

UTILIZING HYBRID FILTERING TO RECOMMEND A PERSONALIZED GIFT

ODIERO ROSANNE OGWEL

101192

An Information Systems Project proposal Documentation submitted to the Faculty of Information and Technology in partial fulfilment of the requirements for the award of Bachelor of Science in Informatics and Computer Science.

20-June-2020

# Declaration and Approval

I, 101192, declare that this project proposal has not been submitted to any other University for the award of a Degree in Bachelor of Science in Informatics and Computer Science or any other degree.

Student Number: 101192

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Supervisor Name: Stephen Obonyo

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# ABSTRACT

The process of giving gifts is an important practice and a universal social norm of human culture. There is some message conveyed by the type of gift given yet choosing the ideal gift to give is problematic as the receivers have a variety of gift preferences that the sender might not be aware of. Some attempts have been made to recommend gifts but have been largely directed to “collaborative filtering” i.e. making generalized gift recommendations based on purchasing patterns of many users. These attempts however have the shortcoming of not truly personalizing the gift selection.

The proposed solution is to design a hybrid recommendation system i.e. a system that is characterized by content-based filtering, collaborative filtering, and knowledge-based filtering, which will generate gift recommendations for a specific user based on the receivers’ interests. User profiles will be generated for users based on their interests and demographic information and the system will monitor the products the users search for, view, like, add to cart and purchase. This data will later be used when the user would like to purchase a gift for someone by providing information about the recipient and the system will recommend a gift that a person with the similar user profile would like. This system will reduce the gifting anxiety givers have when picking out an ideal gift and the amount energy and time consumed when finding a gift will also reduce.

**Keywords**: hybrid recommendation, knowledge-based filtering, content-based recommendation, collaborative filtering.

Contents

[Declaration and Approval ii](#_Toc63732875)

[ABSTRACT iii](#_Toc63732876)

[Chapter 1 : Introduction 1](#_Toc63732877)

[1.1 Background 1](#_Toc63732878)

[1.2 Problem Statement 3](#_Toc63732879)

[1.3 Objectives 3](#_Toc63732880)

[1.3.1 General Objectives 3](#_Toc63732881)

[1.3.2 Specific Objectives 3](#_Toc63732882)

[1.4 Research Questions 3](#_Toc63732883)

[1.5 Justification 4](#_Toc63732884)

[1.6 Scope and Limitations 4](#_Toc63732885)

[Chapter 2 : Literature Review 5](#_Toc63732886)

[2.1 Introduction 5](#_Toc63732887)

[2.2 Current Methods People Use to Select Gifts and The Challenges They Face. 5](#_Toc63732888)

[2.3 Recommendation Systems 5](#_Toc63732889)

[2.3.1 Phases of recommendation process 6](#_Toc63732890)

[2.3.1.1 Information Collection Phase 6](#_Toc63732891)

[2.3.1.2 Learning Phase 7](#_Toc63732892)

[2.3.1.3 Prediction/recommendation phase 7](#_Toc63732893)

[2.3.2 Types of Recommendation Systems 7](#_Toc63732894)

[2.3.2.1 Content-based Filtering 7](#_Toc63732895)

[2.3.2.2 Collaborative Filtering 9](#_Toc63732896)

[2.3.2.3 Hybrid Recommendation systems 10](#_Toc63732897)

[2.3.2.3.2 Techniques for Hybrid Approach 11](#_Toc63732898)

[2.3.2.4 Knowledge based filtering 14](#_Toc63732899)

[2.3.3 Evaluation Metric for recommendation systems 14](#_Toc63732900)

[2.3.3.1 Statistical Accuracy Metrics 14](#_Toc63732901)

[2.3.3.2 Decision support accuracy metrics 15](#_Toc63732902)

[2.4 Related Works 15](#_Toc63732903)

[2.4.1 Amazon 16](#_Toc63732904)

[2.4.2 Netflix 16](#_Toc63732905)

[2.4.3 YouTube 17](#_Toc63732906)

[2.5 Review of methods used 18](#_Toc63732907)

[2.6 Conceptual Framework of the proposed solution 19](#_Toc63732908)

[Chapter 3 : Research Methodology 21](#_Toc63732909)

[3.1 Introduction 21](#_Toc63732910)

[3.2 Research Design 21](#_Toc63732911)

[3.3 Model Development 21](#_Toc63732912)

[3.3.1 Data Collection 21](#_Toc63732913)

[3.3.2 Data Pre-processing 21](#_Toc63732914)

[3.3.3 Model Training and Testing 22](#_Toc63732915)

[3.4 Software Development Methodology 23](#_Toc63732916)

[3.4.1 Requirements Planning 24](#_Toc63732917)

[3.4.2 User Design 24](#_Toc63732918)

[3.4.3 Rapid construction and feedback gathering 25](#_Toc63732919)

[3.4.4 Finalise product / implementation. 25](#_Toc63732920)

[3.5 System Deliverables 25](#_Toc63732921)

[3.5.1 Recommendation Engine 25](#_Toc63732922)

[3.5.2 User Module 25](#_Toc63732923)

[3.5.3 Admin Module 26](#_Toc63732924)

[3.5.4 System documentation 26](#_Toc63732925)

[3.6 System Development Tools and Techniques 26](#_Toc63732926)

[3.6.1 Visual Studio Code 26](#_Toc63732927)

[3.6.2 Draw.io 26](#_Toc63732928)

[3.6.3 Python Programming Language (Version 3.6) 26](#_Toc63732929)

[3.6.4 Django Python Web Framework 26](#_Toc63732930)

[3.6.5 Github 26](#_Toc63732931)

[3.6.6 Bootstrap 26](#_Toc63732932)

[3.7 Ethical Considerations 27](#_Toc63732933)

[Chapter 4 : System Analysis and Design 28](#_Toc63732934)

[4.1 Introduction 28](#_Toc63732935)

[4.2 System Requirements Analysis 28](#_Toc63732936)

[4.2.1 Functional Requirements 28](#_Toc63732937)

[4.2.2 Non-Functional Requirements 28](#_Toc63732938)

[4.3 System Narrative 29](#_Toc63732939)

[4.4 System Analysis Diagrams 30](#_Toc63732940)

[4.4.1 Context Diagram 30](#_Toc63732941)

[4.4.2 Data Flow Diagram 30](#_Toc63732942)

[4.4.3 Activity Diagram 31](#_Toc63732943)

[4.5 System Design Diagrams 33](#_Toc63732944)

[4.5.1 Entity Relationship Diagram 33](#_Toc63732945)

[4.5.2 Database Schema 33](#_Toc63732946)

[Chapter 5 :System Implementation and Testing 34](#_Toc63732947)

[5.1 Introduction 34](#_Toc63732948)

[5.2 Description of the dataset 34](#_Toc63732949)

[5.2.1 Data Pre-processing and Feature extraction 35](#_Toc63732950)

[5.2.2 Creating a Training and Validation Set 36](#_Toc63732951)

[5.3 Model Training and Testing 37](#_Toc63732952)

[5.3.1 Imported Libraries 37](#_Toc63732953)

[5.3.2 Implementing Alternating Least Squares(ALS) for Implicit Feedback 37](#_Toc63732954)

[5.3.3 Evaluating the Recommender System 38](#_Toc63732955)

[5.4 System Testing 39](#_Toc63732956)

[5.4.1 Unit Testing 39](#_Toc63732957)

[5.4.2 Performance Testing 40](#_Toc63732958)

[5.4.3 Integration Testing 40](#_Toc63732959)

[5.4.4 Data Flow Testing 40](#_Toc63732960)

[5.4.5 Black-Box Testing 40](#_Toc63732961)

[5.4.5.2 Sample standard login Test case 41](#_Toc63732962)

[5.4.3 White-Box Testing 42](#_Toc63732963)

[Chapter 6 : Discussions 43](#_Toc63732964)

[6.1 Introduction 43](#_Toc63732965)

[6.2 Achievement of Objectives 43](#_Toc63732966)

[6.2.1 Current method people use to select a gift for their loved ones. 43](#_Toc63732967)

[6.2.2 Challenges that people undergo when choosing a gift. 43](#_Toc63732968)

[6.2.3 Types of recommendation systems and how they work. 43](#_Toc63732969)

[6.2.4 Existing gift selection applications and their short comings. 43](#_Toc63732970)

[6.2.5 Build and Test the proposed solution 43](#_Toc63732971)

[6.3 Discussion of the results of testing 44](#_Toc63732972)

[6.3.1 Explanation of the test result 44](#_Toc63732973)

[Chapter 7 : Conclusion, Recommendation and Future Works 46](#_Toc63732974)

[7.1 Conclusion 46](#_Toc63732975)

[7.2 Challenges faced 46](#_Toc63732976)

[7.3 Recommendations 46](#_Toc63732977)

[7.4 Future Works 46](#_Toc63732978)

[Chapter 8 : References 47](#_Toc63732979)

[Appendix 50](#_Toc63732980)

[Gantt Chart 50](#_Toc63732981)

Table of Figures

[Figure 2.1:Content-based recommendation system 8](#_Toc63732982)

[Figure 2.2: Memory-based Collaborative filtering 9](#_Toc63732983)

[Figure 2.3: Weighted hybridization 11](#_Toc63732984)

[Figure 2.4:Switching hybridization 12](#_Toc63732985)

[Figure 2.5:Mixed hybridization 12](#_Toc63732986)

[Figure 2.6:Feature Combination Hybrid 13](#_Toc63732987)

[Figure 2.7: Cascade hybridization (a) 13](#_Toc63732988)

[Figure 2.8: Cascade hybridization (b) 13](#_Toc63732989)

[Figure 2.9: A screen shot of one of amazon users homepage 16](#_Toc63732990)

[Figure 2.10: Screenshot of User interface of Netflix with movie recommendations. 17](#_Toc63732991)

[Figure 2.11: Screenshot of YouTube homepage with video recommendations 18](#_Toc63732992)

[Figure 2.12:Conceptual Framework 20](#_Toc63732993)

[Figure 3.1: Matrix Factorization 22](#_Toc63732994)

[Figure 3.2: Rapid Application Development Diagram 23](#_Toc63732995)

[Figure 4.1:Context Diagram 30](#_Toc63732996)

[Figure 4.2:Data Flow Diagram 31](#_Toc63732997)

[Figure 4.3:Activity Diagram 32](#_Toc63732998)

[Figure 4.4:ERD Diagram 33](#_Toc63732999)

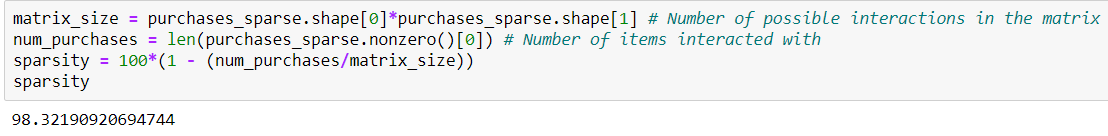
[Figure 4.5:Database Schema 33](#_Toc63733000)

[Figure 5.1: Description of the dataset 34](#_Toc63733001)

[Figure 5.2: Sample Dataset 35](#_Toc63733002)

[Figure 5.3:Resulting matrix of grouped purchases 35](#_Toc63733003)

[Figure 5.4:Sparse Matrix 35](#_Toc63733004)

[Figure 5.5:sparsity of the matrix 36](#_Toc63733005)

[Figure 5.6: Test and Training creation visualization 36](#_Toc63733006)

[Figure 5.7: Masking for creating test dataset 36](#_Toc63733007)

[Figure 5.8: Code for generating train and test data 37](#_Toc63733008)

[Figure 5.9:Alternating Least Squares 38](#_Toc63733009)

[Figure 5.10:Ratings for a particular user 38](#_Toc63733010)

[Figure 5.11: Mean AUC for model evaluation 39](#_Toc63733011)

[Figure 5.12:Sample recommendation test case 41](#_Toc63733012)

[Figure 5.13 Sample Standard Login Test Case 42](#_Toc63733013)

[Figure 6.1: Purchased Item by user 12346 44](#_Toc63733014)

[Figure 6.2:Recommended Product for user 12346 44](#_Toc63733015)

# : Introduction

## 1.1 Background

Gift giving is a universally recognised tradition that dates back to the beginning of written history. This is the process of exchanging gifts between a giver and a receiver. Humans are social species and one of our distinguishing characteristics is creating and sustaining relationships with others. This gives us a sense of purpose and a feeling of satisfaction. Exchanging of gifts to symbolize and celebrate key life events, spiritual history, and family relationships is a good opportunity to strengthen those relationships (CNN, n.d.). Gifting is not only a social lubricant but also an act of symbolic communication of our feelings such as love, joy, appreciation, or gratitude for another person (“The Meaning of Gift Giving,” 2015).

