# AI BOT FOR IMAGE-BASED PRODUCT RECOMMENDATION

SHASHANK PRASAD

Final Thesis Report

DECEMBER 2022

# DEDICATION

I dedicate this thesis to my parents, who have always been my biggest supporters and the driving force behind my ambition and determination. Their love, guidance, and encouragement have been the foundation of my success, and I am forever grateful for their unwavering support.

I also dedicate this work to my siblings, friends, who have always been there for me and provided me with love, laughter, and joy throughout my life. Their presence in my life has been a constant source of happiness and support, and I am grateful for their love and friendship.

Finally, I dedicate this thesis to my partner, who has been a constant source of love, support, and encouragement throughout this journey. Their patience, understanding, and unwavering belief in me have been a driving force behind my success, and I am forever grateful for their presence in my life. Thank you all for your love and support. This work is dedicated to you.

# ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to my supervisor, for their invaluable guidance, support, and encouragement throughout my research and writing process. Their expertise and expertise in the field have been an invaluable resource, and I am deeply grateful for their guidance and support.

I would also like to thank my colleagues and fellow students at LJMU, upGrad for their support and friendship throughout this process. I am grateful for the stimulating discussions and feedback that have helped to shape my thinking and my research.

Finally, I would like to thank my family and friends for their unwavering support and encouragement throughout this journey. Their love and support have been a constant source of motivation and strength, and I am deeply grateful for their presence in my life.

Thank you all for your support and guidance. Without your help, this work would not have been possible.

**ABSTRACT**

The online shopping industry is booming day by day, and they largely depend on text filtering to find the required product. And for customers, it is also very inconvenient to download multiple applications and search for desired products manually. This is a research implementation where we will address and propose a new way to filter products via image filtering and recommend related products to customers through an instant messaging application like Telegram and WhatsApp.

In this research, to interact with the instant messaging application we will create a bot and this bot will call a recommendation system to get the recommended product for the given product image provided by the customer/user.

In this recommendation system, first, we will classify and categorize the product by the image classification model. This image classification model will be just the CNN model which will take the input image and categorize the product it belongs to. In the CNN model, we will also explore different CNN architecture/models like SVM (Support Vector Machine), VggNet, GoogLeNet, ResNet, Xception and validate their training, validation, and test accuracy.

In the second part of the recommendation engine, we help to identify the recommended product based on the input image, for which we will calculate similarities between the feature extracted from the pre-trained model and image. Then, images are suggested based on the calculated distance provided by KNeighbors classifier, listed in descending order. We will also explore recommendation by product name or title for which we will calculate cosine similarity between the product tile. So, higher the cosine similarity, the recommended product will be more similar to the input title searched.

## TABLE OF CONTENTS


DEDICATION ........ I

ACKNOWLEDGEMENTS ........ II

ABSTRACT ........ III

LIST OF TABLES ........ VII

LIST OF FIGURES ........ VIII

LIST OF ABBREVIATIONS ........ X

CHAPTER 1: INTRODUCTION ........ 1

1.1 Background of Study ........ 1

1.2 Problem Statement ........ 3

1.3 Aim and Objectives ........ 3

1.4 Scope of the Study ........ 4

1.5 Significance of the Study ........ 5

1.6 Structure of Study ........ 6

CHAPTER 2: LITERATURE REVIEW ........ 8

2.1 Introduction ........ 8

2.2 Related Research ........ 9

2.3 Research gap ........ 10

2.4 Recommendation System ........ 11

2.4.1 History of Recommendation System ........ 13

2.4.2 Application of Recommender System ........ 14

2.4.3 Types of Recommender System ........ 17

2.5 Challenges in recommender system ........ 20

2.6 Comparison of Techniques ........ 22

2.7 Discussion ........ 22

2.8 Summary ........ 23

CHAPTER 3: RESEARCH METHODOLOGY ........ 24
3.1 Introduction ........ 24
3.2 Methodology ........ 24
3.2.1 Convolution Neural Networks ........ 28
3.2.2 Classification Algorithm and Types of Classification ........ 36
3.2.3 Background Removal ........ 37
3.3 Approach ........ 38
3.4 Required Resources ........ 38
3.5 Summary ........ 40
CHAPTER 4: ANALYSIS ........ 41
4.1 Introduction ........ 41
4.2 Data Sourcing and understanding ........ 41
4.3 Data Preparation ........ 43
4.3.1 Data Cleaning ........ 44
4.3.2 Feature Selection ........ 46
4.3.3 Data Transformation ........ 47
4.3.4 Feature Engineering ........ 48
4.3.5 Dimensionality Reduction ........ 49
4.4 Data Visualization ........ 50
4.4.1 Numerical Data Visualization ........ 51
4.4.2 Categorical Data Visualization ........ 54
4.4.3 Word Cloud ........ 55
4.5 Summary ........ 56
CHAPTER 5: IMPLEMENTATION, EXPERIMENTS AND RESULT EVALUATION ........ 58
5.1 Introduction ........ 58
5.2 Image Classification using CNN ........ 58
5.3 Text Based Recommendation ........ 65

5.4 Models of Recommendation System ... 67
5.4.1 ResNet-50 ... 68
5.4.2 GoogLeNet ... 69
5.4.3 VGG16 ... 71
5.4.4 Xception ... 73
5.5 Image Based Recommendation System ... 75
5.6 Evaluating the Image Recommendation Model ... 78
5.7 Recommendation system with telegram bot. ... 80
5.8 Summary ... 82
CHAPTER 6: CONCLUSIONS AND RECOMMENDATION ... 84
6.1 Introduction ... 84
6.2 Conclusion ... 84
6.3 Future Recommendations ... 86
6.4 Challenges ... 86
6.5 Summary ... 87
REFERENCES ... 89
APPENDIX A: RESEARCH PROPOSAL ... 92

## LIST OF TABLES


TABLE 2. 1: COMPARISON BETWEEN RECOMMENDATION SYSTEMS. .......... 22

TABLE 4. 1:UNIQUE VALUE, DATA TYPE, AND NUMBER OF NULL VALUES OF EACH LABEL .......... 45
TABLE 4. 2: DATA TYPE OF ALL LABEL IN DATASET .......... 48
TABLE 4. 3 DETAIL INFORMATION OF VALUES IN PRICE AND RATING COLUMN .......... 52

TABLE 5. 1: CNN ACCURACY AND LOSS .......... 62
TABLE 5. 2: CLASSIFICATION REPORT FOR CNN MODEL .......... 64
TABLE 5. 3: RESNET-50 ACCURACY AND LOSS .......... 68
TABLE 5. 4: GOOGLENET ACCURACY AND LOSS .......... 70
TABLE 5. 5: VGG16 ACCURACY AND LOSS .......... 72
TABLE 5. 6: XCEPTION MODEL ACCURACY AND LOSS .......... 74
TABLE 5. 7: COMPARING ACCURACY AND LOSS ACROSS ALL MODELS. .......... 83

TABLE 6. 1: ACCURACY LOSS OF VGG16 MODEL .......... 85

## LIST OF FIGURES

Figure 1. 1: Global E-Commerce Market 2019-2025 (Ref: (Thakur, 2021))........................................1
Figure 1. 2: Users in Instant Messaging application. ...........................................................................2
Figure 1. 3: Types of Recommendation System.......................................................................................6

Figure 2. 1: Timeline review of recommendation systems (Ref: (Taneja and Arora, 2018))..........14
Figure 2. 2: Recommendation System and their types ........................................................................23

Figure 3. 1: Overview for implementing image-based recommendation system .............................24
Figure 3. 2: Example of CNN Architecture .........................................................................................26
Figure 3. 3: Illustration of similar product tends to high cosine value or group together. ......27
Figure 3. 4: Simple diagram of neural network ....................................................................................29
Figure 3. 5: Architecture showing different CNN layers ...............................................................29
Figure 3. 6: LeNet-5 CNNs Architecture ...........................................................................................31
Figure 3. 7: AlexNet CNNS architecture for image photo classification (Ref: (Gonzalez, 2007))...........................................................................................................................................32
Figure 3. 8: VGGNet Architecture (Ref: (Simonyan and Zisserman, 2015))....................................33
Figure 3. 9: VGGNeT 3x3 and 5x5 Kernel.........................................................................................34
Figure 3. 10: Example Inception Model of GoogleNet (Ref: (Zeng et al., 2016) )............................34
Figure 3. 11: Residual Network..........................................................................................................35
Figure 3. 12: Recommendation engine flowchart ...............................................................................38

Figure 4. 1: Category of data visualization........................................................................................51
Figure 4. 2: Density per rating of product in dataset........................................................................52
Figure 4. 3 : Density per price of product in dataset .......................................................................53
Figure 4. 4: Count of products in each category. ..............................................................................54
Figure 4. 5: Sample Products in each category..................................................................................55
Figure 4. 6 : WordCloud of product title and product category .....................................................56

Figure 5. 1: CNN Model Accuracy. ......................................................................................................62

FIGURE 5. 2: CNN MODEL LOSS ........................................ 63
FIGURE 5. 3: CONFUSION MATRIX ON TEST DATA ........................................ 63
FIGURE 5. 4: PREDICTED VS ACTUAL CATEGORY ON RANDOM 8 IMAGES ........................................ 65
FIGURE 5. 5: RESNET-50 MODEL ACCURACY. ........................................ 68
FIGURE 5. 6: RESNET-50 MODEL LOSS. ........................................ 69
FIGURE 5. 7: GOOGLENET MODEL ACCURACY. ........................................ 70
FIGURE 5. 8: GOOGLENET MODEL LOSS. ........................................ 71
FIGURE 5. 9: VGG16 MODEL ACCURACY. ........................................ 72
FIGURE 5. 10: VGG16 MODEL LOSS. ........................................ 73
FIGURE 5. 11: XCEPTION MODEL ACCURACY. ........................................ 74
FIGURE 5. 12: XCEPTION MODEL LOSS. ........................................ 75
FIGURE 5. 13: OUTPUT OF IMAGE BASED RECOMMENDATION SYSTEM ........................................ 80
FIGURE 5. 14: STEPS TO ACCESS PRODUCT RECOMMENDATION BOT. ........................................ 81
FIGURE 5. 15: TOP 5 RECOMMENDATION PROVIDED BY RECOMMENDATION BOT. ........................................ 82

FIGURE 6. 1: OUTPUT OF IMAGE RECOMMENDATION SYSTEM ........................................ 85

# LIST OF ABBREVIATIONS

The list below describes abbreviations that are used in this thesis:

| ABBREVIATIONS | MEANING |
|---|---|
| RS | Recommendation System |
| RSs | Recommendation Systems |
| CNN | Convolution Neural Network |
| CNNs | Convolution Neural Networks |
| ANN | Artificial neural network |
| SVM | Support vector machine |
| CF | Collaborative filtering |
| CB | Content-based filtering |
| HF | Hybrid filtering |
| ResNet | Residual Networks |
| VGG | Visual Geometry Group |
| RGB | Red, Green, Blue |
| RF | Random Forest |
| HSV | Hue, Saturation and Value |

## CHAPTER 1: INTRODUCTION

### 1.1 Background of Study

In recent few years, we have seen growth in online shopping and most of businesses are trying to establish their presence online by creating their applications and selling most the products at their online store to conduct ecommerce marketing.

According to research in 2021, global retail online sales will surpass US $15 trillion in 2023 and by 2025 it is expected to cross US $20 trillion, despite slow growth. Below figure 1.1 (Thakur, 2021) showing the trends of global sales in ecommerce industry.

.

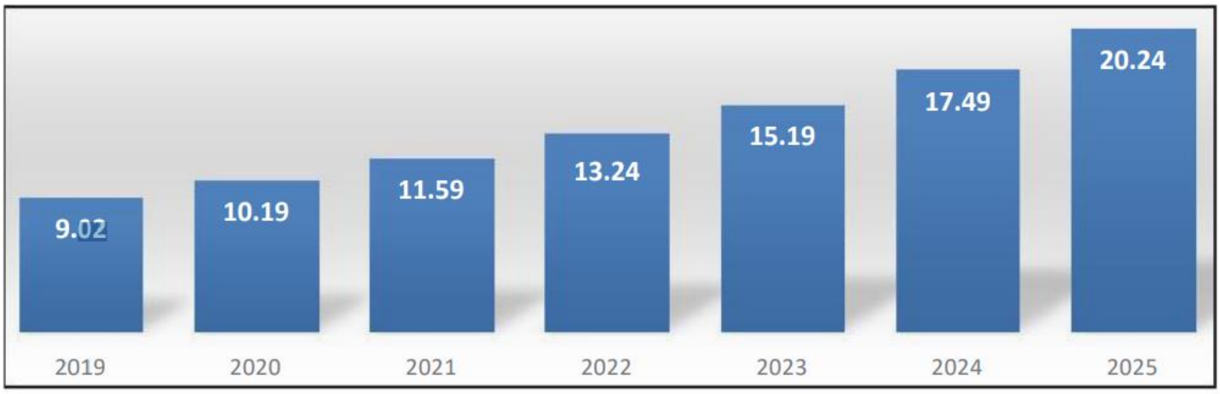


Figure 1. 1: Global E-Commerce Market 2019-2025 (Ref: (Thakur, 2021))

As online shopping is becoming more and more popular due to its convenience. However, to find the right product, the user needs to download multiple applications and search the resided product, filer it and select the right product. This is time-consuming and with increase in mobile applications it becomes a very tedious task for the user.

