field of data analytics, machine learning is used to devise complex models and algorithms that lend themselves to prediction; in commercial use, this is known as predictive analytics. These analytical models allow researchers, data scientists, engineers, and analysts to “produce reliable, repeatable decisions and results” and uncover “hidden insights” through learning from historical relationships and trends in the data set(input).

Suppose that you decide to check out a movie. You browse through the a movie website and search for a movie. When you look at a specific movie, just below the movie description there is a section titled “You might also like these movies”. This is a common use case of Machine Learning called “Recommendation Engine”. Again, many data points were used to train a model in order to predict what will be the best movies to show you under that section, based on a lot of information they already know about you.

**TERMINOLOGIES:**

Model - A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called hypothesis.

Feature - A feature is an individual measurable property of our data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.

Note: Choosing informative, discriminating and independent features is a crucial step for effective algorithms. We generally employ a feature extractor to extract the relevant features from the raw data.

Target (Label) - A target variable or label is the value to be predicted by our model. For the fruit example discussed in the features section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.

Training - The idea is to give a set of inputs(features) and it’s expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.

Prediction - Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

**CLASSIFICATION:**

Machine learning implementations are classified into three major categories, depending on the nature of the learning “signal” or “response” available to a learning system which is as follows:-

• Supervised learning: When an algorithm learns from example data and associated target responses that can consist of numeric values or string labels, such as classes or tags, in order to later predict the correct response when posed with new examples comes under the category of Supervised learning.

• Unsupervised learning: Whereas when an algorithm learns from plain examples without any associated response, leaving to the algorithm to determine the data patterns on its own. This type of algorithm tends to restructure the data into something else, such as new features that may represent a class or a new series of un-correlated values.

• Reinforcement learning: When you present the algorithm with examples that lack labels, as in unsupervised learning. However, you can accompany an example with positive or negative feedback according to the solution the algorithm proposes comes under the category of Reinforcement learning, which is connected to applications for which the algorithm must make decisions (so the product is prescriptive, not just descriptive, as in unsupervised learning), and the decisions bear consequences. In the human world, it is just like learning by trial and error.

• Semi-supervised learning: where an incomplete training signal is given: a training set with some (often many) of the target outputs missing. There is a special case of this principle known as Transduction where the entire set of problem instances is known at learning time, except that part of the targets are missing.

Categorizing on the basis of required Output:

Another categorization of machine learning tasks arises when one considers the desired output of a machine-learned system:

• Classification: When inputs are divided into two or more classes, and the learner must produce a model that assigns unseen inputs to one or more (multi-label classification) of these classes. This is typically tackled in a supervised way. Spam filtering is an example of classification, where the inputs are email (or other) messages and the classes are “spam” and “not spam”.

• Regression: Which is also a supervised problem, A case when the outputs are continuous rather than discrete.

• Clustering: When a set of inputs is to be divided into groups. Unlike in classification, the groups are not known beforehand, making this typically an unsupervised task.

**Database:**

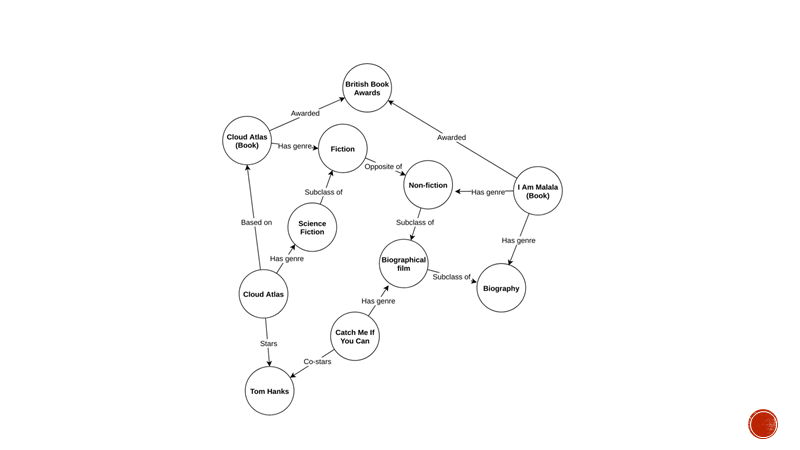
Why do we need Database for recommendation system:

For industrial applications, the database can hold hundreds of millions of users and movies, and billions of rating records, which means the rating matrix A, feature matrices U and V, plus other intermediate variables can consume terabytes of memory during model training. Such a challenge can be resolved by training the latent features in a graph database where the rating graph can be distributed among a multi-node cluster and partially stored on disk. Moreover, graph-structured user data (i.e. rating records) is stored in the database management system in the first place. In-database model training also avoids exporting the graph data from the DBMS to other machine learning platforms and thus better support continuous model update over evolving training data.