In modern society the practice of gift-giving often commences with prenatal "baby shower" gifts and continues during and after a person’s lifetime. Gifts of money and flowers are given to memorialize the dead and extend condolences to the bereaved. During an individual’s lifetime, the array of occasions which gift-giving is ritualized includes graduations, birthdays, anniversaries, weddings, Christmas, Valentine's Day, engagements, Father's Day, Mother's Day and others (Belk, n.d.).

Research and experiences have shown that, the process of selecting the ideal gift is one the most stressful activities during gift-giving occasions. According to the 2019 Bankrate Holiday Gifting Survey, 7 out of 10 people experience “extreme stress” over the holiday season. Statistics on holiday stress show that 51 percent of the survey respondents experience stress due to the “pressure to give or get gifts, especially on the selection process” (Garcia, n.d.). The anxiety and stress of the holiday season, has often manifested in form of symptoms such as: headaches, fatigue, concentration difficulties , sleeping disorders, short temperedness, stomach upset, muscle aches, loss of appetite, depression, behavioural change while at work or school, and a decline in productivity and work performance (“Holiday Stress | Managing Holiday Stress | Stress During the Holidays,” 2014).

Holiday stress and gifting anxiety often stems from the fact that picking the perfect gift to give someone is a formidable task as gifts are often wrapped in a symbolic meaning. Successfully picking an ideal gift calls for a celebration as the recipient feels understood, closer to the giver, surprised, appreciated, delighted, and even impressed by how well the giver knows him/her. Failure on the other hand, is cause for regret as your “wrong” gift is met with a feigned smile and forced thank you or with a “It’s the thought that counts” statement. This can cause a strain in the relationship, as the receiver may feel wrongly perceived, misunderstood, and possibly even insulted. Gifting anxiety is also experienced when finding an affordable and thoughtful gift with limited time and/or money (“Gift-giving Anxiety,” n.d.).

The criteria people use to purchase a gift for someone may vary from one person to the other. Some define the ideal gift by catching hints given by the receiver or choose a gift based on what they think the recipient would like, and then goes to hunt, either in physical stores or online shops, to find a budget friendly gift.

Some companies have tried to automate the process but failed. In 2011, a gift finder which recommended preferred gifts using Facebook profile updates, likes and posts was launched by Etsy but was short lived. Shopycat, another gift recommendation platform that relied on data from Facebook was released in the same year by Walmart but after a couple of months shuttered the service. Later Hunch was developed, and the accuracy of its recommendations depended on answers given by users as the system generated recommendations by asking questions to the user. Its success led eBay to purchase it for $80 million in 2011 but in 2014 it was shut down (“Can AI Finally Help Humans Choose the Right Gift?,” n.d.). Currently, areas where machine learning has gone to automate the process of gift giving, is using collaborative filtering in ecommerce websites to help one pick a gift. Examples of these websites are gifts.com, purpink and amazon gift finder. These attempts however have the shortcoming of not exactly personalizing the gift selection as it is not based on aspects such as the passions, hobbies, and interests of the recipient.

The proposed application is intended to automate the process by utilizing hybrid recommendation algorithms. A user profile will be generated for the users of the system when the system collects information on their behavior within the platform and their interests that will be collected once a user registers into the system. The recommendation algorithms that will make this possible comprise of; content-based filtering, where items will be recommended based on purchase history, collaborative filtering, where similar users are recommended for similar products and knowledge-based filtering, which will collect user information. These algorithms aid in the recommend the ideal gift to purchase for someone as the system will require the user to enter information about the recipient of the gift and a recommendation will be made based on the user profile the recipient belongs to. Similar user profiles get similar recommendations. This application will reduce the gifting anxiety givers have when picking out an ideal gift and the amount of time and energy used to find the gift.

## 1.2 Problem Statement

Gift giving has been a daunting task since the origin of the practice in prehistory. Because of the essentially personal nature of the activity, many gift givers are intimidated by the very real chance of selecting a gift not pleasing to the recipient. There is often a chance of misconception from the receiver, and the possibility of a strained relationship. Even for those who can discern how the gift will be perceived by the recipient, the task may still be very time consuming or stressful.

The current common method of selecting a gift, is based on what the sender thinks the recipient would like. There are e-commerce platforms that aid with the search for the ideal gift. Regardless, choosing a gift that can strain a relationship, is still quite high. This is because the recommendation systems in the e-commerce platforms are not personalized enough. The proposed solution intends to solve this problem by using personalized data of the recipient i.e. recipient’s demographic information and interests and matching it to a user profile. A hybrid recommendation engine is then implemented to recommend a gift to the user, thus making the process more effective and less time consuming.

## 1.3 Objectives

### 1.3.1 General Objectives

To develop a hybrid recommender system that recommends a personalized gift based on the interests of the recipient.

### 1.3.2 Specific Objectives

1. To find out the current method people use to select a gift for their loved ones.
2. To study and analyse the challenges that people undergo when choosing a gift.
3. To research on types of recommendation systems and how they work.
4. To critique existing gift selection applications and their short comings.
5. To build and test the proposed solution

## 1.4 Research Questions

1. What is the current method that people use to select a gift for their loved ones?
2. What are the challenges they face when choosing gifts?
3. What are the types of recommendation systems used and how do they work?
4. Are the existing systems efficient in helping users choose gifts?
5. How will the proposed system be built and tested?

## 1.5 Justification

When the holiday season approaches or during occasions such as graduations, birthdays, or weddings, there is need to get a gift for the involved party. The problem comes in not only in selecting a gift that is thoughtful and within the budget and but also in thinking about how the gift you choose will be perceived by the recipient. There are so many gift ideas to choose from such as; monetary gifts, a purchased product or service, homemade objects, a pre-owned product and property, or even a body organ or blood (Belk, n.d.). With so many options, it’s a bit difficult to choose the preferred gift, that will convey the message you want to send to the recipient, without experiencing holiday stress or gifting anxiety (“Gift-giving Anxiety,” n.d.).

The proposed solution intends on making the process of selecting the perfect gift much easier and less time consuming. It intends on reducing holiday stress caused by pressure to give gifts and gifting anxiety experienced when selecting gifts to purchase for loved ones. The system will only require the gift giver to enter some personal information like gender, age, occasion, interests, and the relationship of the person they are getting a gift for and the application will automatically make a recommendation.

## 1.6 Scope and Limitations

This will be a web application will collect user data and create a user profile based on that data. The data will include their interests, hobbies, preferences, and user behavior i.e. purchase history, previous searches, views and clicks. The web application will use a hybrid recommender system, to analyze the given data and produce recommended products to the user. The user will have the option of purchasing the recommended product from the system. The target users for the system will include, family members, friends, and organizations dealing with specific occasion and events.

The system will not include a mobile based application. The immediate test case will be local based consumers. To use the system, one must have internet connection as it will be a web application.

# : Literature Review

## 2.1 Introduction

The literature review discusses the current ways used to select an ideal gift for someone and the challenges faced when choosing the gift. It also explores the application of recommendation systems in ecommerce and specifically when trying to select a gift for a loved one. This will include reviewing existing systems, focusing on their pros and cons, with the idea being how the system can be improved.

## 2.2 Current Methods People Use to Select Gifts and The Challenges They Face.

The current common method of selecting a gift, is based on what the sender thinks the recipient would like, based on how well the sender knows the recipient. Sometimes the sender defines the ideal gift by catching hints given by the receiver, and then goes to hunt, either in physical stores or online shops, to find a budget friendly gift.

According to a survey by Photobox Group, 48% of people would trust technology to help make personalized gift choices; thus, they opt to use e-commerce sites to pick out a gift. This is because it is more convenient as you can shop from the comfort of your home (“AI gifting ideas for the Holiday season,” 2020). E-commerce sites often suggest what someone would like to buy, based on what most users of the e-commerce application liked or purchased.

Whether the sender is selecting a gift for his/her spouse, aunt, nephew, best friend, or parent, and he/she claims to know them, it's rather surprising how difficult selecting a gift based on the recommendations given can be a formidable task. This because those people are not him/her. The sender might like a product for himself/herself, but it would not be appropriate for the recipient. This is because the gift selected must still say that the sender put some thought into it and will convey the message it is intended to portray. Since, the e-commerce platforms do not give recommendations that are entirely personalized to a specific user, the whole process of selecting a gift remains time consuming, energy draining and stress inflicting.

## 2.3 Recommendation Systems

Since the publishing of the collaborative filtering research in the 1990s, recommendation engines are a significant research field (Park and Kim, n.d.). A recommendation system is a powerful information filtering technology used to predict whether a specific user will like a specific item or to identify a set (n) items that will be of interest to a particular user (S and D, 2017).

### 2.3.1 Phases of recommendation process

This section focuses on the three phases that a recommendation system goes through for a recommendation to be made.

### 2.3.1.1 Information Collection Phase

This phase enables a recommender system to make an accurate recommendation by collecting useful data of the users of the system then a user profile is generated. This information could include, the user’s demographics, behavior, interests, preferences or even content the user access. The knowledge the engine requires to make a reliable recommendation should be as much as possible. The system relies on inputs, from explicit feedback, which is formed when users provide explicit input about their preferences on a certain product or from implicit feedback by acquiring users interests indirectly through the observation of a user’s behaviors within the system. The ability to represent accurate and useful user interests, dictates how successful a recommendation engine is (Isinkaye et al., 2015).

To obtain explicit feedback, the engine prompts its users to give some feedback in relation to the ratings of a product or their general preferences. This enables the system to construct or improve the recommendation model. The shortcoming this information collection has is that it requires personal effort from a user, yet the user may not prefer to provide information. Despite that fact, this method is more reliable as it does not involve obtaining interests from user behaviour. Transparency is also provided in the recommendation process, thus there is more confidence in the recommendations made (Isinkaye et al., 2015).

To obtain implicit feedback, the preferences of a user is automatically inferred by the engine. This is made possible by monitoring the activities if a user within the system. These activities include, historical purchases of the user, time spent on different web contents, navigation history, followed links, button clicks, among others. A lot of effort is not required from the user as opposed to the explicit feedback. This method could be more unbiased compared to explicit feedback where a user could respond based on what he/she thinks is expected (Isinkaye et al., 2015).

Implicit and explicit feedbacks are combined in the hybrid feedback method. This enables maximization of strengths of individual methods while minimizing their individual limitations. This is achieved by utilizing data acquired implicitly to confirm a rating obtained explicitly or by giving the user the opportunity to provide explicit feedback.(Isinkaye et al., 2015)

### 2.3.1.2 Learning Phase

From the data gathered in the information collection phase, a learning algorithm is utilized to filter and handle user attributes. The learning algorithms aid in drawing out appropriate patterns for the application depending on the scenario. Learning can be conducted using statistical analysis models or machine learning algorithms. This includes Term Frequency Inverse Document Frequency (TF/IDF) which is a vector space model or Decision Trees, Naïve Bayes Classifier and Neural Networks which are examples of probabilistic models. The algorithm used depends on the type of recommendation technique being implemented.

### 2.3.1.3 Prediction/recommendation phase

The engine recommends or predicts items a user would most likely prefer based on data gathered from the information collection phase or a memory-based collaborative model that relies on data inferred from the behaviour of a user.(“Machine Learning for Recommender Systems - A Primer,” n.d.) The recommendation process is made possible by a couple of methods. For example, matrix factorisation, similarity measure methods or supervised learning. These methods are implemented in different types of recommender engines.

### 2.3.2 Types of Recommendation Systems

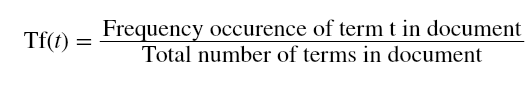
In this section, we will discuss common types of recommender engines. These types include content-based filtering, knowledge-based filtering, collaborative filtering and hybrid recommendation systems.