The other industry which has seen growth in mobile application area is instant messaging application like WhatsApp and Telegram, where from figure 1.2 we can see alone WhatsApp has 2 billion active users and is the most popular messaging service in over 100 countries.

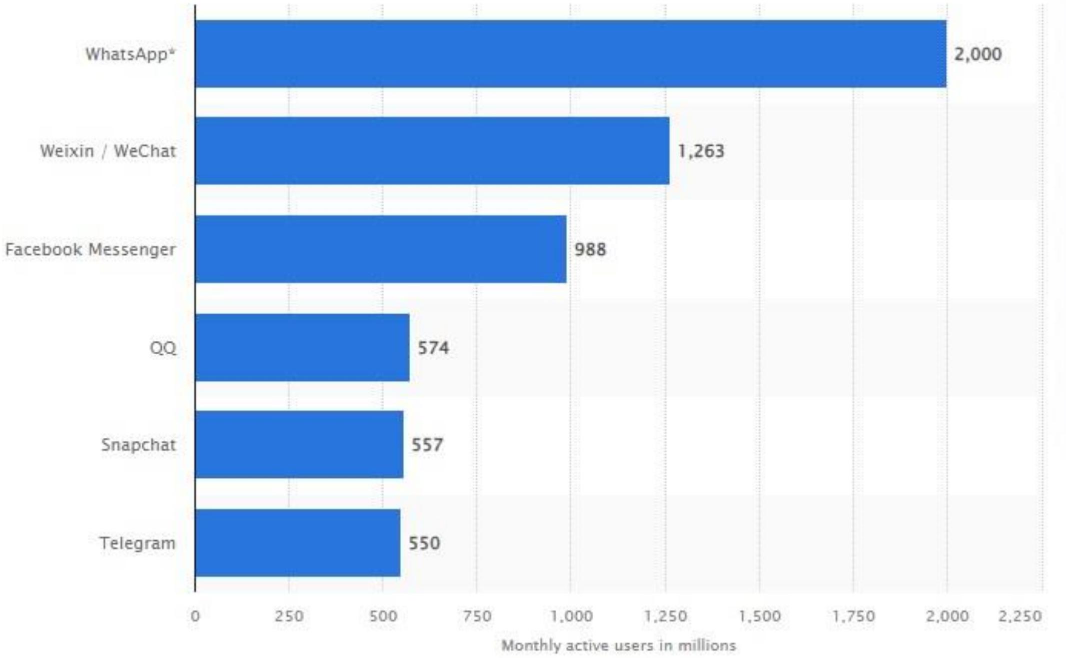


Figure 1. 2: Users in Instant Messaging application.

One solution to solve this problem is to provide user more intuitive way to find relevant products by instant messing application like WhatsApp or Telegram where user can get the list of recommended products based on given image. In other words, An AI bot will help user to identify and recommend online product based on user image for which use can use instant messaging application.

Based on this idea, we will create a bot and run it inside an instant messaging application. User can interact with the bots through the instant messaging application and the bot will return the recommended based on user input. So, the bot here basically would be an image-based recommendation system that will take an image as input. Then this image input would be passed through the CNN model (Convolutional Neural Network) which will help to categorically classify the image to the object it might probably belong. Then the identified product or item is passed as input to the recommendation system. The recommendation system will use the product id or any id and will recommend the suitable product using content-based filtering i.e., products having the same content as the given input image to the recommendation system will be recommended and the user will be able to see the recommended product/ products in the instant messaging application.

## 1.2 Problem Statement

Most of the ecommerce website / application like Amazon, Flipkart largely depends on text filtering to find the required product. And for customer it is inconvenient to download multiple application and search for desired product manually. This is very time consuming and can result in selection of wrong product with inferior quality and high price.

So, "How can we effectively use visual information from images to make personalized recommendations to users?” Is the main focus of the thesis, which is to explore ways to use visual information from images to make more effective recommendations to users. It highlights the need for effective methods for utilizing visual data in recommendation systems, and suggests that this is an important area of research.

In this thesis, we are proposing a new way to filter product via image and recommend related products to customers through instant messaging applications like telegram and WhatsApp. Image base recommendations will help user to filter exact or similar products which is not done in the previous research (Chen et al., 2015)(Ullah et al., 2020)(Jouyandeh and Zadeh, 2022)(Elsayed et al., 2022)(Rashed et al., 2022).

## 1.3 Aim and Objectives

In this thesis, we aim to address the recommendation system problem by analysing the feature of the image itself rather than the rating provided by the seller. Basically, we aim to create a recommendation system where customers tend to prefer products with visual similar attributes, such as shape, size, colour, style, or pattern.

To connect to this recommendation system, customer can use instant messaging platforms like WhatsApp or Telegram where the user/customer can upload image of prefer products and through our recommendation system they will get the desired link/URL of the product from various e-commerce websites like Amazon.

To develop an image-based product recommendation system used primarily in the e-commerce domain. This engine will take image of certain product as input via instant messaging application and get the recommendation of similar products that resemble closely to the input image.

We aim to achieve this goal by implementing a combination of dimensionality reduction, clustering, modelling, and recommendation system techniques. And integrating whole recommendation engine on real time with instant messaging application.

Objective of this research is:

- To recommend the product to customers based on visual/feature and similarity.
- To identify the best model to achieve high accuracy in recommendation using image-based recommendation system for e-commerce like Amazon.
- To run recommendation system as a bot on instant messaging applications.

## 1.4 Scope of the Study

In Score of this thesis, we will:

- Recommendation to the user will be provided based on user input image.
- Recommendation system can work with and without bot i.e., WhatsApp bot or Telegram bot.
- Recommendation system can work with either WhatsApp bot or Telegram bot.
- Since we are dealing with e-commerce data, on which there are tons of data available on internet. So, with respect to this thesis we will collect data for Electronics, Pet Supplies, Home and Kitchen, Cell Phones and Accessories, Clothing Men, Clothing Women, Watches and Shoes.
- Compare different model and identify which model gives us best result and build image recommendation system on that model.

Out of Scope in this thesis:

- User should have smartphone with active phone number and instant messaging applications like WhatsApp or Telegram installed.
- Connection to the bot via instant messaging application should be done manually and will be out of scope of recommendation system.

- Telegram bot is intended for testing purpose and not for production usage.
- Product with English description will be considered for the research and other languages will be out of scope of the research.
- Bot will run on local system and hosting of bot would be out of scope of the thesis.

### 1.5 Significance of the Study

In modern world, the recommendation system plays a key role in defining business. It always has been in demand in every domain and in every industry consider it be in food industry, dating business, online gaming, online streaming media, metaverse or the e-commerce industry.

Personalized product recommendation helps several eCommerce businesses suggest relevant products to their end-users at multiple touchpoints. Quick and instant suggestions will make every user feel valued and give them a personalized shopping experience. Recommendations help the eCommerce industry offer a personalized shopping experience to the end-user and thus witness a rapid boost in user engagement and revenue flow.

Recommendation systems are of different types:

1. Collaborative filtering –

   Collaborative filtering focusses more on gathering, filtering and analysing data based on user's behaviour. For example, if user x likes pizza, pasta, Italian bread, and user y likes pasta, pasta, mac and cheese. So, it would be likely that user x will like mac and cheese and user y will like Italian bread.

2. Content based filtering -

   Content based filtering focusses more on user profile or user's regular/ preferred choices and product profile like description / title of the product. With this type of recommendation system, products are detailed with titles or described using keywords and customer profile will show the product of customer choices/ likes.

3. Hybrid filtering:

   In hybrid filtering products are recommended using both content-based filtering technique and collaborative filtering technique and user /customers are suggested with wide range of product.

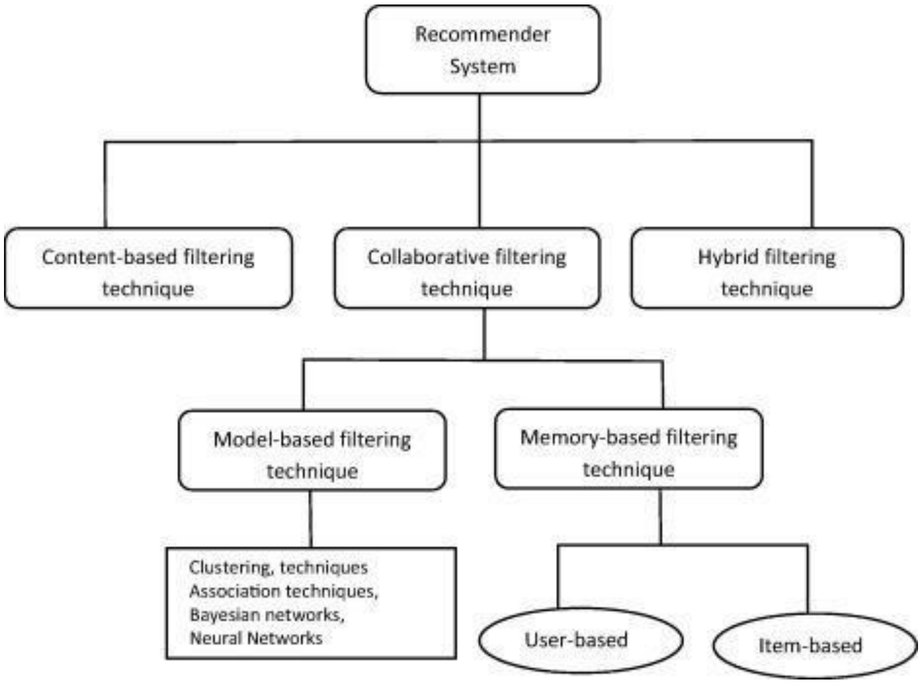


Figure 1. 3: Types of Recommendation System.

With this study we will implement a content-based filtering technique based on user/customer input, we will gather and analyse product visual similar attributes, such as shape, size, colour, style, or pattern and recommend product to the user/customer.

Overall, this study will help to implement end to end recommendation system where user provide product image of their choice into the instant messaging application and get the recommended product based on content-based filtering.

## 1.6 Structure of Study

This thesis is structured into six chapters giving over all idea of the research problem, methodology, work carried out, implementation.

Chapter 1 is providing the background information of the topic, problem statement and explains the purpose of study, significance of the research and drawing the online defining the in and out scope of the research.

Chapter 2 provides an overview of the current state of knowledge on the topic being studied. It typically includes a summary and analysis of previous research on the topic, and may also discuss gaps in the existing literature and areas for future research.

Chapter 3 explains the methodology of the research provide a clear and concise description of the research methods used in the study. This section typically provides the detailed description of the study design, the sample population, the data collection and analysis techniques, and any other relevant information.

Chapter 4 provides details regarding data sourcing, data collection, data cleaning and feature selection, after cleaning the data and identifying the feature of the data is visualized the under the insights of the data.

Chapter 5 of this thesis involves the implementation, experiments and result evaluations in which dataset extracted from chapter 4 is passed through different model to identify the best model and build recommendation system which can ultimately provide recommendation of N product based on the input image provided by the user.

Chapter 6 provides an overall discussion result and provide the summary of all the work carried out in the research and conclude the thesis by discussing about the future work around the image recommendation system.

# CHAPTER 2: LITERATURE REVIEW

## 2.1 Introduction

As the demand of Internet is growing day by day, source of information is multiplying dramatically, creating the issue of unwanted or overload of information which in turn prevents users from quickly finding important piece of information. The creation of the Internet also helped in the rapid development of e-commerce, with companies like Amazon, Flipkart, and others rising quickly, relocating physical storefronts online so that consumers could browse, pick, and purchase a variety of things without ever leaving their homes. It would not be wrong to say that e-commerce has been fully integrated into people's lives and has evolved into a necessity (Huang, 2022).

However, since e-commerce has grown quickly, statistics on the industry suggests an exponential rise that is much greater than what most consumers can handle. This problem is known as "information overload". Users' browsing, selection, and decision-making are greatly impacted by the vast data and information present on e-commerce platforms.

Search engines and recommendation systems were developed in response to the major problem of information overload. People's demands are converted into keywords and then submitting it to the background for search and return of the result information makes search engines more suitable for the clear purpose that people need.