**Graph Database**

For industrial applications, the database can hold hundreds of millions of users and movies, and billions of rating records, which means the rating matrix A, feature matrices U and V, plus other intermediate variables can consume terabytes of memory during model training. Such a challenge can be resolved by training the latent features in a graph database where the rating graph can be distributed among a multi-node cluster and partially stored on disk. Moreover, graph-structured user data (i.e. rating records) is stored in the database management system in the first place. In-database model training also avoids exporting the graph data from the DBMS to other machine learning platforms and thus better support continuous model update over evolving training data.

Since relationships are made explicit by the edge elements, traversing the graph from one vertex to another is both conceptually simple and computationally inexpensive.  
As a result, you can perform relationship-based queries in real time, and quickly query customers’ past purchases, as well as instantly capture any new interests shown in their current online visit, both of which are essential for making real-time recommendations.



The above is an example knowledge graph representing movies and books as well as actors, genres and the complex interrelationships among them. In a knowledge graph, not only do we know what items are related to what properties, we know how they are related and impose no restrictions on what can be related and how.

For example, if a user likes “Cloud Atlas” (the movie), they might like “Catch Me If You Can” because Tom Hanks stars in both of them. On the other hand, they could be looking for something different from fiction. If they’re looking for a book to buy, they might like “Cloud Atlas” (the book), and if they also liked “Catch Me If You Can”, maybe they would like the “I Am Malala” book as it is also a biography and won awards similar to the Cloud Atlas book.

While modeling this with standard SQL technologies is definitely possible, it is usually very difficult because of the rich structure. Instead, in a graph database, modeling such a structure is more straightforward. Also, querying a lot of relationships in an SQL database like this is not exactly a very efficient operation. What’s more is that in a graph database, we are free to extend the structure of our database graph as we’d like and to represent an ever-evolving domain.

MySQL

A database which is used to store the collection of records in an organized form. It allows us to hold the data into tables, rows, columns, and indexes to find the relevant information frequently. We can access and manage the records through the database very easily.

implements a database as a directory that stores all files in the form of a table. It allows us to create a database mainly in two ways:

MySQL Command Line Client

MySQL Workbench

**S/W, H/W, Tools & Technology Requirements**

1. Data science R studio

2. Machine Learning Python

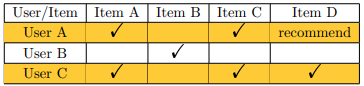
3. Data Collection Process/ Database MySQL

4. Web App/UI React JS Workbox

**Implementation of System**

**Method 1 :User-based collaborative-filtering**

In user-based collaborative filtering, it is considered that a user will like the items that are liked by users with whom they have similar taste. So the first step of user-based collaborative-filtering is to find users with similar taste. In collaborative filtering, the users are considered similar when they like similar items.



**Advantages**

No need for professional knowledge;

Performance improving as the increasing of the user number;

Automatic; Easy to find user

Complex unstructured item can be processed. eg. Music, Video, etc.

**Disadvantages :**

Sparsity problem;

Poor scalability;

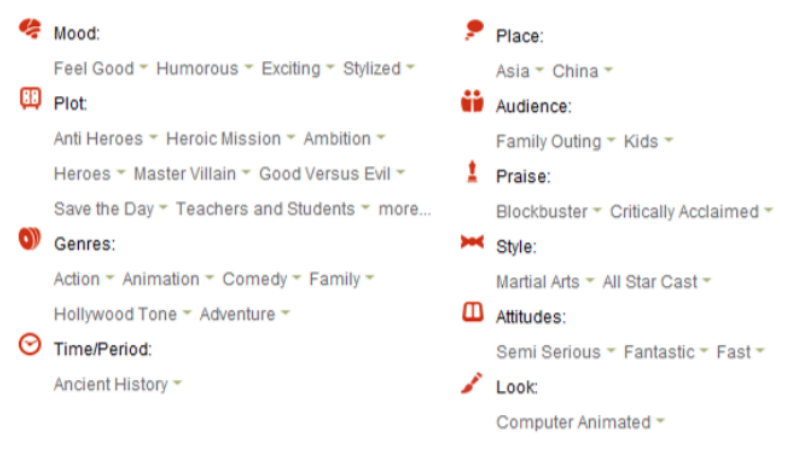
New user and new item problem;

The recommendation quality is determined by users

**Method 2: Content Based Recommendation**

Content-based recommendation is an important approach in recommender systems. The basic idea is to recommend items that are similar with what user liked before. The core mission of content-based recommender system is to calculate the similarity between items and the tastes of a user can be modeled according to the history of what the user liked.