### 2.3.2.1 Content-based Filtering

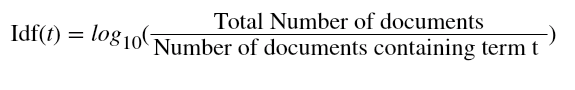
Content-based Filtering hypothesizes that users who liked items with certain attributes in the past, in the future, they would prefer the same kind of items as well. It makes recommendations using user profile features and item attributes. Similar items are grouped using their attributes. A User profile is generated from the user’s previous interactions with the recommendation system or by asking the users questions on their interests explicitly.

The details of the item attributes a stored in text format. To enable easy calculation of item similarity, these details must be converted to numbers. Term Frequency-Inverse Document Frequency (TF-IDF) is employed to measure the originality of a word. This is done by making a comparison between the number of times a word is present in a document with the number of documents the word appears in. It helps in the evaluation of the importance of a word in a document. It is computed by multiplying term frequency with inverse document frequency.

Term frequency is how many times a word appears in a document compared to the number of words in the document.



While inverse document frequency is the number of documents compared to the number of documents that the word is contained (Porta, 2020).



Obvious recommendations are made when using this filtering technique because it has a limitation of excessive specialization. This means when a user U has an interest in categories X, Y, and Z, the engine will not be able to recommend products that are not in that category, despite the chance that the user could have interest in them.

A new user will lack a user profile if they are not asked to provide the information explicitly. This technique has a limitation when it comes to adding new users into the system (“Introduction to Recommender Systems in 2019 | Tryolabs Blog,” n.d.).

Content based filtering has been implemented in Netflix, a movie streaming application to make user-based recommendations based on what they have watched before.

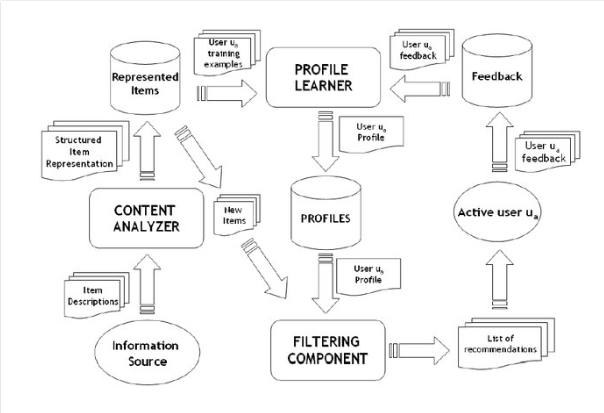


Figure 2.1:Content-based recommendation system

### 2.3.2.2 Collaborative Filtering

This filtering technique simultaneously uses similarities between users and items to generate recommendations. This addresses the limitations of content-based recommendations. Collaborative filtering systems assume that if a user U shows interest in an item X and user B shows interest in item X and item Y, user U could also be interested in item Y. New interactions are therefore predicted using historical data of the systems users. Memory-basedandmodel-based methods are the two types of this filtration method. (“Introduction to Recommender Systems in 2019 | Tryolabs Blog,” n.d.).

Memory-based collaborative filtering consists of the following methods:  item-basedcollaborative filtering and user-based collaborative filtering. Nearest Neighbour approach is used to find out either similar users or similar products.

In user-based collaborative filtering, we hypothesize that two items that receive ratings similar to each other from the same user are similar. A prediction of an item will then be made to the user when a calculation of the average weighted ratings on the highest number of similar items according to the user is done.

Products that are similar to the product that a user previously purchased are recommended in item-based filtering. This is because similarity is based on co-occurrences of purchases. For example, If item X and Y were purchased by both users X and Y then both are similar (M, 2019).

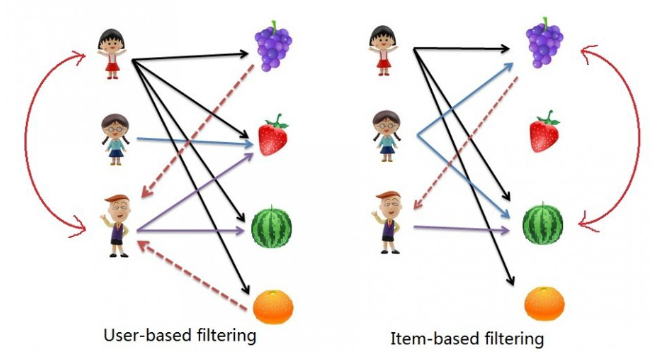


Figure 2.2: Memory-based Collaborative filtering

Model-based collaborative filtering algorithms on the other hand, employ data mining and machine learning techniques. Models are trained to make predictions using these methods. For example, the top-N products that a user may prefer can be predicted by implementing an existing user-item interaction to train the model. This method is more advantageous compared to memory-based method because it can be employed to large datasets of items and users. The machine learning algorithms that aid in the formation of the model include **clustering,** **rule-based** approaches, and **Bayesian network.**

The main problem of using these systems include Cold start. When new users of a system have not interacted with products in the generation of recommendations becomes a challenge due to the scarcity of data needed to make accurate predictions. This mainly occurs when the engine is newly setup. (“Model-based Collaborative Filtering Algorithms.,” n.d.)

### 2.3.2.3 Hybrid Recommendation systems

This technique is a combination of two or more types of recommender algorithms. The main objective of the implementation of this technique is to cater for the limitations that exist in the individual implementation of the other types of recommendation techniques. The combination of collaborative recommendation and content-based recommendation is the main hybrid approach that can occur. (Pereira et al., 2017)

The performance of the hybrid approach is more efficient compared to the performance of the individual approaches on their own. This is because when combining different techniques their individual shortcomings are eliminated. An example is the cold start issue.(“How do Recommendation Engines work?,” 2017)

There are four main combination approaches that are used to combine content-based filtering and collaborative filtering which are described as below:

**Combining Results Separately:** When combining results separately, each recommendation technique operates separately but the best results or recommendations of the individual techniques are combined and delivered to users at the final recommendation stage.(Madadipouya and Chelliah, 2017)

**Adding Content-based Filtering Characteristics to Collaborative Filtering:** This combination method combines features of content-based filtering to collaborative filtering. In this approach, similarity between two users is calculated by use of content-based user profiles and items. The aim of implementing this approach is to solve the data scarcity problem and minimizing provision of poor recommendations to users. (Madadipouya and Chelliah, 2017)

**Adding Collaborative Filtering Characteristics to Content-based Filtering:** This combination method combines collaborative features to content-based filtering.

**Developing a Single Unified Recommender System:** In this technique, a unified model is created that utilizes features of Content-based and Collaborative methods. This method is studied extensively and focuses on making more accurate recommendations. (Pereira et al., 2017)

### 2.3.2.3.2 Techniques for Hybrid Approach

In Weighted hybridization, a recommendation is generated by merging the outputs of the recommender engines implemented and the scores from each of the methods is computed by a linear formula to determine the recommended item score. Collaborative filtering and Content-based are both implemented. When the engine starts running, they are both assigned equal weights, but the weights change as the recommendations change. All the strengths of the techniques implemented are utilized in the engine (Isinkaye et al., 2015).

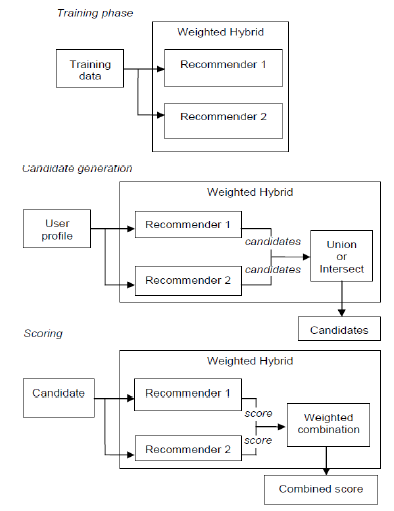


Figure 2.3: Weighted hybridization

Switching hybridization enables the system to swap between the implemented recommendation methods depending on which method can produced a more accurate recommendation. In a collaborative/content hybrid, the content-based technique will be employed first. If it is unable to make an accurate prediction, the collaborative filtering method is executed.

Additional complexity is introduced when the switching criteria is being determined. parameterization also increases(Isinkaye et al., 2015).

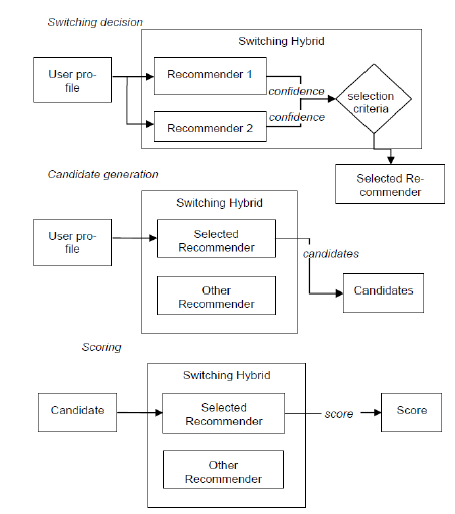


Figure 2.4:Switching hybridization

Mixed hybrids is implemented to enable generation of predictions from at least two methods to be displayed collectively.This method is implemented in the PTV application (Smyth and Cotter, 2000) when a recommendation of a program from the TVs film database is generated. Content-based technique is implemented on the text descriptions of the shows and collaborative technique is used on information about the other user’s preferences. For the final suggested program to be displayed, the recommendations from the two techniques are merged.(Isinkaye et al., 2015)

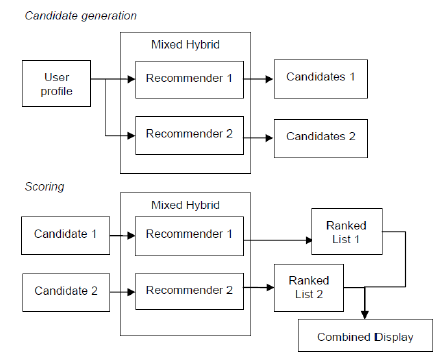


Figure 2.5:Mixed hybridization

In feature combination approach, the features generated by one recommender method is made available to the other recommender methods. For example, similar user’s ratings, which is a collaborative filtering feature, is used to when the similarity of products is being determined in a case-based recommender engine. This approach enables the recommender engine to use collaborative data if needed for more accurate results.

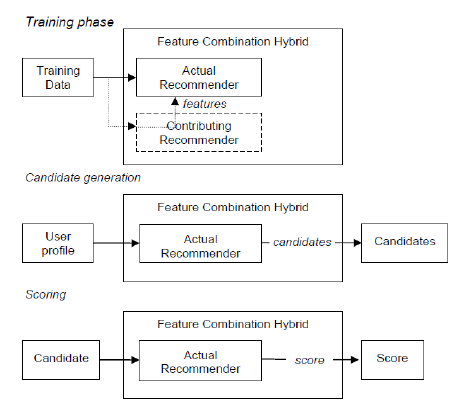


Figure 2.6:Feature Combination Hybrid

The cascade hybridization technique involves a staged process, where the first recommendation method implemented, generates a coarse ranking of candidates and the other recommendation method refines it. This technique is efficient and noise tolerant because of the coarse-to-finer nature of the staged process. The system avoids executing the second technique on already well-defined items by the first. It also ensures that items that have a low rating will not be predicted at any given time.

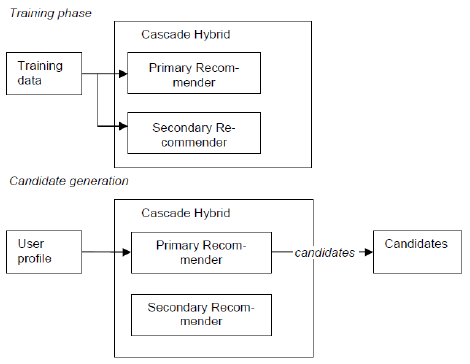


Figure 2.7: Cascade hybridization (a)

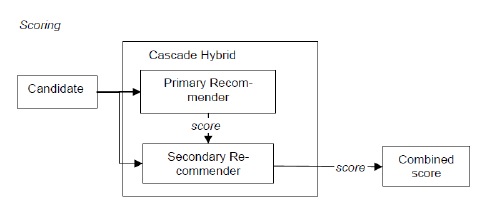


Figure 2.8: Cascade hybridization (b)

### 2.3.2.4 Knowledge based filtering

A recommender system is knowledge-based when it makes recommendations based not on a user’s rating history, but on specific queries made by the user. It is implemented in specific domains where the user’s purchased history is small. This is mainly for new users of the system. Before giving recommendations, the algorithm considers, the features of the items that meet user taste, user’s preferences that were asked explicitly and recommendation criteria. The model’s accuracy is established based on how much the user likes the recommended item.