However, there is an issue with search engines. The result returned by search engine are easily impacted by other users and it is challenging to precisely get the necessary data. If we talk about recommendation system, they are more tailored and proactive and using personal usage history they will push more interesting content to consumers.

In last few decades, to solve the issue of information overload has made the developers think of a greater number of solutions. In order to address the issue of cold start in conventional collaborative filtering algorithms, some researchers presented an apparel recommendation algorithm which uses collaborative filtering and applied visual attention model to apparel photos. According to experimental findings, this algorithm outperforms conventional collaborative filtering algorithms when it comes to recommending apparel. A hybrid proposal approach is the one that combines several different types of recommendation techniques. The hybrid proposal algorithm, which lacks applicability in e-commerce recommendation and struggles to satisfy user expectations, is created by combining content-based recommendation with collaborative filtering recommendation. The foundation of knowledge-based recommendation algorithms is the development of a knowledge base and the acquisition of domain rules. These algorithms use domain knowledge to reason in accordance with domain rules and then recommend the final inference findings. In conclusion, the above-mentioned recommendation algorithms are not appropriate for usage with an e-commerce recommendation system, hence a new algorithm must be proposed to create an e-commerce IRS.

Also, Due to rapid growth in information technology and imaging technologies number of images is increasing on internet weather image of product on e-commerce, image of new fashion trend, image of people with common like. So, business to grow attract new user into their business at the same time make life of the people easier recommendation system will play a vital role.

**2.2 Related Research**

(Wang et al., n.d.) The paper is specific to handbags and proposes methods through which handbags can be recommended to each customer. This paper uses join learning attribute projection along with one class SVM classification for the images clicked by user/customer. Through feature extraction feature of bag images clicked by customer is extracted and mapped to projection matrix. Then projection matrix is used along with SVM classifier to get the new recommendation.

(Chen et al., 2015) Convolutional neural networks and image-based product recommendations are topics covered in the paper. The paper uses CNN to determine the category that this thing most likely falls under before passing via an SVM, AlexNet, and VGG layer to determine the suggested purchase.

(Ullah et al., 2020) In the paper, a recommendation system based on content-based picture retrieval is proposed, where the first class or kind of the product is determined using a Random Forests (RF) classifier and where JPEG coefficients are employed for feature extraction. The second part of the suggested suggestion method involves retrieving closely related products.

(Sulthana et al., 2020) The main topic of the paper is Improving the performance of an image-based recommendation system with deep learning and convolutional neural networks. A deep architecture and several "convolution" processes that result in the overlapping of edges and blobs in images are used to facilitate improvisation.

(Stewart, 2012) The paper discusses Modelling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering in which novel models are built for the One-Class Collaborative Filtering setting, where our goal is to estimate users' fashion-aware personalized ranking functions based on their past feedback.

(Dwivedi et al., 2020) Paper discusses different types of recommendation system though which products can be recommended by recommendation system precisely on amazon data. Also, in this research popularity-based recommendation engine and recommendation using collaborative filtering is discussed and evaluated best model based on rmse, mean square error, mean absolute error.

(McAuley et al., 2015) This paper discusses recommendation model for style and substitutes. With this research model is discussed and developed which recommend compatible product which goes with each other or not like which clothes and accessories goes together and which do not go together which helps in styling. In this research using F-dimensional feature vector is calculated using CNN model which is used to group the product which can go along with each other.

### 2.3 Research gap

All the research done till now like (Chen et al., 2015)(Ullah et al., 2020)(Jouyandeh and Zadeh, 2022)(Elsayed et al., 2022)(Rashed et al., 2022) is more with respective to find the best model for recommendation on different dataset and user experience is not considered in the research. In this research we have covered and considered the research recommendation from end user prospective where best identified model is pickup to recommend product to end user based on visual similarity. Also, the recommended product to user not only contains product image but some other essential information like product rating, cost, brand and navigation url to buy the product which were missing in the existing research.

Existing research (Stewart, 2012), (McAuley et al., 2015) , (Ullah et al., 2020), (Chen et al., 2015) done on same dataset have less sample and categories which has affected overall accuracy and efficiency of the model. Also, the experiment done in this research misses some crucial models like GoogLeNet and Xception model.

Existing research rely solely on the visual content of the images, without considering the broader context in which the images were taken or the user's personal preferences and interests. This is research we tried to incorporating contextual information product recommendation and recommending the product details like product rating, cost, brand.

Also, all the existing implementation in the previous researches and testing is done on the data provide in the dataset but in this research and thesis we have considered the image from dataset as well as images from user handset either saved images or image from camera which can pass to recommendation system through bot.

## 2.4 Recommendation System

The Recommendation systems assist users in finding information, goods, or services. It aids e-commerce sites in boosting sales and has become into a significant research area.

Recommendation systems generally functions with two different types of information:

- Characteristic information: This data relates to things (keywords, categories, etc.) and people (preferences, profiles, etc.).
- User-item interactions: This data includes things like ratings, the quantity of transactions, likes, etc.

This allows us to differentiate between three algorithms that are applied in recommender systems.

In general, recommendation system can be classified into:

- Collaborative filtering (CF)
- Content-based filtering (CB)
- Hybrid filtering (HF)

Collaborative filtering-based image recommendation systems heavily rely on keywords. Tagging the keywords to every image in the massive image databases is a significant task. At the same time, with just a few keywords, it is challenging to satisfy the needs of a variety of consumers as well as all the information included in an image. Also, sometimes consumers may struggle to come up with the best keywords to fully express what they desire. Consequently, the content-based image recommendation system is gaining popularity and improving swiftly.

Content-based image recommendation filtering is based on image resemblance and usually includes following steps:

1. Feature extraction: Typically, the image is represented by the characteristics. A wide range of features, from low level to high level, have been proposed to define an image. Over the past several decades, researchers have worked extensively on image feature extraction.

2. Feature matching: Typically, the attributes of the query images are used as a template to match the attributes of the images in the resource database. The most difficult task is figuring out how to search the similarity rank effectively.

3. Hybrid filtering integrate both sorts of information in order to prevent the issues that arise when working with only one kind.

### 2.4.1 History of Recommendation System

Recommendation system (RSs) has gained popularity along with the growth of internet. With great advancements in internet technology and e-commerce we have seen rise in research in various areas of recommendation system. Recommendation system are quite beneficial for e-commerce and has enhance business of e-commerce industry by helping customer to identify the product with no knowledge on online shopping, by selling the product to customer which are related to each other and improved customer confidence.

The research in field of recommendation system gained some momentum when Amazon launched collaborative filtering in 1990s, through implementation of RSs amazon has enhanced there business online and effectively increased its sales when it introduced Collaborative Filtering (CF) at the end of the 1990s. As a result of Amazon's popularity, other online companies began to adopt RSs on their websites. And later Amazon has secured a United States Patent for its CF technique.

The RS can be seen as a subset of information filtering because the primary objective of an RS is to locate the preferred information and delete information that a user dislikes. After analysing a user's preferences using historical data, the data is processed using machine learning algorithms to create a prioritised list of suggestions that reflect the user's tastes.

Since the dawn of computing, there has been talk of using computers to recommend the best product to the user. The Grundy system, a computer-based librarian that recommended books to users, was the first application of the RS concept. It debuted in 1979. After that in early 1990s, the first commercial recommendation system (RS) was introduced known as Tapestry.

Another RS implementation was launched by GroupLens, a research lab at the University of Minnesota in the United States in early 1990s, for assisting users in finding their chosen articles.
GroupLens Recommender System is the term given to the system in honour of the group. This system asserts to share characteristics with Jester, Tapestry, Ringo, and BellCore. One of the most well-known RS technologies, Amazon Collaborative Filtering, was implemented as part of the late-1990s RS development. Collaborative filtering-based RSs have grown in

popularity since this time and are now being used by numerous internet and e-commerce platforms. Additionally, many RS toolboxes have been created. The success of Amazon has also sparked the creation of other hybrid RS algorithms, which integrate several different techniques.

After the prosperous period at the end of the 1990s, the industry generously provided financing to put RS' research into practise. A company that offers online streaming media, Netflix, hosted the most well-liked tournament in RSs. In 2006, they introduced the Netflix Prize1, which awards the greatest recommendation for an RS movie with a million dollars. In 2009, they made the victorious team public. An RS was also added to YouTube's website in 2010.

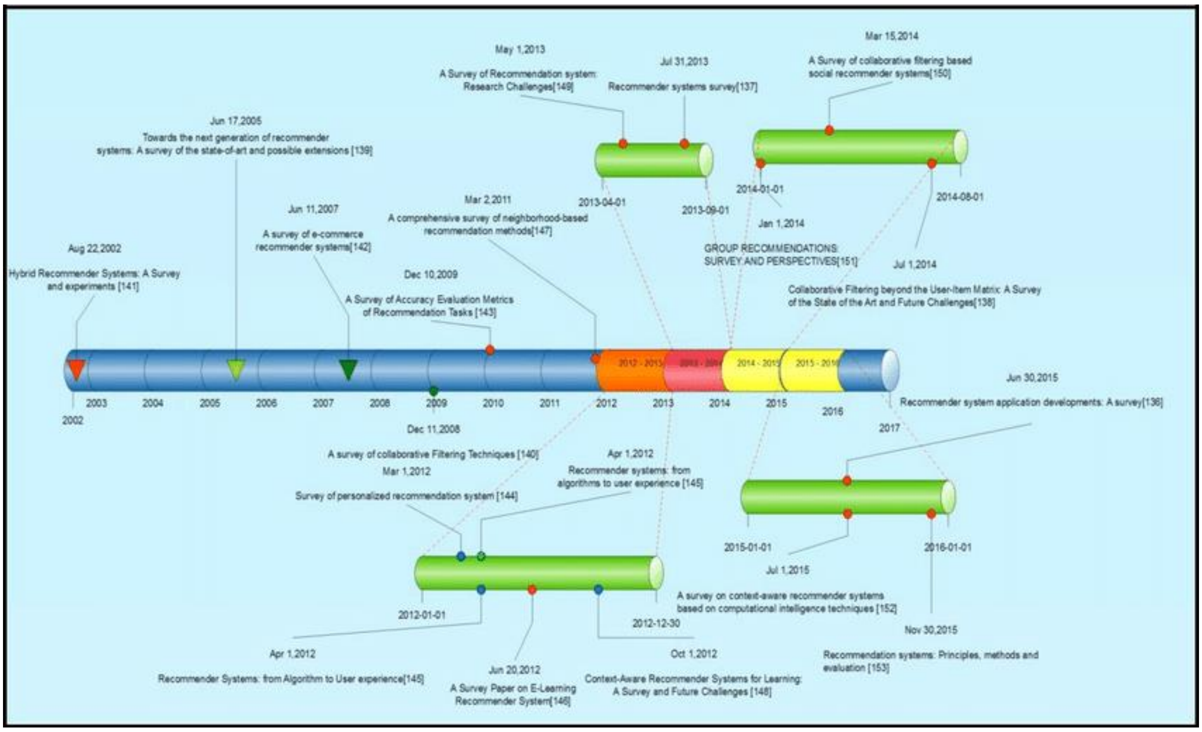


Figure 2. 1: Timeline review of recommendation systems (Ref: (Taneja and Arora, 2018))

### 2.4.2 Application of Recommender System

Nearly all businesses may benefit from a recommendation system. Two important aspects determine how much a firm gain from a recommendation system:

- Breadth of data:

  A business with a limited number of customers that act differently won't find an automatic recommendation system to be very helpful. When it comes to learning from a few instances, humans continue to be light years ahead of robots even in modern times. In these circumstances, your team will employ logic, customer information that is both quantitative and qualitative, to provide good suggestions.

- Depth of data:

  With only one piece of information about each consumer, recommendation algorithms are ineffective. Detailed information about clients' online behaviour and, if feasible, offline purchases can serve as a reference for accurate suggestions.

We can identify the following sectors that would benefit from recommendation systems using this framework:

- e-Commerce:

  E-Commerce was the first industry which show the growth when recommendation system was incorporated in their system. Today ecommerce industry has millions of clients and based on their clients’ online behaviours they produce accurate suggestions.

- Retail:

  Back in the 2000s, when Target's computer systems could predict pregnancies even before parents were aware about the pregnancies, they alarmed customers. As the most immediate indicator of a customer's intent, shopping data is the most important type of information. Companies that make accurate recommendations are working with retail companies who have large set of shopping data available with them.

- Media:

  Different media house can provide content to the users as per their choices and preferences and also suggest the trending items. This is possible with the recommendation system.

- Banking:

  Today bank sectors provide their services online which is easily accessible to consumers. Mass banking and Small medium enterprises (SME) provide wide variety of proposals. Analysing customers financial conditions, their past preferences and comparing with the data from lakhs of other consumers having similar experiences can be used to create products which will be influential and beneficial.