Using keywords to model item is an important step for many recommender systems. But extracting keywords of an item is also a difficult problem, especially in media field, because it is very hard to extract text keywords from a video. For solving this kind of problem, there are two main ways. One is letting developers tag the items and another one is letting users tag them. We also plan to implement data scraping program for the same.



**Advantages;**

Recommendation results are intuitive and easy to interpret

Not dependent on amount of users, it works on historical data

No new item problem and sparsity problem

Supported by mature technologie of classification learning

**Disadvantages;**

Limited by the features extraction methods

New user problem

The training of classifier needs massive data

Poor scalability

**Pre-Requirement : Segregated Database on movies**

Data will scraped in csv format with optional parameters stored in SQL database implemented using MySQL

Certain streaming services provide API access to their catalog but it is currently out of scope

Note that not all movies have complete features, some of the movies may miss some feature information, and this fact has been considered in the algorithm

Open Source datasets of existing movies :

MovieLens 25M dataset,

MovieLens Tag Gnome Dataset,

IMDB titles.ratings.tsv.gz dataset

Manual Data Entry

Web Scraping Program

**Method 3 : User Profiling - Behavioral analytics**

A user profile is a collection of settings and information associated with a user. It contains critical information that is used to identify an individual, such as their name, age, demography, individual characteristics and any special preferences.

Every user on signing up will be asked to enter relevant information based on the above information

This information will be used to create a user profile based on which content will be recommended

The user will be prompted to log their consumption manually and after a certain amount of data is collected we can combine the user profile and their behaviour to improve recommendation.

**Advantages**;

Does not depend on a huge number of users i.e can also function with single user

This approach is very easy to implement

Most of the data is given upfront by the user itself

Scalable, gets better with more data & time.

**Disadvantages;**

The users need to be active and produce meaningful logs

The predictions get better with time, the user needs to spend more time to get results

**Web scraping program**

Web Scraping is the process of gathering useful information from the web and making meaningful insights from it. In a way, web scraping is automating the process of data collection.

Our scraping program will look for parameters such as :

Mood

Plot

Genre

Sub-Genre

Time-Period

Setting

Audience

Awards

Style

Attitudes

Look

**Additional Parameter : Emotional Niche & External Influences**

Many users have preferences which do not fit their profile and or is not recognizable by any algorithm simply due to the unpredictable nature of human psychology

Suppose a person gets influenced in real life by his social peers, an incident in the past or by unpredictable factor which cannot be accounted by the program

These exceptions will be collected and stored for human review, if there is a recurring pattern which can help improve user profiling

This whole process has to be implemented manually.

**Additional Parameter : Negative user feedback (high weightage)**

Most avid consumers watch movies for time-pass or entertainment and have average content engagement

For most users if they don't like a movie then they stop watching or if they implicitly give a negative review, this helps the system to get rid of irrelevant recommendation, this information is reliable as there the least inherent biases and this behaviour does not change

This will be directly implemented in the algorithm

**Additional parameter: Calculating Popularity Index (Factor)**

A piece of content could be a user preference because of a internet fad or general trending factor. These preferences would not be recognised with CBR, CF or User Profiling but would heavily influence the user choice.

So we will implement a separate algorithm based on “popularity” of a movie using python to calculate the popularity index based on factors like filtered web results, citations and number of social media posts, all in a fixed time frame.

Implementation through Web-App

A web applications runs on a users web browser over an active internet connection

This system will be implemented using a intuitive GUI web-app

The user profiling and logging will go through this

The final results will be given to the user

It is implemented using JavaScript, using libraries like WorkBox and React

**References**

**[1] Recommendation System Using Python - Academic Paper Tushar Kholia**

**[2]Content-based Recommender - EXAMENSARBETE INOM INFORMATIONS- OCH KOMMUNIKATIONSTEKNIK**

**[3]Improved Collaborative Movie Recommendation System-Zan Wang Department of Software Engineering, School of Computer Software, Tianjin University**

**[4]Expert Systems with Applications - www.elsevier.com/locate/eswa**

**[5]Electronic Letters on Computer Vision and Image Analysis 19(3):18-37, 2020 - Computer Engineering, Government Engineering College, Modasa, Aravalli, India**

**[6]International Journal of Research in Industrial Engineering www.riejournal.com - Int. J. Res. Ind. Eng. Vol. 9, No. 1 (2020) 84–98**

**[7]Chapter 1 Getting started with R and RStudio | Introduction to Data Science**

**[8]https://realpython.com/beautiful-soup-web-scraper-python/**

**[9]https://imerit.net/blog/13-best-movie-data-sets-for-machine-learning-projects-all-pbm/**

**[10]https://grouplens.org/datasets/movielens/**

**[11]https://datasets.imdbws.com/**