An example is when you are developing a recommender engine the aids in the recommendation of household electronics and the users of the system are all new users. Additional data like specification is obtained from the users for the generation of user profiles and the features of the products is considered. The term for these engines is constraint-based recommendation systems. (*Knowledge-based recommender systems - Building a Recommendation System with R*, 2015)

### 2.3.3 Evaluation Metric for recommendation systems

Different types of measurements, which could be coverage or accuracy, are used to estimate the effectiveness of a recommender engine. The type of recommendation algorithm implemented dictates the type of metric used. Accuracy is the percentage of correct predictions made from the total predictions, while Coverage is the ratio of users that a filtration algorithm can provide item recommendations for or number of products that can be given as a recommendation to a user. The metrics that are used to measure accuracy of the recommender engines, are divided into decision support and statistical accuracy metrics.

### 2.3.3.1 Statistical Accuracy Metrics

This evaluation metrics evaluates the accuracy of a recommendation method based on a comparison between predicted ratings and the actual user ratings. Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and correlation, are employed as a statistical accuracy metric. Mean Absolute Error (MAE) is the measure of deviation of recommendation from user’s specific value. It is a common and very popular method. It is calculated as follows,



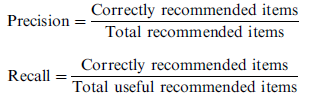
Where, **pui**is the rating prediction of a user **u** on item **i**, **rui**on the other hand is the actual rating of the item and **N** is the number of all the ratings of an item. The lower the MAE, the more accurate the filtration engine is in predicting ratings provided by a user(Isinkaye et al., 2015).

A filtration engine with a low RMSE is more accurate than one with a higher one.It is computed as follows;



### 2.3.3.2 Decision support accuracy metrics

This evaluation metrics assists users of the system in selecting the most relevant items in a sample of available products. The system of measurement that are commonly used in this category includes, Weighted errors, Reversal rate, Precision Recall Curve (PRC), and Receiver Operating Characteristics (ROC), Recall, Precision and F-measure. The listed metrics view the procedure of prediction as a process that helps distinguish good products from bad. When a comprehensive assessment of an algorithm’s performance is being performed, ROC curves are used. Recall is the ratio of preferred products that are part of the set of predicted products while Precision is the ratio of predicted products that is useful to the system’s user. They are calculated as follows:



F-measure simplifies recall and precision into a single system of measurement. The comparison between filtration methods and data sets is made very simple and straightforward using this method. It is calculated as follows.



(Isinkaye et al., 2015)

## 2.4 Related Works

This section focuses on existing systems that apply similar or an almost similar technology to recommender systems.

### 2.4.1 Amazon

Amazon is an e-commerce platform that enable users to buy goods from the comfort of their home. Amazon uses its recommendation algorithm throughout its website as a targeted marketing tool. It recommends products that a user may be interested in purchasing by sorting the user’s visited, searched, previously purchased and rated products in an item-to-item matrix, then cosine similarity is used to compute the recommendation algorithm. This enables the recommendation of similar products to be displayed to the user. The system has implemented Item-to-item collaborative filtering method mainly because it generates high-quality real time recommendations and scales to massive data sets. A personalized shopping experience for the users has been made possible by their recommendation algorithm. This results to an increase in average order value thus the amount of revenue generated from each customer increases. The biggest problem with this method is the cold start problem. This is the fact that the system is unable to give recommendations until the new users is more familiar with the system.

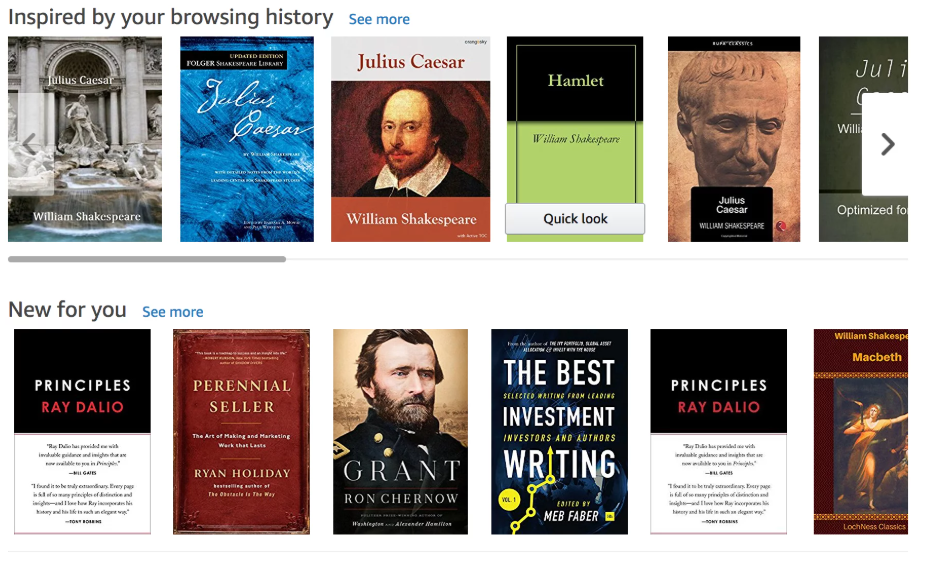


Figure 2.9: A screen shot of one of amazon users homepage

### 2.4.2 Netflix

Netflix is the largest movie and series streaming application in the world. The online television network is successful mainly because it has implemented a recommendation algorithm that generates personalized recommendations of movies and series to its users. There are 139 million subscribers who pay of Netflix globally and almost 300 million who watch tv shows from the platform. The artificial Intelligence model enables Netflix to make a profit of almost $1 billion annually.(India, 2019)

Netflix recommendation technology uses machine learning, making it easy for a user to find what they would personally like to watch. Around 70% of what Netflix users watch is what is recommended to them. Recommendations on Netflix start a as soon as a person signs up where, feedback is collected explicitly from a taste survey where the user fills out what they are interested in. In addition to what a user watches Netflix also looks at what others watch that could be a good match for you. The system creates personalised genres that take movie attributes such as actors, time period, story lines etc(Underwood, n.d.)

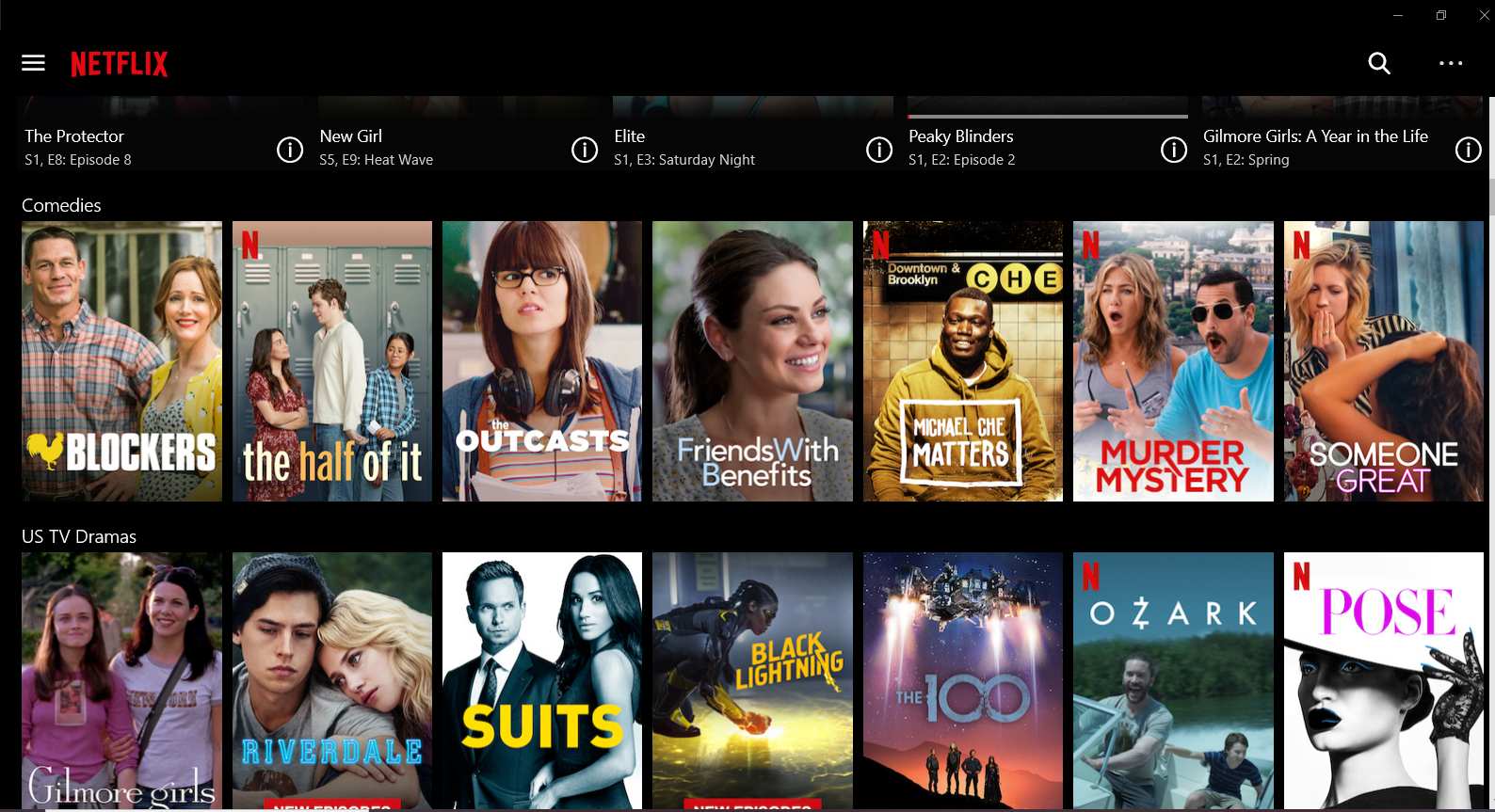


Figure 2.10: Screenshot of User interface of Netflix with movie recommendations.

### 2.4.3 YouTube

YouTube is an online platform where people from all over the world share content in form of videos. It uses a deep neural network technique to make personalized recommendations for its users. This enables the users to easily find videos that they may be interested. The recommendations get updated regularly as the user interacts with the system. This keeps a user engrossed for hours due to its amazing recommendation. The recommendation system is implemented using two neural networks where one gathers information on what users have previously watched and utilizes collaborative filtering to select many videos. It uses feedback from the users to train the model. The second neural network is used for ranking the videos from the most popular to the least popular. 60 percent of recommended videos are often clicked by its users thus confirming the success rate of the recommendation system (Underwood, n.d.).