- Telecom Sector:

  Related interactions exist between IT and banking. Millions of clients that have all of their interactions recorded are accessible to telcos. When compared with other sectors their product selection is quite constrained and due to which telecom recommendations are simple.

- Utilities:

  Same as Telecom but due to smaller selection of items in utility sector it makes recommendations easy.

Examples of businesses that make use of recommendation engines:

1. Amazon:

Amazon uses item-to-item collaborative filtering recommendations. As per the McKinsey 35% of Amazon business, comes from recommendations. Here are a few instances when Amazon uses recommendation systems:

   a. Related to items you've viewed
   b. Recommended for you, XXXX (name)
   c. Amazon.com: Bestselling Canon Cameras

2. Netflix:

Another data-driven business that uses recommendation engines to raise consumer satisfaction is Netflix. The same Mckinsey study that we discussed before shows that recommendations account for 75% of Netflix viewing. In fact, Netflix is so committed to offering users the best outcomes that it has held data science competitions called the Netflix Prize, with the winner receiving a prize of $1,000,000.

3. Spotify:

Spotify is a subscription-based platform in which based on the subscribers' musical preferences they create a personalised playlist weekly known as Discover Weekly for their subscribers.

Using their musical intelligence and data analytics business Echo Nest they developed a music recommendation engine which uses three types of recommendation models:

a. Collaborative filtering: Using cross-referencing for e.g.: Based on different users listening history songs are selected.

b. Natural language processing: Searching online for information on specific performers and songs. Then, each artist or song is assigned a dynamic list of top phrases that is weighted by relevance, updated daily. Next, the engine decides if two musical works or musical performers are comparable.

c. Audio file analysis: For audio analysis we have different parameters like volume, temp, key, time signature etc. These parameters are analysed by the programme and suggestions are provided based on the analysis.

4. LinkedIn:

LinkedIn employs "You may also know" and "You may also enjoy" types of suggestions, just like any other social networking platform.

### 2.4.3 Types of Recommender System

Recommender systems are systems that aggregate user recommendations before sending them to the appropriate recipients. Additional definitions include a system that produces individualised suggestions as an output or one that guides a user in a specific way toward appealing options among a larger selection of options.

Most common recommender system kinds are six:

I. Collaborative Recommender System:

It is one of the most popular, widely applied, and cutting-edge technologies available right now. Using a combination of user ratings and suggestions, collaborative recommender systems can determine user similarities and provide new recommendations based on user comparisons. The primary benefit of collaborative techniques is that they can be utilised for complex issues where a significant amount of the variation in preferences is due to differences in taste. Additionally effective for complicated items, these strategies. The foundation of collaborative filtering is the assumption that individuals would continue to prefer similar products as they have in the past.

II. Recommender systems based on content:

Generally speaking, it is consistent with the advancement and continuance of information filtering research. The attributes that go along with each item in this system serve as their primary definition. Based on the characteristics of the products the new user has rated, a content-based recommender builds a profile of personal tastes. Keywords are utilised to define the objects in a manner similar to a recommender system that uses specific keywords. Customers are therefore provided recommendations for products similar to those they currently or previously enjoy in a content-based recommender system.

III. Recommender system based on demographics:

This method aims to categorise users based on characteristics and offer guidance based on demographic categories. As it is not overly complicated and is simple to apply, several industries have adopted this type of strategy. For a demographic-based recommender system to work, extensive market research in the intended market and a quick survey to gather data for categorization are first needed. While using distinct data, demographic algorithms provide "people-to-people" correlations similar to collaborative ones. The advantage of using a demographic approach is that user ratings in the past are not required.

IV. Utility based recommendation system:

Before producing recommendations, a utility-based recommender system evaluates the relevance of each object for the user. How to create a utility for individual users is, of course, the key challenge for this kind of system. Each industry will have its own method for determining a utility function that is unique to the user and applying it to the considerations made in a utility-based system. The primary advantage of using a utility-based recommender system is that non-product factors, like vendor dependability and product availability, can be considered when evaluating utility. It is possible to give the user a real-time inventory display of the object by doing this.

V. Recommender System based on knowledge:

This kind of recommendation engine tries to generate product suggestions based on assumptions about the requirements and preferences of a user. This recommendation system is designed on functional knowledge because they understand the association between the user requirement and proposed recommendation since they learn how a certain item satisfices user needs.

VI. Hybrid Recommender System:

The two systems are combined in hybrid recommender systems in a way that is appropriate for a particular industry. Since it combines the benefits of many systems and addresses any potential limitations that could result from employing a single recommender system alone, this is the recommender system that many firms pursue.

The systems can be coupled in a variety of ways, such as:

- Weighted Hybrid Recommender:

This system determines a recommended item's score by combining the outcomes of all the different recommendation methods. As an example, the P-Tango system combines content-based and collaborative recommendation algorithms, initially giving both equal weight but gradually modifying it as assumptions on user ratings are validated or disproven. Instead of using scores, Pazzani's combination hybrid considers the output of each recommender as a collection of votes that are then merged in a consensus system.

- Switching Hybrid Recommender:

  According to predefined criteria, the Switching Hybrid Recommender switches between various suggesting techniques. The switching hybrid recommender may initially deploy the collaborative-based recommender system and the content-based recommender system if that doesn't function if the collaboration and content-based recommender systems are combined.

- Mixed Hybrid Recommender:

  When it is possible to make numerous recommendations at once, we should employ mixed recommender systems. As a result of the simultaneous presentation of recommendations from many methodologies, the user can choose from a wide range of options. The majority of media and entertainment businesses employ the PTV system, which was created by Smyth and Cotter and is mostly a recommended programme to encourage viewers to watch television.

### 2.5 Challenges in recommender system

Although the skyrocketing rise indicates that companies all over the world are investigating what recommendation engines can do for them, implementing this technology effectively has its share of difficulties.

1. Cold start:

   When new users or goods are introduced to the system, this problem arises. It is challenging to forecast the decision or interest of users when an item is first put to the recommendation system without any ratings or reviews, which leads to fewer accurate recommendations.

   For instance, a recently released film cannot be recommended to a user until it has received several reviews. Because it is impossible to identify another person who shares the same interests or preferences without knowing the person's past behaviour, a problem brought on by the addition of a new user or item is difficult to resolve.

2. Sparsity:

   It frequently occurs that most users don't rate or review the products they buy, making the rating model highly sparse and potentially causing data sparsity issues. This makes it harder to identify groups of users that share ratings or interests.

3. Synonymy:

   When two or more names or lists of objects with similar meanings are used to refer to the same thing, this is known as synonymy. In this case, the recommendation system is unable to distinguish whether the words refer to the same thing or something else entirely. For instance, recommendation engines assume that "action movie" and "action film" are equivalent terms.

4. Privacy:

   In order to receive more helpful services, a person often has to input his personal information (have an experience with hyper-personalization) to the recommendation system. However, doing so raises concerns about data privacy and security, and many users are hesitant to do so.

   In order to offer individualised suggestion services, the recommendation system is obligated to obtain the personal information of users and use it to the maximum extent possible. The recommendation systems must assure user confidence in order to address this problem.

5. Scalability:

   The scalability of algorithms using real-world datasets for the recommendation system is one of the largest problems. Since user-item interactions like ratings and reviews create a lot of changing data, scalability is a major worry for these datasets.
   Large dataset results are inefficiently interpreted by recommendation systems; to address this problem, sophisticated large-scaled approaches are needed.

6. Latency:

   As additional items are uploaded to the databases of recommendation systems, we note that only those that have already been rated are recommended to users. Newly added products are not yet reviewed.

   Consequently, a latency issue emerges. Combining the category-based strategy, collaborative filtering, and user-item interaction may be the answer to this issue.

### 2.6 Comparison of Techniques

Below table compares several techniques used in recommendation systems.

| | Collaborative Recommendation Filtering | Content based Recommendation Filtering | Hybrid Recommendation Filtering |
|---|---|---|---|
| Working / Advantage | •Determine how people will rate for unrated items.<br>•Determine similarity score among users.<br>•No prior understanding of user features is necessary.<br>•Serendipitous recommendation. | •Work on user past history<br>•No need for data on other users.<br>•No cold start and sparsity.<br>•Able to recommend users with unique taste, new and unpopular items. | •Combine more than one technique.<br>•Overcome Limitation of Content base filtering and collaborative filtering. |
| Disadvantage | •Expensive Model building Lose useful.<br>•Information for Dimensionality reduction Technical.<br>•New user Cold start<br>•Data sparsity<br>•Scalability<br>•User Data Privacy | •Limited content<br>•Analysis Over specialization | •Increased complexity<br>•Increased expense of implementation |
| Techniques | •Association rule<br>•Clustering<br>•KNN<br>•Decision Tree<br>•CNN<br>•Regression<br>•Correlation Based<br>•Pearson correlation<br>•Cosine based | TF-IDF Term Frequency and Inverse Document Frequency | Combination of Collaborative and content based filtering Techniques |

Table 2. 1: Comparison between recommendation systems.

### 2.7 Discussion

In this section so far, we have discussed:

- Brief about recommendation system.
- Different types of recommendation system.
- How recommendation system evolved.
- Application and use cases where recommendation system is used.
- Advantages and disadvantages of recommendation system.

- Challenges of recommendation system.
- Compared different types of recommendation system.

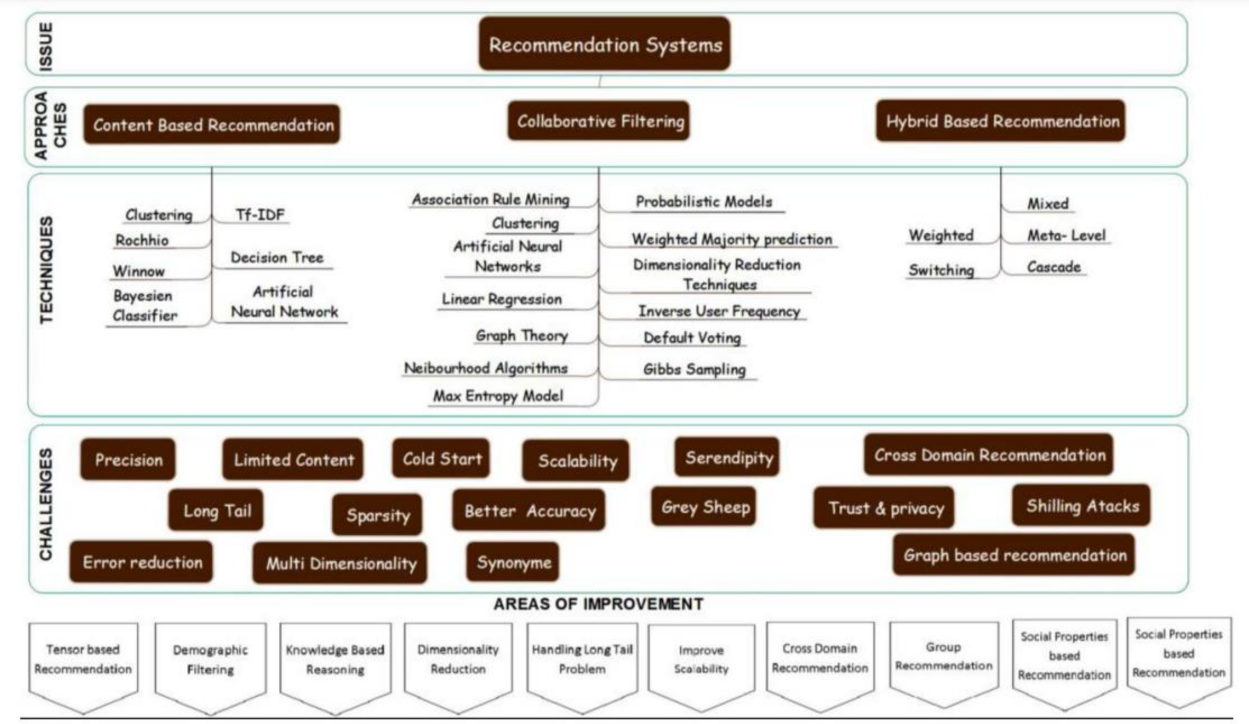


Figure 2. 2: Recommendation System and their types

### 2.8 Summary

With this implementation we will take user input from instant messaging application and pass through the recommendation filter that would be content-based filtering technique since filtering would be done based on user/customer input, later we will gather and analyse product visual similar attributes, such as shape, size, colour, style, or pattern and recommend product to the user.

Recommendation system based on content-based filtering technique will implement two major tasks:

- Classify and categorise input image provided by customer using CNN.
- List the product based on visual similarity and recommend the most common/similar product to the customer.

# CHAPTER 3: RESEARCH METHODOLOGY

## 3.1 Introduction

This chapter presents the research methodology, involving the approach and implementation plan where first user provides input image from bot and this image will be passed through content-based recommendation system which recommend the similar product to the user. Research methodology also involve designing recommendation system based on CNN architecture models' different architecture model is compared against each other and identify the best CNN architecture which will be used to build final model.

## 3.2 Methodology

This research involves complete end to end implementation, where in first phase we need to create bot and second phase we will create recommendation system which will take input from the bot and respond to bot with the result.

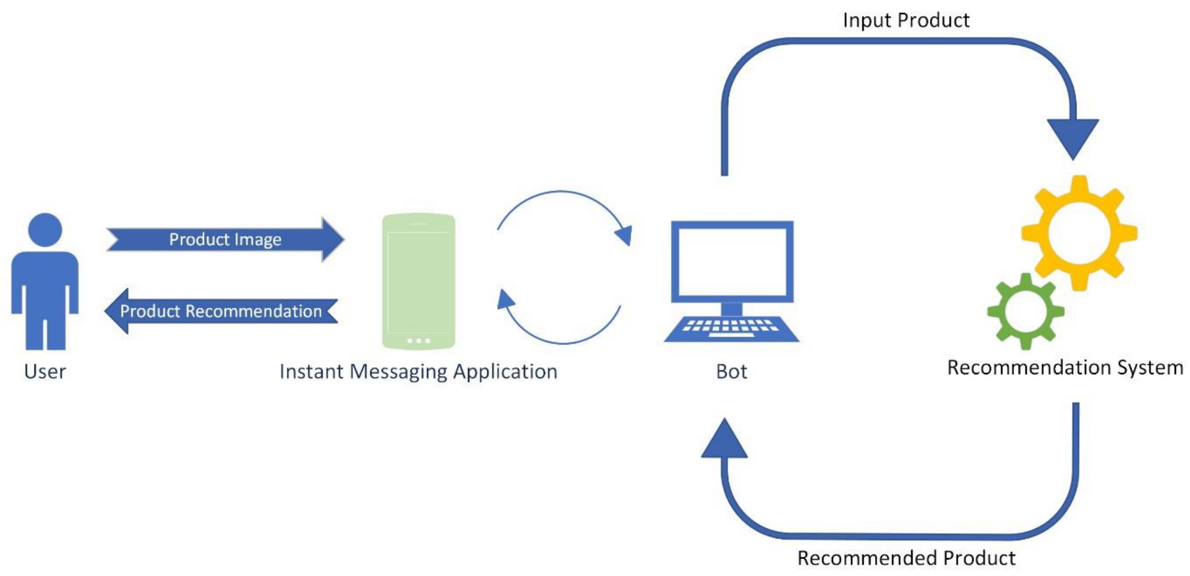


Figure 3. 1: Overview for implementing image-based recommendation system

From the above architecture we can see, user through instant messaging application like WhatsApp and Telegram will connect to bot and provide image as input to the bot which will connect to recommendation system and recommendation system will return the recommended product to the user.

Methodology to setup and create new Telegram bot:

- To create a new bot, we need to login into the Telegram application and search for “BotFather”. (BotFather is the bot provided by telegram application through which we can create new bot account and manage existing bots)

- On “BotFather” chat create new bot by “/newbot” command.

- “BotFather” will ask for a bot name and username and will generate authentication token for newly create bot.

- This token value is unique for every bot, and we will use this token in our python code and connect to the recommendation system.

- Along with token, name provided will be displayed in contact details section through which user/customer can connect to bot-based recommendation system.

Recommendation System:

In Recommendation system, we will implement two major tasks:

1. Classify and categorise input image provided by customer.
2. Get the list of similar products and recommend the most similar product.

We will classify and categorise the input product using a convolution neural network as the initial step of the recommendation system. Convolution Neural Networks are a deep learning algorithm and are specialised particularly working in the neural network field of visual data like images and videos.

In Convolution Neural Networks, image classification is done by complex network with multiple layers and this layer will help to get the feature and identify the product, Example, consider a complex neutral network.

1. Horizontal and vertical edges are extracted in the first layer as raw features.
2. More abstract elements, such textures, are extracted in the second layer (utilizing the attributes that the first layer extracted)
3. Based on the textures, the subsequent layers may be able to distinguish specific aspects of the product image, such as belt, cloth, material, shape, size, etc.
4. Further layers may reveal type, colour, style, or pattern, etc.
5. And last layers finally, classify and categories the image as Electronics, Pet Supplies, Home and Kitchen, Cell Phones and Accessories, Clothing Men, Clothing Women, Watches and Shoes.

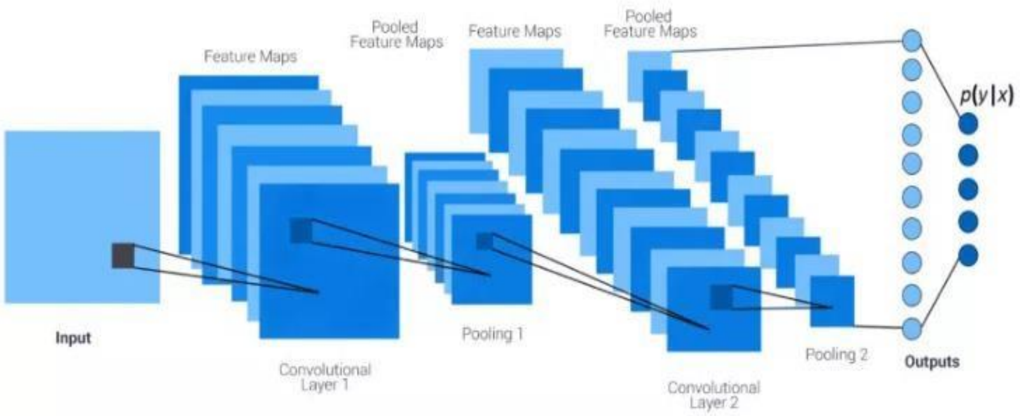


Figure 3. 2: Example of CNN Architecture

The above figure 3.2 is a simple architecture of CNN, where we have used multiple Convolutional Layers and multiple pooling layers (Pooling layers in typical CNN architecture signifies the layer which is statistical aggregate of previous layers, and these layers are used to reduce the dimensions of the feature).

In this thesis, we will try to classify the product using the different CNN Architecture models to compare it and for better training, validation, and test accuracy. Different architecture we will try to explore are:

- SVM (Support Vector Machine)
- VggNet
- GoogLeNet
- ResNet
- Xception

After, the first step i.e., of classification using CNN module we will categorize the product provided by customer under its respective category. After this we will get the list of similar products and recommend the most similar product.

In the second part of recommendation system, our recommendation engine will recommend product based on image, for which we will calculate cosine similarities between the feature extracted from pre trained model and image. Then, images are suggested based on the calculated cosine similarity, listed in descending order. So, higher the cosine similarity, the recommended image will be more similar to the input image.

Cosine similarity formula,

$$similarity(A,B) = \frac{A.B}{||A|| \times ||B||} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

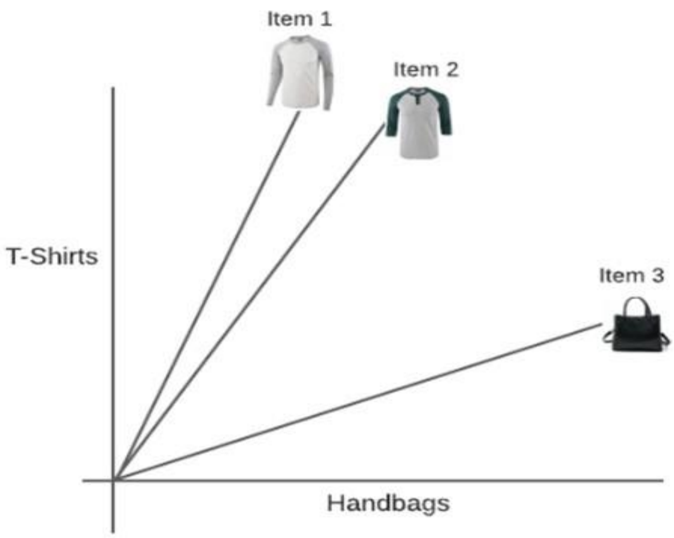


Figure 3. 3: Illustration of similar product tends to high cosine value or group together.

For better result, we can calculate cosine similarity between the text i.e., text feature like title and brand. In other words, for effective recommendation we can include product recommendation based on text for which we will calculate cosine similarity score using weighted average formula which is individual cosine similarity score along with weight of each type.

Formula for the weighted average,

$$Similarity\ Score = \frac{Weight_p * Similarity_p + Weight_b * Similarity_b}{Weight_p + Weight_b}$$

### 3.2.1 Convolution Neural Networks

Machine learning consist of various algorithms, along with neural network is an integral part of ML, it is a subset of machine learning which is based on the idea of how biological neural networks are organised and operate.

Neural Networks consist of neuron which are made up of small, discrete unit and are group together to form layers and each layer are interconnect to create a network where data flow from input layers to output layers. The neuron in each layer's perfume certain calculation and send data to next layer which again performs certain calculations and in the end output of transmitted though these network layers of neurons.

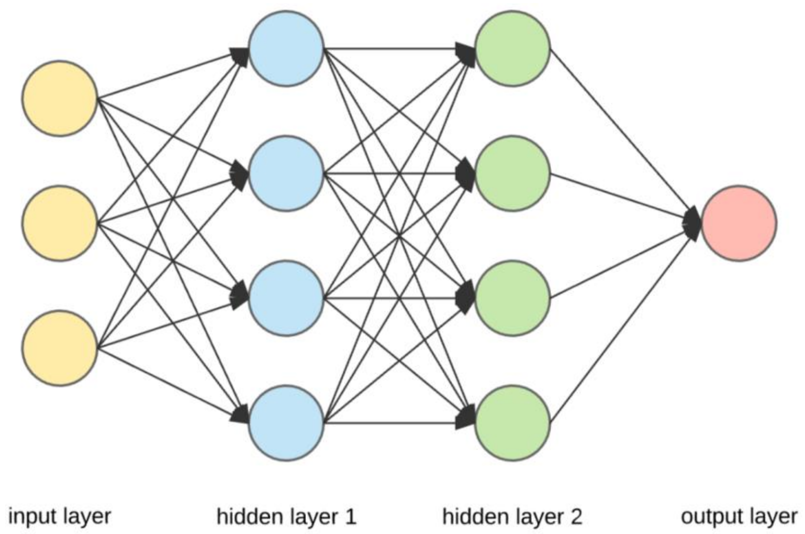

Figure 3. 4: Simple diagram of neural network

Figure above illustrates simple neural network where data from input layer is passed to output layers through hidden layers and each layers has the capability to vary computing power and based of accumulation of experience transmit output through output layers. This neural network become more complex and able to solve wide range of complex problem by more technological structure called deep learning. Image classification and recommendation is one of the examples of that.

Yann LeCun first proposed convolutional neural networks (CNN) in 1988 as a unique type of artificial neural network (ANN) design. Image categorization is one of the most widely used applications for this architecture.

Example - CNN is used by Facebook's automatic tagging algorithms, Amazon uses CNN in their product recommendation system and google uses CNN to search photo’s.
CNNs are made up for multiple convolutional layers and pooling layers to make a network of layers there are many ways to arrange their layers in the network.

**Different layers in CNN**

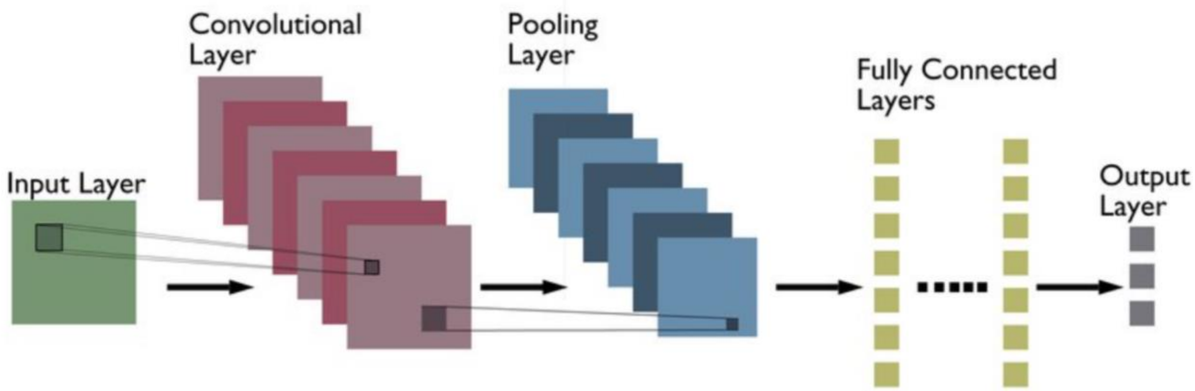


Figure 3. 5: Architecture showing different CNN layers

- Convolutional layer:

  A collection of filters (also referred as kernels) used to develop convolutional layers on which input image from input layer is passed. Output of this layer is a feature map which is characterization of input image on which convolution layer attaches filter. Convolutional layers can be combined to build more sophisticated models that can extract finer details from images.