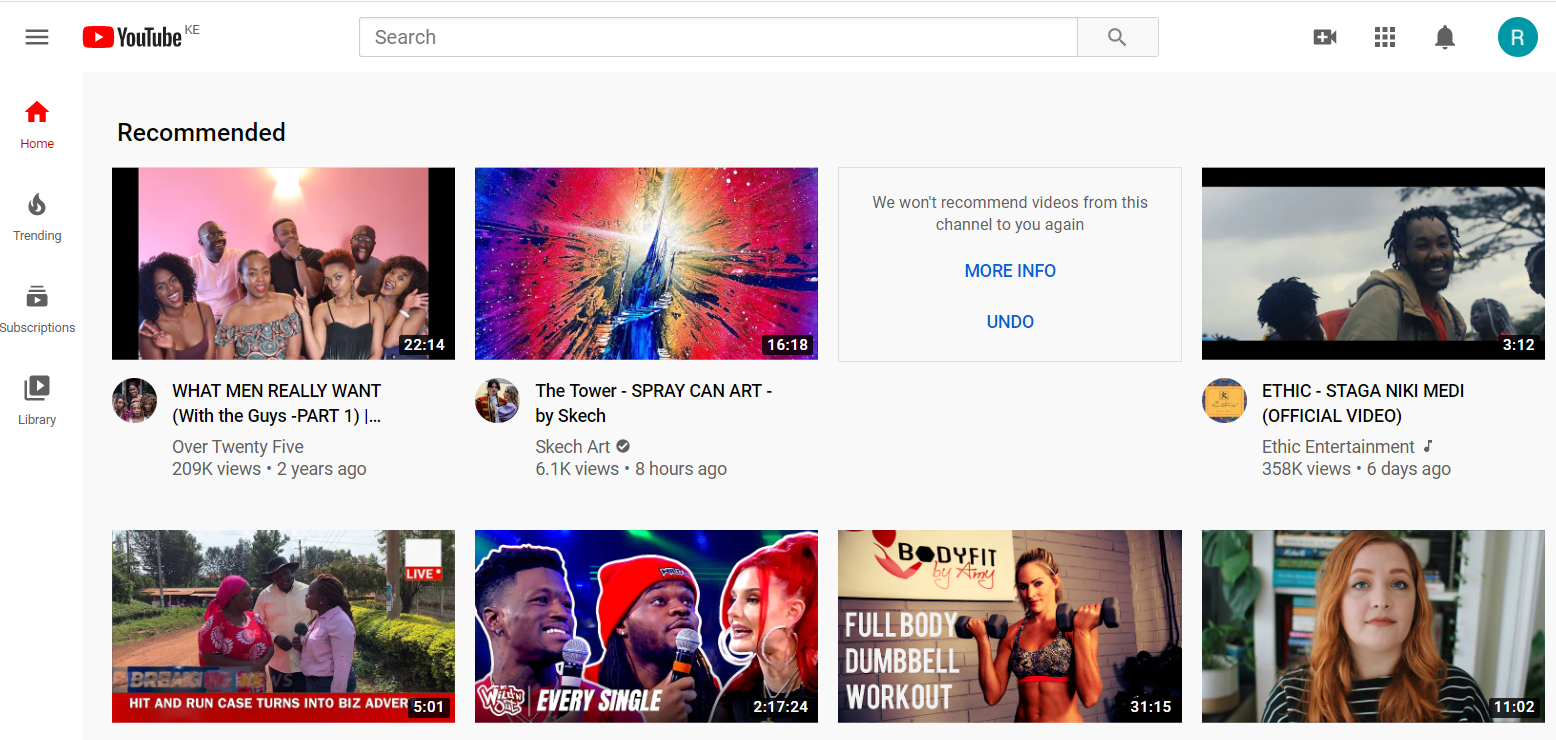


Figure 2.11: Screenshot of YouTube homepage with video recommendations

## 2.5 Review of methods used

Recommendation engines create new opportunities for retrieving information that is personalized on the internet. When implemented in a business website, recommendation engines tend to increase profit in the business as it helps customers in decision making. The type of recommendation technique one choses to employ in their application might not be personalized enough to help a customer make a decision, if use case scenario of why the customer is using the platform is not catered for. There are a few short comings that can come about depending on the recommendation technique implemented in an application.

The limitations of implementing content-based filtering technique is the fact that it does not capture complex behaviour patterns of a user or inter-dependencies. For example, one might like machine learning articles that both the practical application and the theory are present, and not just the theory. This kind of information will not be taken into consideration by this technique. Another limitation may come about when a user’s preferences change over time, the system will still recommend items based on the user’s historical taste and the system will end up being useless to the user.

Implementing collaborative filtering technique, one will face a cold start problem which means, the system will not be able to make recommendations to a new user or will make poor recommendations if there is not enough data. This problem is mainly present in cases where there is a new user, a new item or when a system is new. Collaborative filtering might have the shortcoming of not being entirely personalized for users as it operates on the assumption that similar users like similar product. This might not entirely be the case when it comes to a specific user of a system.

Implementing knowledge-based filtering, the system will not improve over time as the suggestion ability is static, this is because the system only gets information about a user when the user logs in to the system for the first time. The user’s preferences might change over time and the system will not be able to adjust accordingly to the change in the user’s interests.

## 2.6 Conceptual Framework of the proposed solution

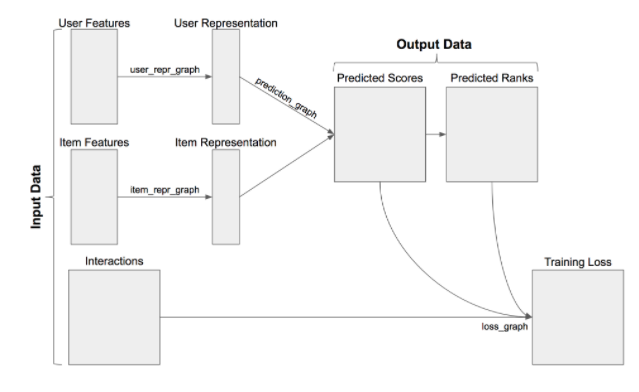


Figure 2.12:Conceptual Framework

# : Research Methodology

## 3.1 Introduction

A methodology is a body of practices, rules, and procedures to be used in the developing of a system that conceptualizes an idea (Labaree, n.d.). This chapter discusses the software development methodology used in the project and highlights the system analysis and design approaches used. It also describes the research design, methods, and approaches to data collection. It is guided by the research questions and objectives outlined in chapter one. Finally, the chapter gives a description of the tools and techniques to be used in the implementation of the project.

### 3.2 Research Design

Research design refers to the overall strategy that you choose to integrate the different components of the study in a coherent and logical way, thereby, ensuring you will effectively address the research problem; it constitutes the blueprint for the collection, measurement, and analysis of data(Labaree, n.d.).

This project will use a qualitative research design that is exploratory in nature. This is ideal when one wants gain concrete, contextual, in-depth knowledge about a specific real-world subject. This study explores the human behavior during the process of gift selection.

## 3.3 Model Development

### 3.3.1 Data Pre-processing

Data Pre-processing is done to deal with incomplete, noisy or even inconsistent data in a dataset. Incomplete data in this case can lead to inaccurate modelling due to lacking attributes of interest. The data can also be containing errors and outliers which are then smoothed or removed which results in the resolving of the inconsistencies. Pre-processing in this case will be done using python libraries like pandas.

## 3.3.2 Model Training and Testing

The proposed solution was the development of a hybrid recommendation engine. Deep learning models can be very powerful in combining collaborative filtering and content-based filtering. There are several techniques and methods for building hybrid recommender system. But in this project, Matrix Factorization using Alternating Least Squares(ALS) was used. It turns out we can if we apply matrix factorization. Often, matrix factorization is applied in the realm of dimensionality reduction, where we are trying to reduce the number of features while still keeping the relevant information. This is the case with principal component analysis (PCA) and the very similar singular value decomposition (SVD). Essentially, we can take a large matrix of user/item interactions and figure out the latent (or hidden) features that relate them to each other in a much smaller matrix of user features and item features. That’s exactly what ALS is trying to do through matrix factorization.

Let’s assume we have an original ratings matrix R of size MxN, where M is the number of users and N is the number of items. This matrix is quite sparse since most users only interact with a few items each. We can factorize this matrix into two separate smaller matrices: one with dimensions MxK which will be our latent user feature vectors for each user (U) and a second with dimensions KxN, which will have our latent item feature vectors for each item (V). Multiplying these two feature matrices together approximates the original matrix, but now we have two matrices that are dense including a number of latent features K for each of our items and users.

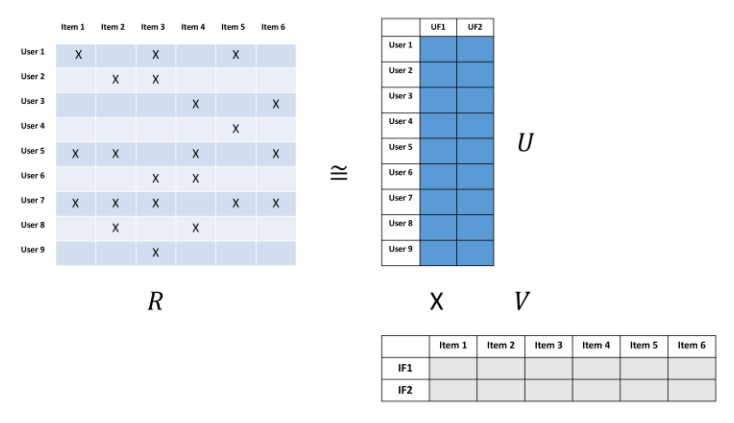


Figure 3.1: Matrix Factorization

In order to solve for U and V, we could either utilize SVD (which would require inverting a potentially very large matrix and be computationally expensive) to solve the factorization more precisely or apply ALS to approximate it. In the case of ALS, we only need to solve one feature vector at a time, which means it can be run in parallel. To do this, we can randomly initialize U and solve for V. Then we can go back and solve for U using our solution for V. Keep iterating back and forth like this until we get a convergence that approximates R as best as we can. After this has been finished, we can simply take the dot product of U and V to see what the predicted rating would be for a specific user/item interaction, even if there was no prior interaction.

The metrics for evaluating the performance of the trained model is AUC score. This can be very useful while working towards building better models and achieving higher accuracy.

## 3.4 Software Development Methodology

In this project the methodology used is Rapid Application Development or RAD. It is an approach to designing software applications where a prototype of the application is developed quickly and then refined through several iterations(“What is Rapid Application Development, and Is It Right for My Business?,” 2020). What makes RAD ideal for this project is the fact that it is flexible, useful when one needs deliver an application in a very short amount of time and responsive to user input. This is ideal for a recommendation engine where the objective is to make a system whose customer experience is worthwhile.

The rapid application development method contains four phases: requirements planning, user design, construction, and cutover. The user design and construction phases repeat until the user confirms that the product meets all requirements.

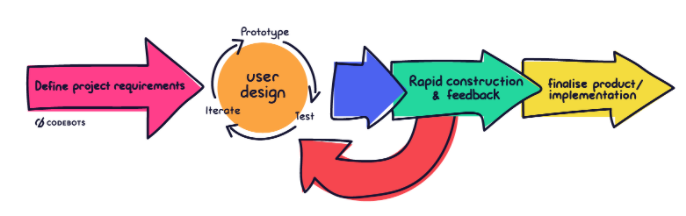


Figure 3.2: Rapid Application Development Diagram

### 3.4.1 Requirements Planning

During this phase project stakeholders, define and finalize project requirements. The requirements determined can be categorized into business modelling, Data Modelling, Technical requirements, and Interface requirements.

In this phase, final productivity measures of the project are determined when modelling the business. The information acquired during the modelling of the business is reviewed and analysed and data objects that are vital for the business are formed. The attributes of all data sets and their correlations with relevance to the business model are established and defined in detail. The sources and systems involved in data collection are also determined during data modelling.

The technical requirements are also defined in this stage. This refers to how and where the application will run, including both user operating system and backend infrastructure. The Interface requirements are also determined in this stage. This is what the application will look like. These requirements are captured once, but the iterative approach of RAD means they may change over time (O’Carroll, 2020).

The proposed system will use both primary data collected from the users and secondary data from Kaggle.com and it will be a web application that will run on all available web browsers.

### 3.4.2 User Design

Once the scope of the project is chalked out, development begins by building out the user design through various prototype iterations. The goal is to rapidly produce a working design that can be demonstrated to the users. The developer works together with users until a final product is ready, to ensure the users’ needs are being met. This step is often repeated as often as it is necessary. This phase is mainly multiple loops of; designing the initial prototype, testing the prototype with consideration of user and technical requirements, Refining the prototyping until it reaches a suitable end-state.

Through prototyping, the development team can easily evaluate the feasibility of complex components. Consequently, software is more robust, less error-prone, and better structured for future design additions (O’Carroll, 2020).

There will be the design of the Unified Modelling language (UML) diagrams to describe flow of information and interaction of the different system components. These diagrams include; context diagrams, sequence diagrams, ERD diagrams and a database Schema.

### 3.4.3 Rapid construction and feedback gathering

Rapid construction is where application coding, system testing, and unit integration occurs, converting prototype and beta systems into a working model. Ecommerce application and the recommendation engine will be tested during this phase, ensuring the result satisfies user expectations and objectives. The Developer works with end users to collect feedback on interface and functionality and improve all aspects of the product. Users give thorough input throughout this stage, suggesting alterations, changes, or new ideas that solve problems as they are discovered. Users may find that in practice, some concepts don’t work. With this input, the developer either resume prototyping, or if feedback is strictly positive, move onto the final step (O’Carroll, 2020).The development language that will be implemented during this stage is Python.

### 3.4.4 Finalise product / implementation.

The final phase of rapid application development is where developers may optimize or even re-engineer their implementation to improve stability and maintainability as they finalise the gift recommendation application for launch. The implementation phase is where the developer moves components to a live production environment, where any necessary full-scale testing for product bugs or user training can take place. The developer writes a thorough documentation and complete other necessary maintenance tasks, before confidently handing the user a complete product (O’Carroll, 2020).

## 3.5 System Deliverables

By the end of the project the system will deliver:

### 3.5.1 Recommendation Engine

The system should be able to recommend a gift for a new user based on the user behavior in the system.