- Pooling layer:

  It is another type of convolutional layer which reduces spatial size of image and makes it easier to process. Pooling facilitates parameter reduction and accelerates training. In terms of memory consumption pooling layer consumes less memory.

  There are two primary types of pooling:

  - Max Polling – Each feature map's maximum value is extracted using max pooling.
  - Average pooling – Average pooling takes average value from each feature map.

  In order to reduce the size of the input before it is fed into a fully connected layer, pooling layers are often used after convolutional layers.

- Fully connected layer:

  In fully connected layer each neuron of this layer is fully connected to every other neuron in the layer below it. Fully connected layers is mostly used that the end of CNN when aim to predict based on the feature extracted from the previous layers. For example, in our case CNN will be used to classify and categorise input image of product, the final fully connected layer will take features learned from previous layers and then classify the images as shirt, shoes, watch etc.

**Architectural Design for CNNs**

So, far we have discussed different layers in CNNs model. Challenging part of CNNs is designing simple model architectures which would be best for given elements. Below is different architecture of CNNs:

**LeNet-5**:

Is was the first know application of CNNs, LeNet-5 CNNs was first described by Yann Can (Lecun et al., 1998). The model suggests a pattern consisting of average pooling layer, also known as a sub-sampling layer, followed by a convolutional layer. Prior to flattening the generated feature maps and sending them to numerous fully linked layers for analysis and a final prediction, this cycle is repeated 2.5 times.

The model suggests a pattern consisting of a convolutional layer, then an average pooling layer, or subsampling layer. This process is done two and a half times, and the final feature maps are flattened and sent to several fully connected layers for interpretation and a conclusion.

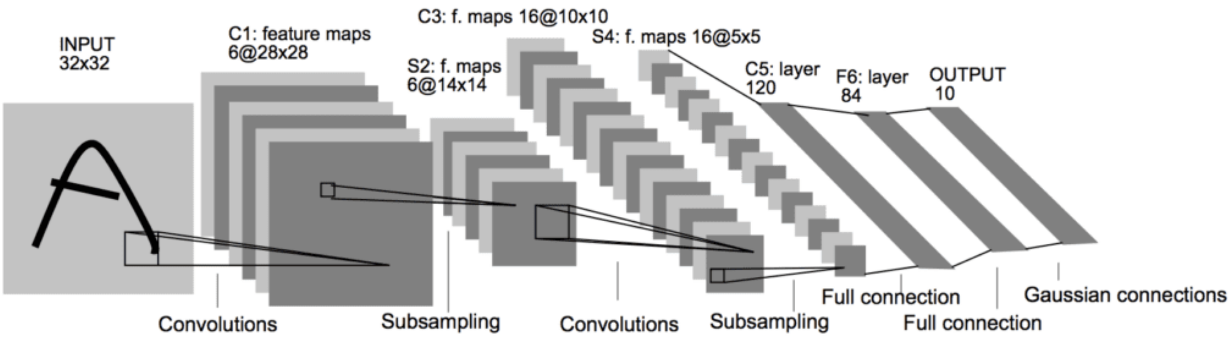


Figure 3. 6: LeNet-5 CNNs Architecture

Key components of the architecture:

- Fixed-size photos for input.
- Blocks of convolutional and pooling layers should be created.
- The architecture repeats building blocks from convolutional pooling.
- Filters become more prevalent as the network depth increases.
- Distinct components of the architecture for feature extraction and classifier.

**AlexNet**

AlexNet was one of the first architectures which uses 8 layers in their architecture among which 5 convolutional and 3 fully connected. One of the features of AlexNet is it has very large kernel of size (11,11), (5,5). AlexNet architecture where first among all the architecture to use dropouts. AlexNet architecture also made you of ReLU in each convolutional layer.

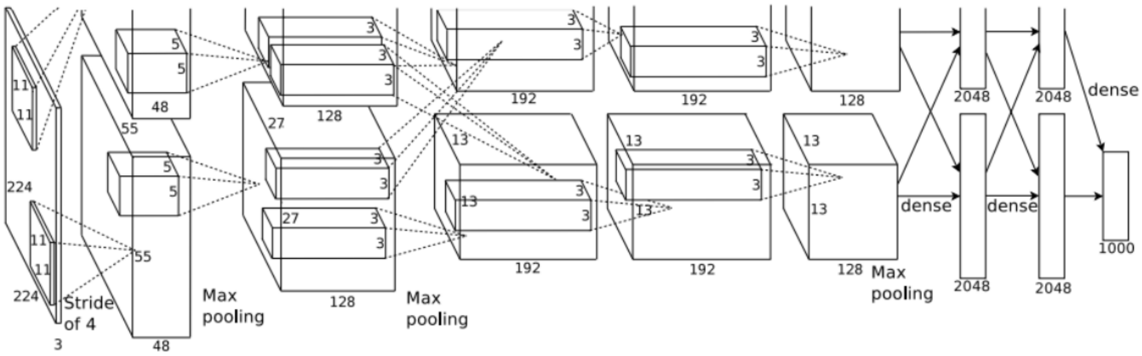


Figure 3. 7: AlexNet CNNS architecture for image photo classification (Ref: (Gonzalez, 2007))

Key components of architecture:

- After convolutional layers, ReLU activation function is used, and SoftMax is used in the output layer.
- Instead of average pooling, max pooling is done.
- Between completely connected layers, dropout regularisation is performed.
- Data augmentation is used.
- A convolutional layer's pattern is directly supplied to another layer of convolutions.

**VggNet**

VggNet was specially designed with classification task of 1000 categories for ImageNet challenge. Consequently, the final SoftMax layer includes 1000 categories. Below figure shown is a VggNet architecture in which blue layers are the convolutional layers and yellow are the pooling layers while green layers are fully connected layers with 4096 neurons and output of this would be vector size of 4096. Overall, combining all the layers it acts as a network for feature extraction for images.

VggNet in compassion to AlexNet:

1. Reduction in Model parameters
2. Increase in depth

3. Decrease in the kernel size
4. Depth increases from 8 layers to 19 layers.

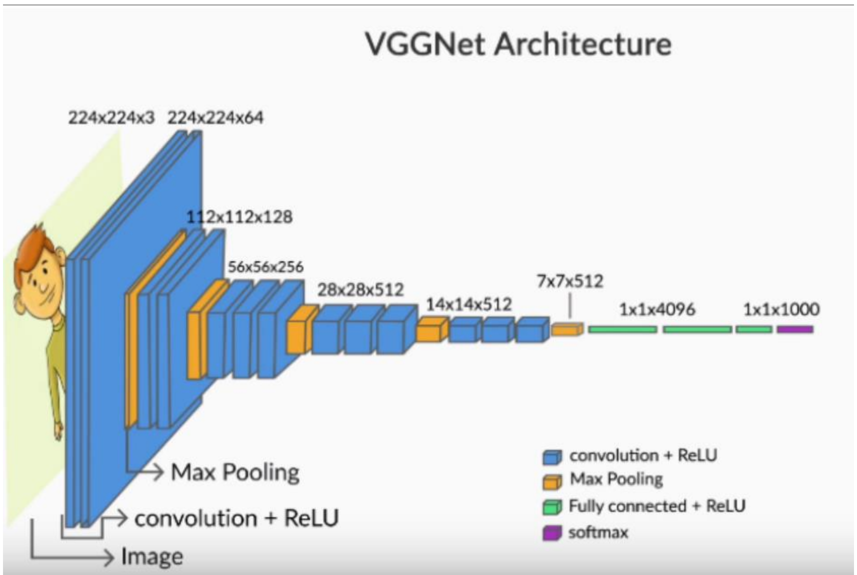


Figure 3. 8: VggNet Architecture (Ref: (Simonyan and Zisserman, 2015))

VggNet used smaller filters to create a deeper network. It is crucial to remember that smaller filters are favoured to larger ones. Below example shows two filter of size (5,5) and (3,3) and two (3,3) filter has same receptive field as of one (5,5) filter. This is due to the fact that both of these convolutions result in an output (here of size 1 x 1) with a receptive field that is the same 5 x 5 image.

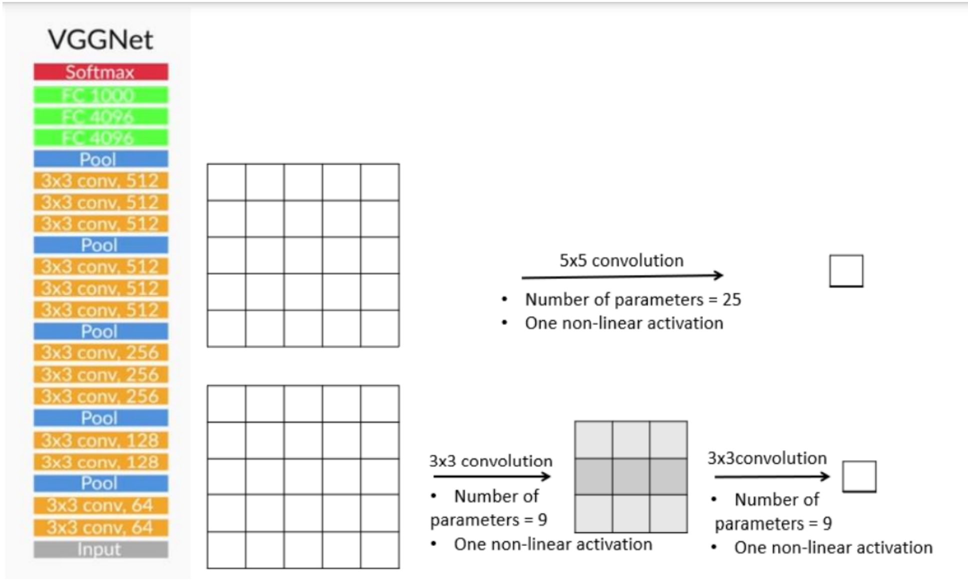


Figure 3. 9: VggNet 3x3 and 5x5 Kernel

**GoogLeNet:**

GoogLeNet has introduce new type of convolution technique called inception model. GoogLeNet inception model combines multiple convolution layers operating in parallel on the same image along with 1x1 filter and all are pooled.

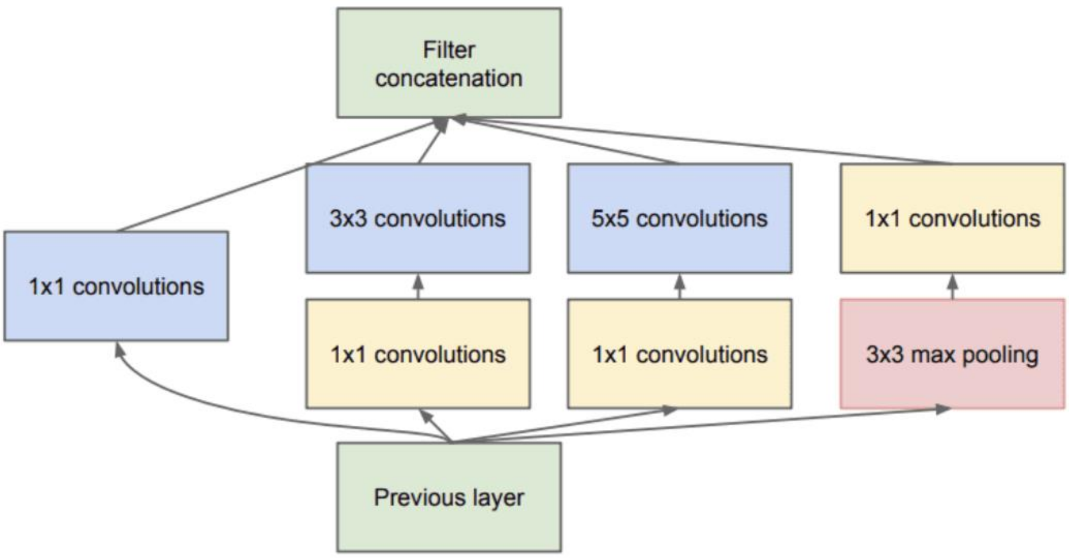


Figure 3. 10: Example Inception Model of GoogLeNet (Ref: (Zeng et al., 2016) )

Key aspect of GoogLeNet architecture:

1. Inception model are stack together in repetition to create multiple layers of total 22 layers.
2. Use of 1x1conlolution layer to reduce number of channels.
3. At multiple point use of error feedback in the network.
4. Use average pooling of size (3x3) at the output of network.
5. There is no full convoluted layer at the end except SoftMax layer is used for classification.
6. Reduced by 4 million over 60 million parameters (AlexNet)

**ResNet:**

ResNet is short from of Residual Network, In ResNet layer connected not just from one layer to next adjacent layer but it will also have skip connection. The main driving force behind the ResNet architecture was the finding that, empirically, adding more layers did not consistently improve the outcomes. Skip connection enabled researchers (Kaiming He in paper (He et al., 2016)) to train the network deep to 152 layers.

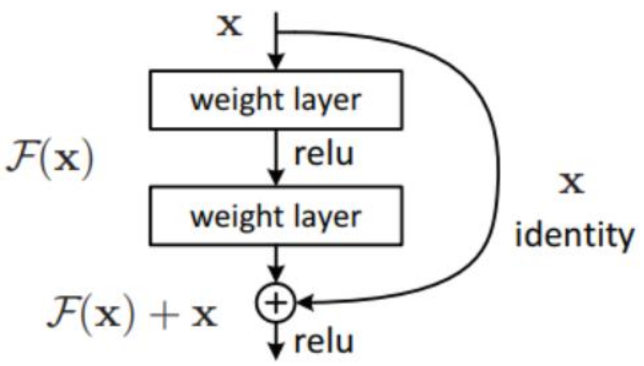


Figure 3. 11: Residual Network

Key feature of ResNet architecture:

- Use of skip connection.
- ImageNet with 152 layers.
- Each residual block with two 3x3 convolution layers.
- Average pooling for last convolution layers.
- 3.57% error rate on ImageNet.

Details with respected Xception model can we see in the research done by (Chollet, 2017)

### 3.2.2 Classification Algorithm and Types of Classification

The process of recognising, comprehending, and classifying things and concepts into predetermined groups, often known as "sub-populations," is known as classification.

Machine learning programmes use a variety of algorithms to classify upcoming datasets into appropriate and pertinent categories with the aid of these pre-categorized training datasets.

<u>Types of Classification Algorithms</u>

Depending on the dataset you are using, you can use a wide variety of classification techniques

- Logistic Regression
- Naive Byes
- K-Nearest Neighbours
- Decision Tree
- Random Forest Algorithm (RF)
- SVM - Support Vector Machine

In this research we will discuss more about SVM. SVM is a liner classification model and have ability to compute heavy data sets. SVM belong to liner machine learning model which uses liner function (y = ax + b) to set the relationship between the input x and output y.

In our research we will take SVM as a base model and compare it against VggNet, GoogLeNet, ResNet.

In SVM we will use L2 as loss function. For input image i, RGB pixels will be used as input features. $x_i \in \mathbb{R}^{\tilde{d}}$, (where d is 224 X 224) and we will be calculating score for n classes.

$$s = Wx_i + b$$

Where, $W \in \mathbb{R}^{n \times d}$ is weighted matrix and $b \in \mathbb{R}^{n}$ is bias term.

SVM loss is given by

$$L_{SVM}(W, b; x_i) = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

Where, $y_i$ is the label for the true class.

### 3.2.3 Background Removal

Background removal is one the most import task in the implementation because input image is directly from the user and it might contain noise so identification of the product, we would require to remove noise from the image by removing background.

In an image there are two attributes:

1. Centre of image – What user want to focus is at the centre of image.
2. Background – Usually even around the centre.

We divide the image into two parts as a result, using the HSV colour as the feature. With reference to paper (Yu et al., 2018) discussed background removal algorithm.

Step 1: Determine the HSV colour of each pixel in the image.

Step 2: Segment the photos into the potential segmented areas using WT to obtain an over-segmentation.

Step 3: Choose the pixels in the image's outer ring to serve as the seeds.

Step 4: Utilize the LCF to combine nearby and similar regions into a single region.

Step 5: Merge the small areas within the merged regions as the final background.

### 3.3 Approach

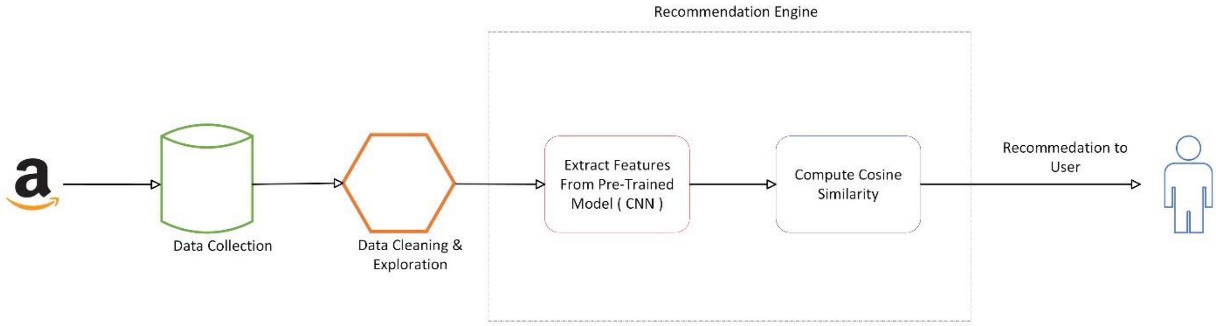


Figure 3. 12: Recommendation engine flowchart

Above figure describes the steps involved in building the recommendation engine.

- In the first step we will get the data from amazon and store it in CSV file. The data extracted is can be found in: [Amazon Dataset]

- After data loading, we will clean the data and categories the data. In data cleaning we will also remove duplicates and perform exploratory data analysis on clean data.

- After cleaning the data, perform exploratory analysis to explore the trend in the data.

- Create image classification model with aim to categories the random image using CNN model.

- Post categorisation we, will compute cosine similarity, and products with higher cosine value for the random product (or product provided by user) can be recommended.

### 3.4 Required Resources

As part of research implementation below software and hardware resources would be required:

- Software Resources
    - Python latest version installed on development environment.
    - Libraries required for development:
        - Pandas
        - NumPy
        - Seaborn
        - Sklearn
        - Tensorflow
        - Wordcloud
        - Scipy
        - Matplotlib
        - Pathlib
        - OS
        - Flask
        - Ngrok
        - Python-telegram-bot
        - Twilio
        - Requests

    - Twilio account to create a sandbox environment for WhatsApp bot.
        - For data extraction parse hub tool can be used on Amazon website.

- Hardware Resources:
    - User should have a smartphone with an active phone number and instant messaging application like WhatsApp or Telegram installed.
    - Desktop/Laptop with anaconda installed on machine.
    - System with hardware accelerator GPU/TPU and high ram.

### 3.5 Summary

In this chapter we have discussed the research methodology and understand different steps involved in the setup a telegram bot which this telegram bot will be connected to a recommendation system to recommend the product based on image similarity.

We also gone through CNN model in details and discussed able CNN architecture models Since comparing CNN model with different architecture is in scope of the thesis and research, we discussed different architecture of below models:

- SVM (Support Vector Machine)
- VggNet
- GoogLeNet
- ResNet
- Xception

Later in the end of chapter we discussed the implementation approach and required resources to build the recommendation system.

# CHAPTER 4: ANALYSIS

## 4.1 Introduction

This chapter presents the research analysis, involving data sourcing and understanding and where we will gain insights of data. In data sourcing and understanding data understanding we will first load the data from data source understand the different labels present the data access the quality of data and discuss about the source of data.

After the data is loaded, data is prepared for modelling in which analyse is done to identify the required columns, removal of duplicate values in the loaded dataset, identify new columns which could be required for modelling, drop/remove/replace existing columns, replace null values with the values best suited in the dataset. This is also called as data cleaning.

Post data cleaning, exploratory data analysis and data visualization is performed on dataset to validate the record count, datatypes of the column, missing value in the dataset, anomalies in the data, compare different group or category in the data.

## 4.2 Data Sourcing and understanding

A data source could be the place where data first appears or the place where physical information is first converted to digital form, but even the most specialised data could function as a source as long as another process can access and use it. A data source can specifically be a database, a flat file, real-time measurements from physical objects, web data that has been scraped, or any of the countless static and streaming data services that are abundant online.

As the e-commerce industry is evolving, dataset involving it is also evolving. For research work there are many ways to collected data, one by scrapping data by code or external tools like ParseHub, other way by collecting data from verified data source. In our research we can collected data from Amazon product data.

The dataset consists of amazon product review and product metadata:

- Amazon product review consist of 233.1 million data consisting of ratings, text and votes.
- Product metadata for each review consisting information like Product id, Title, Price, Image Url, Related Product, Sales Rank, Brand, Categories.

Sample metadata:


```
{
  "asin": "0000031852",
  "title": "Girls Ballet Tutu Zebra Hot Pink",
  "price": 3.17,
  "imUrl": "http://ecx.images-
amazon.com/images/I/51fAmVkTbyL._SY300_.jpg",
  "related":
  {
    "also_bought": ["B00JHONN1S", "B002BZX8Z6", "B00D2K1M3O",
"0000031909", "B00613WDTQ", "B00D0WDS9A", "B00D0GCI8S", "0000031895",
"B003AVKOP2", "B003AVEU6G", "B003IEDM9Q", "B002R0FA24", "B00D23MC6W",
"B00D2K0PA0", "B00538F5OK", "B00CEV86I6", "B002R0FABA", "B00D10CLVW",
"B003AVNY6I", "B002GZGI4E", "B001T9NUFS", "B002R0F7FE", "B00E1YRI4C",
"B008UBQZKU", "B00D103F8U", "B007R2RM8W"],
    "also_viewed": ["B002BZX8Z6", "B00JHONN1S", "B008F0SU0Y",
"B00D23MC6W", "B00AFDOPDA", "B00E1YRI4C", "B002GZGI4E", "B003AVKOP2",
"B00D9C1WBM", "B00CEV8366", "B00CEUX0D8", "B0079ME3KU", "B00CEUWY8K",
"B004FOEEHC", "0000031895", "B00BC4GY9Y", "B003XRKA7A", "B00K18LKX2",
"B00EM7KAG6", "B00AMQ17JA", "B00D9C32NI", "B002C3Y6WG", "B00JLL4L5Y",
"B003AVNY6I", "B008UBQZKU", "B00D0WDS9A", "B00613WDTQ", "B00538F5OK",
"B005C4Y4F6", "B004LHZ1NY", "B00CPHX76U", "B00CEUWUZC", "B00IJVASUE",
"B00GOR07RE", "B00J2GTM0W", "B00JHNSNSM", "B003IEDM9Q", "B00CYBU84G",
"B008VV8NSQ", "B00CYBULSO", "B00I2UHSZA", "B005F50FXC", "B007LCQI3S",
"B00DP68AVW", "B009RXWNSI", "B003AVEU6G", "B00HSOJB9M", "B00EHAGZNA",
"B0046W9T8C", "B00E79VW6Q", "B00D10CLVW", "B00B0AVO54", "B00E95LC8Q",
"B00GOR92SO", "B007ZN5Y56", "B00AL2569W", "B00B608000", "B008F0SMUC",
"B00BFXLZ8M"],
    "bought_together": ["B002BZX8Z6"]
  },
  "salesRank": {"Toys & Games": 211836},
  "brand": "Coxlures",
  "categories": [["Sports & Outdoors", "Other Sports", "Dance"]]
}
```


Were,

- asin - ID of the product, e.g., 0000031852
- title - name of the product
- price - price in US dollars (at time of crawl)
- imUrl - url of the product image
- related - related products (also bought, also viewed, bought together, buy after viewing)
- salesRank - sales rank information
- brand - brand name
- categories - list of categories the product belongs to.

As, part of this research work we have extracted data for below 8 categories with 5000 images in each category:

- Cell Phones and Accessories

- Clothing Men
- Clothing Women
- Electronics
- Home and Kitchen
- Pet Supplies
- Shoes
- Watches

Download dataset from location: Extracted Amazon Product Data referenced cited in (Stewart, 2012), (McAuley et al., 2015) and (Chen et al., 2015). After data is downloaded extract all the csv file creates data frame with column:

```
Index (['asin', 'related', 'title', 'price', 'salesRank', 'imUrl', 'brand', 'categories',
 'description', 'rating', 'timestamp', 'imUrl_local', 'filename', 'user'], dtype='object'
```

Size of data: (40000, 14)

### 4.3 Data Preparation

Data preparation is the process of cleaning and transforming raw in such a way machine learning algorithms can be applied on that using which model for classification can be designed. Data preparation is one of the phases of ML lifecycle. Once the data is gathered from data source, data is cleaned and transformed into real-time machine learning projects to find the insights and predict the future.

Data preparation is a process of gathering, combining, cleaning, and transforming raw data, that way we can call data preparation as “data pre-processing”, “data cleaning” and “feature engineering”.

When working with data in data preparations step there are few essential tasks:

- Data Cleaning: This task entails spotting mistakes and fixing them.
- Feature Selection: Identify the important variable in the data required for the model.