### 3.5.2 User Module

The user should be able to create an account, log in into the system, add items to cart, input interests, Wishlist items, view items, purchase an item and log out of the ecommerce system. This will in turn build a user model for recommendation.

### 3.5.3 Admin Module

The Admin should be able to create an account, log in into the system, add products, delete products, add brand name, assign product tag values to the products and select category of the product.

### 3.5.4 System documentation

A documentation that will include a step by step manual that will guide a new user on how to use the system.

## 3.6 System Development Tools and Techniques

When developing the system, we will use the tools that are commonly known to create web-based applications. These include:

### 3.6.1 Visual Studio Code

Visual Studio Code is a light weight but powerful source code editor.

### **3.6.2 Draw.io**

This in an online UML drawing site that contains a rich collection of tools that are necessary for aiding a developer to create a data flow diagram for the project among others.

### 3.6.3 Python Programming Language (Version 3.6)

This language will be used in the development of the machine learning model.

### 3.6.4 Django Python Web Framework

This framework will be used in the development of the web based application part.

### 3.6.5 Github

Version control used for back up of code base. It also allows for collaboration and the coordination of changes that are made by the developers without overwriting or loosing

Information.

### 3.6.6 Bootstrap

Bootstrap is a free and open-source front end framework used in styling websites and web applications. It comes with predefined class which help quicken the development of websites by making styling easy. It is a very powerful tool which will reduce development time and provide uniformity in code for better team collaboration.

## 3.7 Ethical Considerations

The research study is subject to certain ethical issues and considerations. The researcher should ensure the participant in the evaluation is fully informed about the evaluation being conducted and made aware of the purpose of the project. They should at no point feel any coercion to participate in the study. The confidentiality of the information supplied by research subjects and the anonymity of respondents must be respected.

# : System Analysis and Design

## 4.1 Introduction

This chapter looked at the system analysis and design diagrams that were used to build the Recommendation system. It also highlights the database schema, system narrative, functional and non-functional requirements of the system.

## 4.2 System Requirements Analysis

System requirement analysis refers to the process of determining the user expectation of a new system.(“System Requirements,” n.d.)

### 4.2.1 Functional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. (“Functional and Nonfunctional Requirements,” n.d.).

For this particular system, the functional requirements include:-

1. Gift Recommendations- The system recommends gift to the user. This could be for the user herself or for the recipient of the gift. The recommendations are given based on the interaction between a user and an item. In this case it will be the purchase history of the users. Going by the assumption that both the giver and the recipient are both users of the system.
2. Authentication- The buyers and the admin should be able to create an account, login into the system, change his/her password, verify his email address, and change his profile details.
3. Purchase- The Buyer should be able to add a product to the cart, and the system should be able to calculate the total amount to be paid by the buyer. The buyer should then be able to check out the products and pay for the products.
4. Manipulation of product details- The admin should be able to add and delete a product. View purchased products and quantity

## 4.2.2 Non-Functional Requirements

This captures conditions that do not directly relate to the behavior or functionality of the system. These requirements describe generally the quality attributes, design and implementation constraints and external interfaces which the system must have. The non-functional requirements include:-

1. Security - The system should prevent unauthorized users from entering it. In this case there are different modules and at no time should it allow either to enter where they are not supposed to.
2. Reliability - The system will be able to be trusted by users even after the system has been used for a long time.
3. User Friendly - An easy interface should be developed for the students and lecturers to enable a good user experience.

## 4.3 System Narrative

For a gift giver and a gift recipient to use the system he or she has to create an account then log in into the system. The giver can then view products. The giver can there after request for recommendations of a specific user provided, he/she knows the recipients email. The user can then select the ideal gift based on the recommended list which will be ranked from most popular to least popular. Once the user has seen what to buy, he or she can add the item to the cart where he later pays for the item. The user can there after logging out of the system and his details will be saved in the database for them to be accessed at a later date for better future recommendations.

For an admin to use the system, he or she has to create an account then log in into the system. The admin should later be able to add a product, manage users, monitor transactions.

## 4.4 System Analysis Diagrams

This section contains the diagrams that have been used to examine the system's working features. The diagrams specify what the system should do. Some of the diagrams which have been used include;

### 4.4.1 Context Diagram

A system context diagram (SCD) is a diagram that defines the boundary between the system, or part of a system, and its environment, showing the entities that interact with it. This diagram is a high level view of a system.

Diagram

Description automatically generated

Figure 4.1:Context Diagram

Figure 4.1 basically illustrates the relationship between the entities and the system. The user, request for recommendations of a specific user which the ecommerce system gets from the recommendation engine after requesting for it then later the user gets the recommendation. The user therefore selects a product which was added by an admin then adds it to cart and later purchases it. The admin later monitors the activities of the user.

### 4.4.2 Data Flow Diagram

This is graphically representing the processes that capture, manipulate, store, and distribute data between a system and its environment and among system components.

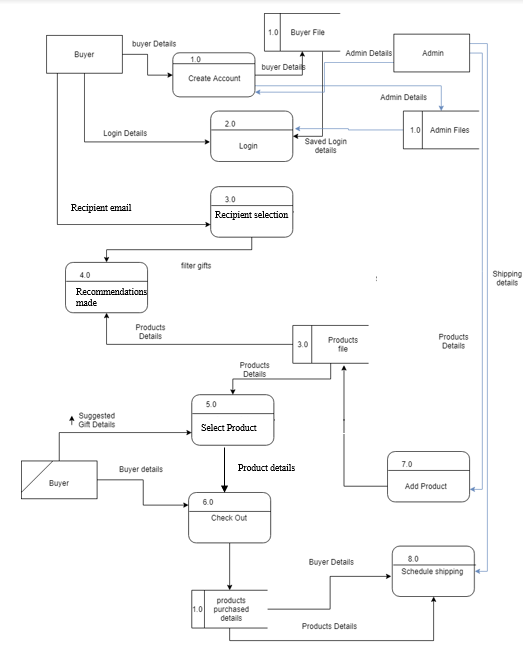


Figure 4.2:Data Flow Diagram

Figure 4.2 is a Data Flow Diagram that defines our system, The buyer creates an account using the buyer details which contains user name, password, email address, mailing address, and gender. These details are later stored in the buyer filer. The buyer can log into the system using the login details, which include the email and password that was stored in the buyer file. He can then request for recommendations for a specific user provided he/she knows the recipients email. He can then check out the product by purchasing it. The checkout details are stored in the products purchased details. The Admin can create an account using admin details which contains full name, password, email address, and gender then login using the saved login details contains the email address and password. He can then add a product by entering the product details.

### 4.4.3 Activity Diagram

Activity diagram is basically a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another.

Diagram, schematic

Description automatically generated

Figure 4.3:Activity Diagram

Figure 4.3 illustartes the activities of a user when they access the ecommerce platform. The user tries to login to the account then the details are authenticated. The user can then proceed to search for an item or get recommendation for a specific user. If the user does not make a decision, he/she can chose the vice verser option, then checkout. The user can also change account details or view previous purchases concurrently.

## 4.5 System Design Diagrams

System design is the process of defining the elements of a system such as the components, architecture and modules. It is also concerned with the interfaces of those modules and the data that goes through the system. The aim of the system design is to satisfy the specified requirements in order to develop a well-defined system. This phase also identifies how end users used it. The feasibility study done in the Analysis phase was examined again in this phase. This section contains diagrams that where used in designing the system. They include:-

### 4.5.1 Entity Relationship Diagram

ERDs, depict only structural features and provide a static view of the system. i.e. a logical representation of the entities, associations and data elements for an organization.

Figure 4.4:ERD Diagram

### 4.5.2 Database Schema

This diagram is used to show the different objects in a system, their attributes, their operations and the relationships among them.

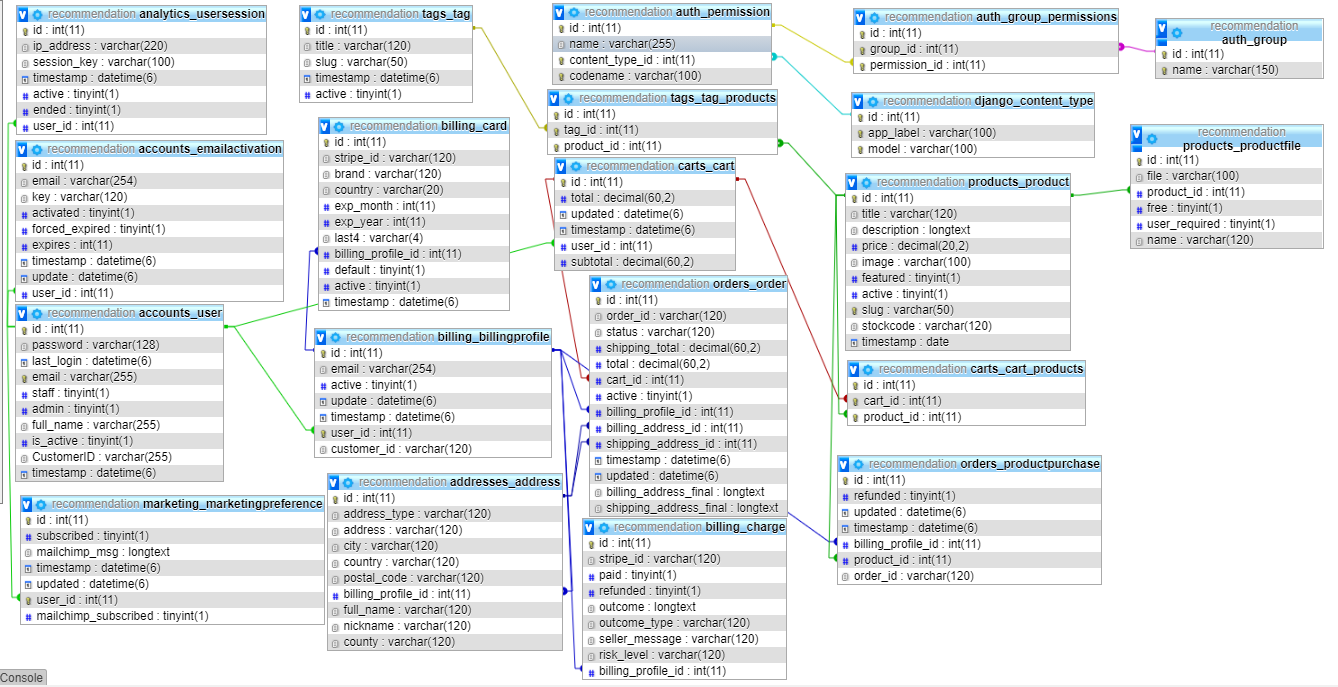


Figure 4.5:Database Schema

Figure 4.4 shows the relationships between tables in the database.

# :System Implementation and Testing

## 5.1 Introduction

This chapter focused on displaying the concepts proposed and mentioned in the previous chapters and how they were actualized. It also highlights the testing put in place, to ensure everything is working as expected.

## 5.2 Description of the dataset

The data used in the implementation of this project, comes from the infamous UCI Machine Learning repository. The dataset is called “Online Retail” and is found [here](https://archive.ics.uci.edu/ml/datasets/Online+Retail). This dataset contains all purchases made for an online retail company based in the UK during an eight-month period.

I used the pandas profiling library, which generated a summary of the dataset. This is shown in the figure 5.1

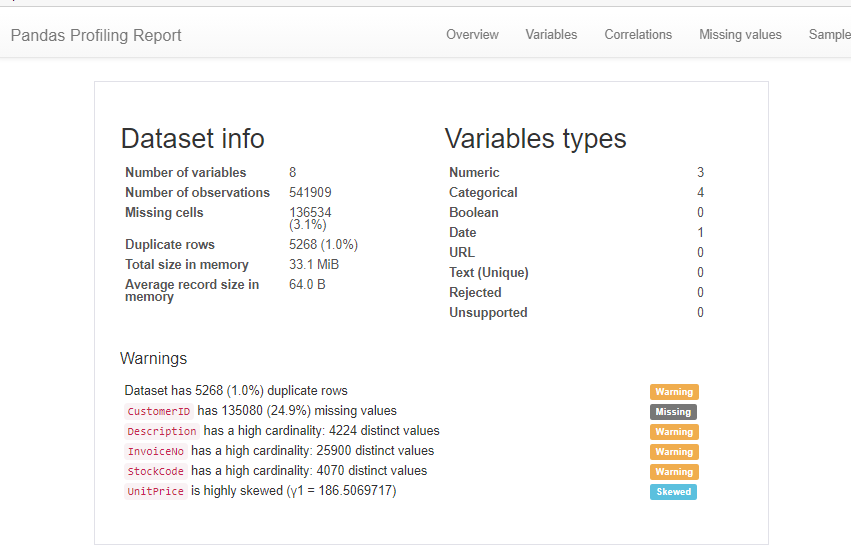


Figure 5.1: Description of the dataset

The dataset has 541909 rows and 8 independent variables with 3 Numerical variables and 4 categorical fields. 3.1% of the dataset has missing values and 1% of the data is duplicate values, of which are dropped when cleaning the data. There are 3664 unique products in the dataset with 4338 unique customers.

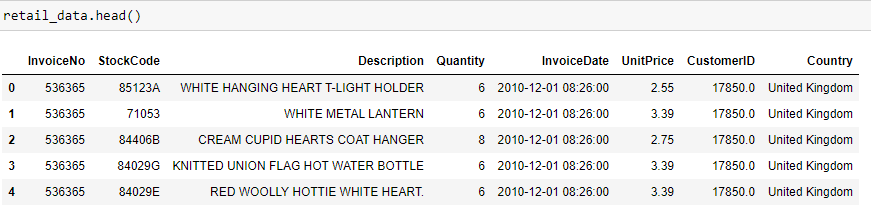


Figure 5.2: Sample Dataset

### 5.2.1 Data Pre-processing and Feature extraction

In preparation of the dataset to be used for the recommendation model, Missing values were removed, grouped purchase quantities together by productid and CustomerId, Change any sums of quantity that equal zero to one (this happened if items were returned, thus indicating a negative quantity value, but the interaction between the user and the Item needs to be indicated as opposed to assuming no interaction that ever took place)

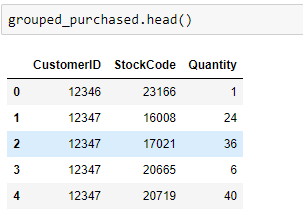


Figure 5.3:Resulting matrix of grouped purchases

The purchase quantity represents a “confidence” in terms of how strong the interaction was. Items with a larger number of purchases by a customer can carry more weight in our ratings matrix of purchases.

The last step of preprocessing involves creating the sparse ratings matrix of users and items. This step is especially important to prevent unnecessary memory issues. The matrix contains thousands of items and thousands of users with a user/item value required for every possible combination. Thus resulting to a very large matrix, so we can save a lot of memory by keeping the matrix sparse and only saving the locations and values of items that are not zero.

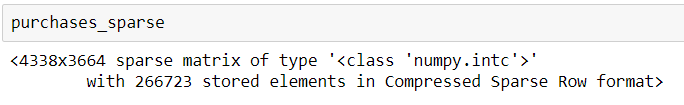
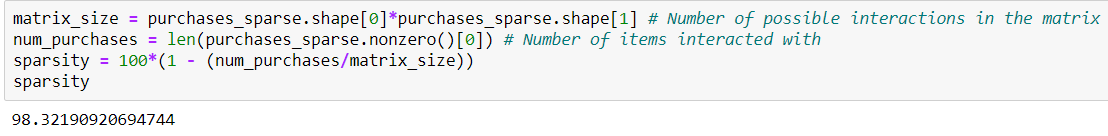


Figure 5.4:Sparse Matrix

There are 4338 customers with 3664 items. For these user/item interactions, 266723 of these items had a purchase. In terms of sparsity of the matrix, that makes:

Figure 5.5:sparsity of the matrix

98.3% of the interaction matrix is sparse.

### 5.2.2 Creating a Training and Validation Set

Typically, in Machine Learning applications, there is a need to test whether the model trained is efficient even when new data it hasn’t yet seen before from the training phase is used. A test set completely separate from the training set is therefore created by taking a random sample of the training data rows in our feature matrix and separate it away from the training set. That normally looks like this:

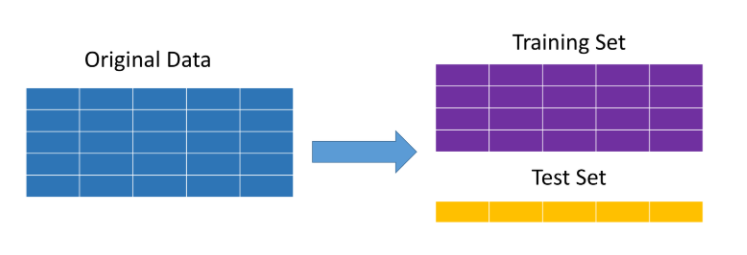


Figure 5.6: Test and Training creation visualization

The method that is used in the implementation of this project is masking. This is hiding a percentage of the user/item interactions from the model during the training phase chosen at random. Then, check during the test phase how many of the items that were recommended the user actually ended up purchasing in the end.

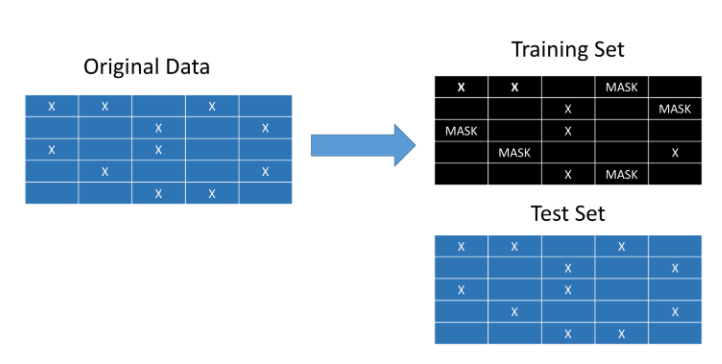


Figure 5.7: Masking for creating test dataset

The test set is an exact copy of the original data. The training set, however, will mask a random percentage of user/item interactions and act as if the user never purchased the item (making it a sparse entry with a zero). A check in the test set is then performed to identify which items were recommended to the user that they ended up actually purchasing. If the users frequently ended up purchasing the items most recommended to them by the system, we can conclude the system seems to be working.



Figure 5.8: Code for generating train and test data

This will return the training set, a test set that has been binarized to 0/1 for purchased/not purchased, and a list of which users had at least one item masked. The performance of the recommender system will be tested on these users only. Only 20% of the user/item interactions is masked in this project.

## 5.3 Model Training and Testing

### 5.3.1 Imported Libraries

In the implementation of the project. Several libraries were imported. These include:- pandas and numpy for data manipulation, scipy.sparse, scipy.sparse.linalg for generation of the sparse matrix, pandas\_profiling for report generation, random for picking random interactions for the test dataset, implicit for speeding up ALS matrix factorization, sklearn specifically metrics for evaluating the recommendation engine and sklearn.preprocessing more specifically MinMaxScaler

### 5.3.2 **Implementing Alternating Least Squares(ALS) for Implicit Feedback**

After the ratings matrix which is sparse (represented by the product\_train sparse matrix object) is created. It is turned into a confidence matrix.

Cui=1+αrui

Where Cui is the confidence matrix for our users *u* and our items *i*. The α term represents a linear scaling of the rating preferences (in our case number of purchases) and the rui term is our original matrix of purchases.

The minimization of the cost function for our users U is there after done using:

xu=( YTCuY + λI )−1 YTCup(u)

This computation can be speeded up through some linear algebra that changes this equation to:

xu=(YTY+YT(Cu−I) Y+ λI)−1 YTCup(u)

Notice that we can now precompute the YTY portion without having to iterate through each user u. We can derive a similar equation for our items:

yi = (XTX + XT (Ci−I) X + λI)−1 XTCip(i)

These will be the two equations are iterated back and forth between until they converge. We also have a regularization term λ that helped prevent overfitting during the training stage as well, along with our binarized preference matrix p which is just 1 where there was a purchase (or interaction) and zero where there was not. This calculation is converted into a function in order to utilize the alternating least squares in our project.

Figure 5.9 illustrates a single iteration of the code when it is run, to see how it works. 20 latent factors as the rank matrix size is chosen along with an alpha of 15 and regularization of 0.1



Figure 5.9:Alternating Least Squares

The ratings for a particular user can be investigated by taking the dot product between the user and item vectors (U and V).

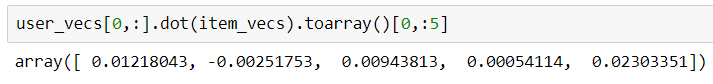


Figure 5.10:Ratings for a particular user

This is a sample of the first five items out of the 3664 in our stock. The first user in our matrix has the fifth item with the greatest recommendation out of the first five items.

### 5.3.3 Evaluating the Recommender System

The training set had 20% of the purchases masked. This will allowed us to evaluate the performance of the recommender system. Essentially, we needed to see if the order of recommendations given for each user matches the items they ended up purchasing.

A commonly used metric for this kind of problem is the area under the Receiver Operating Characteristic (or ROC) curve. A greater area under the curve means we are recommending items that end up being purchased near the top of the list of recommended items. Usually, this metric is used in more typical binary classification problems to identify how well a model can predict a positive example vs. a negative one. It was very helpful when attempting to rank recommendations.

A function that calculated a mean area under the curve (AUC) for any user that had at least one masked item was used. As a benchmark, The mean AUC was calculated as it would have been used if we were simply recommending the most popular items. Popularity tends to be hard to beat in most recommender system problems, so it makes a good comparison.

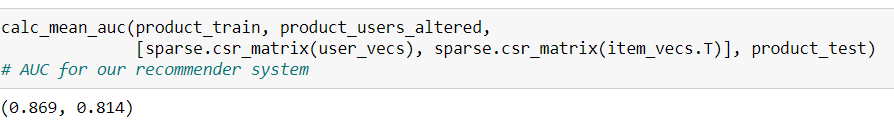


Figure 5.11: Mean AUC for model evaluation

The trained recommender system beat popularity. The system had a mean AUC of 0.87, while the popular item benchmark had a lower AUC of 0.814. An AUC of 0.87 means the system is recommending items the user in fact had purchased in the test set far more frequently than items the user never ended up purchasing.

## 5.4 System Testing

System Testing is the process or method of finding errors in an application or program so that the application functions according to the end user's requirement. It is an important engineering activity responsible for a significant portion of the costs of developing and maintaining software. The following testing types where used in the testing of my system.

### 5.4.1 Unit Testing

I performed unit testing on the individual functions of the system such as making sure that whenever users registered on the form, their details are correctly stored in the database. This also helped me ensure that there was correct validation during user log in such that only registered users can log in.

I also tested session activity in order to ensure that all active user sessions were being started and destroyed whenever necessary.

### 5.4.2 Performance Testing

During this phase of testing I wanted to ensure that the system is able perform all its tasks and functionalities correctly as required to from start to end of a session. I also corrected any errors that may have been brought up due to missing parts of code or wrongly coded parts. Through the help of my colleagues, I also tested how easy it would be for a user and an administrator to access the services being offered by the system.

### 5.4.3 Integration Testing

This phase of testing was mainly to ensure that all the units of my system worked properly together as one. I tested all the links to the different pages of the system and ensured a consistency in design of the various webpages. Any redirections I missed I was able to rectify at this phase.

### 5.4.4 Data Flow Testing

This testing technique was used to determine whether data was being sent to the database and whether the user was receiving the data. The test was also done to determine whether retrieval of data was a smooth process.

### 5.4.5 Black-Box Testing

This is the technique of testing without having any knowledge of the interior workings of the application. The tester is oblivious to the system architecture and does not have access to the source code. The main goal of this testing type is to check whether everything works well from the point of view of an ordinary user. Typically, when performing the black-box test, the tester interacted with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

#### 5.4.5.1 Sample standard Recommend gifts Test case

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Test Steps** | **Test Data** | **Expected Results** | **Actual Results** | **Pass/ Fail** |
| 01 | In the User module Recommend gift for an existing user in the system. | 1. Go to the main page 2. Click on Login 3. Enter a valid email. 4. Enter the password. 5. Click Login. 6. Go to recommend gift 7. Input email of existing user | Email=rosanneodiero4@gmail.com  Password = Serenity76!   * + - 1. Recipients email= taylor99@gmail.com | The User should be given recommendations for the user with the email, taylor99@gmal.com | As Expected | Pass |
| 02 | In the User module Recommend gift for a user that doesn’t exist in the system. | 1. Go to the main page 2. Click on Login 3. Enter a valid email. 4. Enter the password. 5. Click Login. 6. Go to recommend gift 7. Input email of a non existant user | Email=rosanneodiero4@gmail.com  Password = Serenity76!   * + - 1. Recipients email= taylor@gmail.com | User should be unable to get any recommendations from the application.  The user should be alerted that user does not exist. | As Expected | Pass |

Figure 5.12:Sample recommendation test case

### 5.4.5.2 Sample standard login Test case

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Test Steps** | **Test Data** | **Expected Results** | **Actual Results** | **Pass/Fail** |
| 01 | In the User module login to the system with valid data | 1. Go to the main page 2. Click on Login 3. Enter a valid email. 4. Enter the password. 5. Click Login. | Email=rosanneodiero4@gmail.com  Password = Serenity76! | User should be able login to the application. | As Expected | Pass |
| 02 | In the User module login to the system with invalid data | 1. Go to the main page 2. Click on Login 3. Enter a valid email. 4. Enter the password. 5. Click Login. | Email =rosanne@gmail.com  Password = qwerty | User should be unable to login to the application. | As Expected | Pass |

Figure 5.13 Sample Standard Login Test Case

### 5.4.3 White-Box Testing

The core idea of this approach to software testing is taking a look at the internal structure design and at the code of the program to test it. In White box testing, the tester can see the entire code of the program and he is tasked to verify the flow of how inputs and outputs work in the program. Unlike black box testing, which is more focused on testing the functionality of the program, it is concerned with testing the internal structures of the program. Taking a look at the program in this way allows us to work on improving the design, usability and making the product more secure.

# : Discussions

## 6.1 Introduction

This chapter discusses the results of my test data by explaining why the result reflects the recommendation model is working. It also highlights how the objectives stated in chapter one has been achieved after the conclusion of the research.

## 6.2 Achievement of Objectives

The five objectives that are specified in chapter one where successfully accomplished.

### 6.2.1 Current method people use to select a gift for their loved ones.

To determine the current method people, use when selecting a gift, I conducted a phone interview with the respondents being my classmates and asked around fifty of them this particular question. I had a variety of responses but the main response included:- Asking the recipient for what they would love for a gift, Guessing an ideal gift based on how they think they know the recipient, giving the recipient money or a gift voucher so that the recipient purchases his/her own gift or even just hand make the gift for sentimental recipients.

### 6.2.2 Challenges that people undergo when choosing a gift.

The challenges these respondents undergo when purchasing the gift included but not limited to:- they get anxious that the respondent might not love the gift thus weakening their friendship or relationship, they identify a gift that the recipient might appreciate but its not within the budget or even they don’t know what to buy entirely so they don’t purchase any gift.

### 6.2.3 Types of recommendation systems and how they work.

In chapter two, this objective was achieved where I explained all the types of recommendations and how they work and their evaluation metrics

### 6.2.4 Existing gift selection applications and their short comings.

The existing gift selection application in Kenya includes purpink where the recommendations are very generalised thus beating the purpose of the suggestions for the gift giver.

### 6.2.5 Build and Test the proposed solution

The proposed solution was built using alternating least squares which is basically a matrix factorization model for implicit feedback thus getting personalized recommendations based on what the user purchased. The model was tested using Area under the curve, and benchmarked using mean area under the curve, which is for identifying the most popular product and our model performed better.

## 6.3 Discussion of the results of testing

As mentioned in the earlier chapters the method used was matrix factorization, specifically using alternative least square. The reason for using this type of matrix factorization as opposed to the standard matrix factorization i.e. singular value decomposition (SVD) is that, SVD would require inverting a potentially very large matrix, which could be computationally expensive. Thus, to solve the factorization more precisely we applied ALS to approximate it. In the case of ALS, we only need to solve one feature vector at a time, which means it can be run in parallel. Thus, less computationally expensive.

### 6.3.1 Explanation of the test result

Figure 6.1 is the output for the item purchased by a user

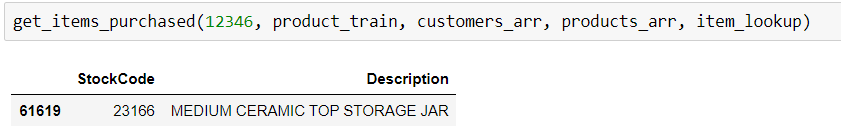


Figure 6.1: Purchased Item by user 12346

Figure 6.2 is the output of the item recommended for a user, based on the item purchased.

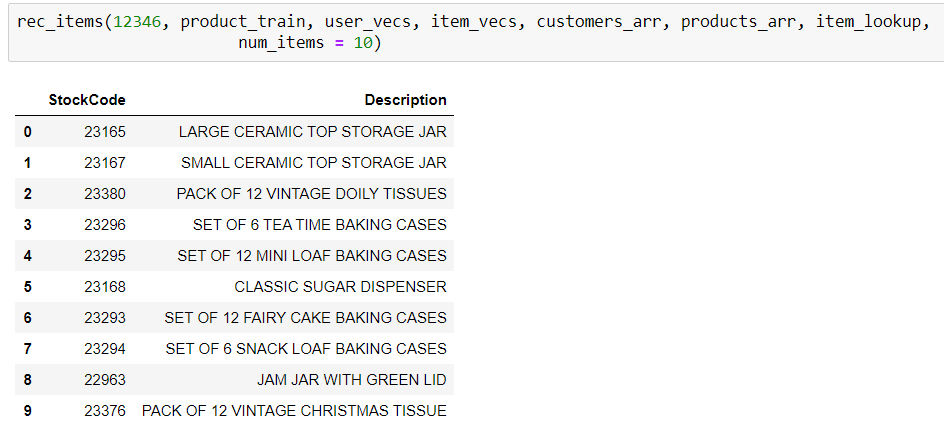


Figure 6.2:Recommended Product for user 12346

The result for the test case recommendation is ‘as expected’ because these recommendations seem quite good! Remember that the recommendation system has no real understanding of what a ceramic jar is. All it knows is the purchase history. It identified that people purchasing a medium sized jar may also want to purchase jars of a differing size. The recommender system also suggests jam jar and a sugar dispenser, which is similar in use to a storage jar. I personally was blown away by how well the system seems to pick up on these sorts of shopping patterns.

# : Conclusion, Recommendation and Future Works

## 7.1 Conclusion

The proposed system implementation will certainly reduce the time it takes to choose an ideal gift, it will make the process easier and ultimately strengthen the bonds between gift givers and recipients.

## 7.2 Challenges faced

The main challenge faced when developing the system, is lack of adequate data. Ideally, it would be best if we had more demographic data, and the items that a user adds to cart but does not purchase or a product that a user views. This would certainly provide more personalized recommendations.

## 7.3 Recommendations

To run this web application, one needs a laptop, phone or tablet that has access to the internet.

## 7.4 Future Works

If I had more time to develop the project I would periodically collect the data from users in order to get more features for the user recommendations.

# : References

AI gifting ideas for the Holiday season [WWW Document], 2020. . DefinedCrowd®. URL https://www.definedcrowd.com/ai-gifting-ideas-for-the-holiday-season-2/ (accessed 6.21.20).

Belk, R.W., n.d. Gift-giving behavior. 1977 96.

Can AI Finally Help Humans Choose the Right Gift? [WWW Document], n.d. . Huge. URL http://express.hugeinc.com/articles/can-ai-finally-help-humans-choose-the-right-gift (accessed 6.20.20).

CNN, J.P., n.d. 5 rules for successful gift giving [WWW Document]. CNN. URL https://www.cnn.com/style/article/rules-of-gift-giving-scn/index.html (accessed 6.13.20).

Data Collection [WWW Document], n.d. URL https://ori.hhs.gov/education/products/n\_illinois\_u/datamanagement/dctopic.html (accessed 10.11.20).

Functional and Nonfunctional Requirements: Specification and Types, n.d. . AltexSoft. URL https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/ (accessed 2.5.21).

Garcia, A.D., n.d. Holiday Brings Spending Stress For Most Americans, Survey Reveals [WWW Document]. Bankrate. URL https://www.bankrate.com/surveys/holiday-gifting-november-2019/ (accessed 6.14.20).

Gift-giving Anxiety [WWW Document], n.d. . Psychol. Today. URL http://www.psychologytoday.com/blog/presence-mind/201412/gift-giving-anxiety (accessed 6.20.20).

Holiday Stress | Managing Holiday Stress | Stress During the Holidays, 2014. . AllOne Health. URL https://allonehealth.com/holiday-stress-guide/ (accessed 6.14.20).

How do Recommendation Engines work? And What are the Benefits?, 2017. . Maruti Techlabs. URL https://marutitech.com/recommendation-engine-benefits/ (accessed 6.21.20).

India, S., 2019. How Netflix’s Recommendation Engine Works? [WWW Document]. Medium. URL https://medium.com/@springboard\_ind/how-netflixs-recommendation-engine-works-bd1ee381bf81 (accessed 6.22.20).

Introduction to Recommender Systems in 2019 | Tryolabs Blog [WWW Document], n.d. URL https://tryolabs.com/blog/introduction-to-recommender-systems/ (accessed 6.21.20).

Isinkaye, F.O., Folajimi, Y.O., Ojokoh, B.A., 2015. Recommendation systems: Principles, methods and evaluation. Egypt. Inform. J. 16, 261–273. https://doi.org/10.1016/j.eij.2015.06.005

Knowledge-based recommender systems - Building a Recommendation System with R, 2015.

Labaree, R.V., n.d. Research Guides: Organizing Your Social Sciences Research Paper: Types of Research Designs [WWW Document]. URL https://libguides.usc.edu/writingguide/researchdesigns (accessed 10.11.20).

M, M., 2019. Introduction to recommendation systems and How to design Recommendation system,that resembling the… [WWW Document]. Medium. URL https://medium.com/@madasamy/introduction-to-recommendation-systems-and-how-to-design-recommendation-system-that-resembling-the-9ac167e30e95 (accessed 6.21.20).

Machine Learning for Recommender Systems - A Primer [WWW Document], n.d. URL https://www.dasca.org/world-of-big-data/article/machine-learning-for-recommender-systems-a-primer (accessed 6.23.20).

Madadipouya, K., Chelliah, S., 2017. A Literature Review on Recommender Systems Algorithms, Techniques and Evaluations 8, 16.

Model-based Collaborative Filtering Algorithms. [WWW Document], n.d. URL http://www10.org/cdrom/papers/519/node8.html (accessed 6.21.20).

O’Carroll, B., 2020. What is Rapid Application Development (RAD)? [WWW Document]. Codebots. URL https://codebots.com/app-development/what-is-rapid-application-development-rad (accessed 10.11.20).

Park, D.H., Kim, H.K., n.d. A Literature Review and Classification of Recommender Systems on Academic Journals 14.

Pereira, C. de P., Parente da Costa, R., Dias Canedo, E., 2017. Mobile Gift Recommendation Algorithm:, in: Proceedings of the 19th International Conference on Enterprise Information Systems. Presented at the 19th International Conference on Enterprise Information Systems, SCITEPRESS - Science and Technology Publications, Porto, Portugal, pp. 565–573. https://doi.org/10.5220/0006330405650573

Porta, S., 2020. Recommendation System using kNN [WWW Document]. Auriga IT. URL https://www.aurigait.com/recommendation-system-using-knn/ (accessed 6.23.20).

S, P., D, M.H., 2017. Recommendation Systems for E-Commerce: A Review. IJARCCE 6, 500–504. https://doi.org/10.17148/IJARCCE.2017.6496

Smyth, B., Cotter, P., 2000. A personalised TV listings service for the digital TV age. Knowl.-Based Syst. 13, 53–59. https://doi.org/10.1016/S0950-7051(00)00046-0

System Requirements [WWW Document], n.d. URL https://www.umsl.edu/~sauterv/analysis/F2015/System%20Requirements.html.htm (accessed 1.21.21).

Underwood, C., n.d. Use Cases of Recommendation Systems in Business – Current Applications and Methods [WWW Document]. Emerj. URL https://emerj.com/ai-sector-overviews/use-cases-recommendation-systems/ (accessed 6.22.20).

What is Rapid Application Development, and Is It Right for My Business? [WWW Document], 2020. . Skelet. Key. URL https://skeletonkey.com/what-is-rapid-application-development-and-is-it-right-for-my-business/ (accessed 10.10.20).

# Appendix

## Gantt Chart