- Data transforms: Convert raw data into the format suited for the model.
- Feature Engineering: This task involves driving new variables from dataset required for modelling.
- Dimensionality Reduction: Without altering the data, the dimensionality reduction procedure involves translating features from higher dimensions to lower ones.

### 4.3.1 Data Cleaning

As the same suggest data cleaning is nothing but fixing bad data in the dataset. The bad data can be empty cell, data is wrong format, wrong data or duplicate data.

Remove duplicate data

To remove duplicate data, we will use two function of pandas dataframe:

1. duplicated (self [, subset, keep]) - Return bool denoting duplicate rows.

```
# Validate duplicate value
isDuplicate = data.duplicated().any()
```

2. drop_duplicates(self[, subset, keep, inplace]) – Return dataframe by removing duplicate values.

```
# Drop duplicate values
data.drop_duplicates(keep=False, inplace=True)

# Drop duplicate products
data.sort_values("asin", inplace=True)
data.drop_duplicates(subset="asin", keep=False, inplace=True)
```

Fill Empty Cell

To fill empty cell, we first need to validate the column having empty or null value.

```
# Checking the unique observations, datatype & null values for every feature
d={"Feature":[i for i in data.columns],"Nunique" :data.nunique().values ,'Type' : data.dt
ypes.values, "No: of nulls" : data.isnull().sum() }
description = pd.DataFrame(data = d)
```

```
description
```

| | Feature | Nunique | Type | No: of nulls |
|---|---|---|---|---|
| **asin** | asin | 39367 | object | 0 |
| **title** | title | 39233 | object | 0 |
| **price** | price | 7598 | float64 | 0 |
| **imUrl** | imUrl | 39366 | object | 0 |
| **brand** | brand | 4108 | object | 23861 |
| **rating** | rating | 5 | float64 | 297 |
| **filename** | filename | 8 | object | 0 |

Table 4. 1:Unique value, data type, and number of null values of each label

In the current data we can see brand column have 23861 null value and rating column have 297 null values. Since brand is of type object and rating is float, we will replace brand null value with "Unknown" and rating null value to 0.

```
data['brand'].fillna(value = "Unknown" , inplace = True)
data['rating'].fillna(value = 0 , inplace = True)
```

Also, as part of data cleaning process we will identify and consider the data having images of the product. For the we tried to download image of the data from 'imUrl' to create image dataset.

While downloading the image if we face any issue, error or exception with the download url we will remove the product id for which this issue is observed from the dataset. This is to make sure the product dataset has all the valid images. This helps in having better quality of data and further more improve helps in model building.

```
# Download image and save image to create image dataset
from PIL import Image
error_id = []

def save_img(img,row):
    img.save('Data\Images\Image_'+ str(row['product_id'])+'.jpeg')

# Downlaod images, run once
for idx, row in tqdm(data.iterrows()):
    try:
```

```
        url = row['imUrl']
        response = requests.get(url)
        img = Image.open(BytesIO(response.content))
        if img.mode in ['RGBA', 'P']:
            img = img.convert('RGB')
        save_img(img,row)
    except:
        error_id.append(row['product_id'])
        print("Error in : \n%s", str(row['product_id']))

# Reduce the data from dataframe for which image is not present
for eid in error_id:
    data.drop(data[data['product_id'] == eid].index, inplace = True)
```

### 4.3.2 Feature Selection

The process of feature selection involves keeping the dataset's existing features while either including the crucial elements or removing the unnecessary ones. It is the process of choosing, either automatically or manually, the subset of features that are most pertinent and appropriate to be employed in model construction.

Need for feature selection:

In order to achieve better results with machine learning, a high-quality input dataset must be provided. To improve our model's learning and training, we gather a tonne of data. The dataset typically comprises of noisy data, useless data, and some usable data in small amounts. Additionally, the enormous volume of data slows down the model's training process, and with noise and irrelevant input, the model might not predict correctly or perform well. Therefore, it is imperative to eliminate these noises and less-important data from the dataset. To do this, feature selection techniques are employed.

Benefits of feature selection:

- By doing so, the dimensionality curse is avoided.
- It assists in simplifying the model so that researchers can quickly analyse it.
- It cuts down on training time.
- It lessens overfitting, which improves generalisation.

Our dataset comprises of below column:

```
'asin', 'related', 'title', 'price', 'salesRank', 'imUrl', 'brand', 'categories', 'descri
ption', 'rating', 'timestamp', 'imUrl_local', 'user'
```

For our research we will consider below column:

```
'product_id', 'asin', 'title', 'price', 'imUrl', 'brand', 'category'
```

### 4.3.3 Data Transformation

It's challenging to track or comprehend raw data. Because of this, it needs to be pre-processed before any information can be extracted from it. Without altering the substance of the datasets, data transformation is the technological process of converting data from one format, standard, or structure to another. This is often done to make the data more usable by users or apps or to improve the quality of the data.

Modifying the format, structure, or values of data is referred to as data transformation and the quality of machine learning models depends on the data used to train them. It is essential for training data to be presented in a way that promotes learning and generalisation.

Data transformation in machine learning can be done manually or using automation tool. There are various automation tool using which data transformation can be done:

- IBM Infosphere
- SAP Data Service
- Dataform
- Azure Data Factory
- Qlik Compose
- Data build tool

For our dataset we will consider manual data transformation by validating and converting dataset columns to below type:

```
# Checking the datatype of dataset column
d = {"Feature":[i for i in data.columns] ,'Type' : data.dtypes.values}
description = pd.DataFrame(data = d)
description
```

| | Feature | Type |
|---|---|---|
| 0 | product_id | int64 |
| 1 | asin | object |
| 2 | title | object |
| 3 | price | float64 |
| 4 | imUrl | object |
| 5 | brand | object |
| 6 | rating | int64 |
| 7 | category | object |

Table 4. 2: Data Type of all label in dataset

### 4.3.4 Feature Engineering

In general, input data is used by all machine learning algorithms to produce output. The input data continues to be presented in a tabular format with rows for instances or observations and columns for variables or attributes; these attributes are frequently referred to as features.

The pre-processing stage in machine learning known as feature engineering is used to turn raw data into features that may be utilised to build a prediction model using either machine learning or statistical modelling. It aids in better communicating a fundamental issue to predictive models, increasing the model's accuracy for unobserved data. The feature engineering method chooses the most practical predictor variables for the model, which is composed of predictor variables and an outcome variable.

Here are a few advantages of feature engineering in machine learning:

- It helps prevent the dimensionality curse.
- The model is made simpler as a result, making it simpler for researchers to interpret.
- It cuts down on training time.
- It improves generalisation by reducing overfitting.

While working with extracted amazon product dataset we have extracted two new features:
Category – Category are the different types of products in the data set as shown below

- Electronics

- Pet Supplies
- Home and Kitchen
- Cell Phones and Accessories
- Clothing Men
- Clothing Women
- Watches
- Shoes

In our dataset file we will be using filename as category of the product.

```
# Rename filename with category name.
data.columns = data.columns.str.replace('filename', 'category')

# Replace '_' with ' '
data['category'] = data['category'].str.replace('_',' ')
```

- Product id is the index id of the dataset.

```
# Reset the index back to the default 0, 1, 2 etc indexes
data=data.reset_index()
# Drop index column
data.drop(['index'], axis=1, inplace=True)
# Add 'product_id' column
data.insert(loc=0, column='product_id', value=np.arange(len(data)))
```

### 4.3.5 Dimensionality Reduction

There are frequently too many factors used in machine learning classification challenges to make the final categorization. In essence, these elements are features or variables. It becomes more difficult to visualise the training set and subsequently work on it as the number of features increases. Sometimes, the majority of these traits are redundant since they are connected. Algorithms for dimensionality reduction are used in this situation.

Dimensionality reduction reduces the number of random variables without altering the data and dimensionality reduction features from higher dimensions to lower ones.

While working with Amazon Extracted dataset we will consider below columns for our research:

```
‘product_id’, 'asin', 'title', 'price', 'imUrl', 'brand', ‘category’
```

Hence, we will drop columns for:

```
‘related’, ‘salesRansk’, 'categories', 'description', 'timestamp', 'imUrl_local', 'user’
```

```
# Remove related, salesRank, timestamp, user, imUrl_local, description columns
data.drop(['related'], axis=1, inplace=True)
data.drop(['salesRank'], axis=1, inplace=True)
data.drop(['timestamp'], axis=1, inplace=True)
data.drop(['user'], axis=1, inplace=True)
data.drop(['description'], axis=1, inplace=True)

# We will remmove categories column as we will replace file name as categories of the pro
duct
data.drop(['categories'], axis=1, inplace=True)
```

### 4.4 Data Visualization

The graphical depiction of information and data in a pictorial or graphical manner is known as data visualisation (Example: charts, graphs, and maps). Tools for data visualisation offer a simple approach to spot and comprehend trends, data patterns, and outliers. Tools and methods for data visualisation are crucial for processing vast volumes of data and making data-driven decisions. The idea of utilising visuals to comprehend data has been around for a very long time. Charts, tables, graphs, maps, and dashboards are the standard types of data visualisation.

Benefit of Data Visualization:

- It helps in identification and discovery of trends in the data.
- Provides a perspective in the data.
- Helps in putting the data in correct context.
- Data visualization, helps in saving lot of time.

In industry there are various data visualization tool like Tableau, Looker, Zoho Analytics, Sisense, IBM Cognos Analytics, Qlik Sense, Domo, Microsoft Power BI, Klipfolio, SAP Analytics Cloud.

Categories of Data Visualization

Data visualisation is essential to market research because it allows for the visualisation of both numerical and categorical data, increasing the impact of insights and lowering the danger of analysis paralysis. Thus, the following categories for data visualisation exist:

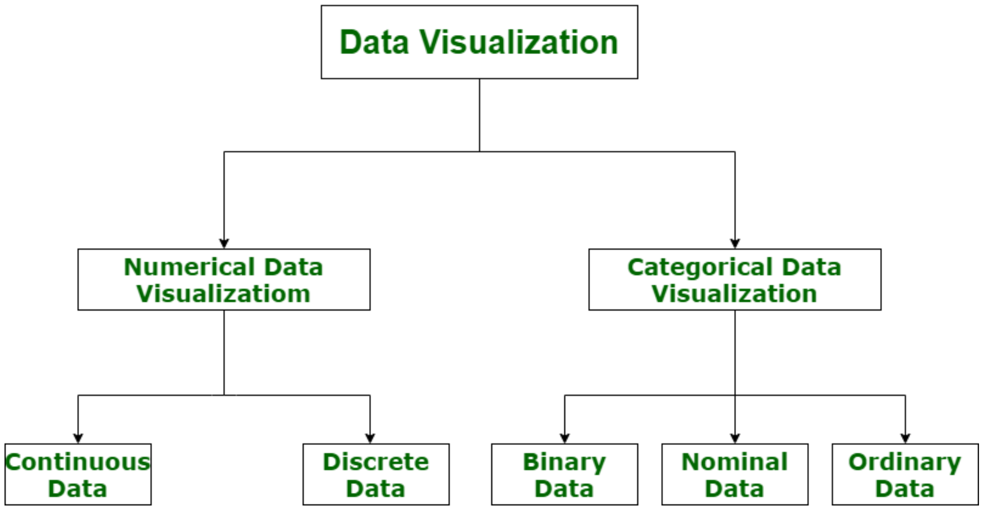


Figure 4. 1: Category of data visualization

### 4.4.1 Numerical Data Visualization

The simplest method of visualising data is with numbers. It is typically employed to assist individuals in understanding enormous amounts of raw data and numbers in a way that makes it simpler to grasp and put into practise. In research, we have done numerical data visualization by:

1. Extracting the information for each column by using the describe () function.

The describe () method returns the description of the data of the dataframe. i.e.

- count - Number of values that are not empty
- mean - The mean value, or average.
- Std – Standard deviation of the data.
- Min – minimum value.
- 25% - Value less than 25% percentile.

- 50% - Value less than 50% percentile.
- 75% - Value less than 75% percentile.
- Max – Maximum value

| | product_id | price | rating |
|---|---|---|---|
| **count** | 39194.00 | 39194.00 | 39194.00 |
| **mean** | 19620.51 | 58.61 | 4.64 |
| **std** | 11348.01 | 99.79 | 0.76 |
| **min** | 0.00 | 0.01 | 0.00 |
| **25%** | 9799.25 | 10.00 | 4.00 |
| **50%** | 19597.50 | 22.41 | 5.00 |
| **75%** | 29402.75 | 60.88 | 5.00 |
| **max** | 39366.00 | 999.99 | 5.00 |

Table 4. 3 Detail information of values in price and rating column.

2. Plotting density per rating given to the product in the dataset:

Below plot (figure 4.2) signifies the rating density in the dataset where we can see that product have rating 5.0 is more as compared to the product 4.0 and there are very few products with rating of 0, 1, 2 and 3.

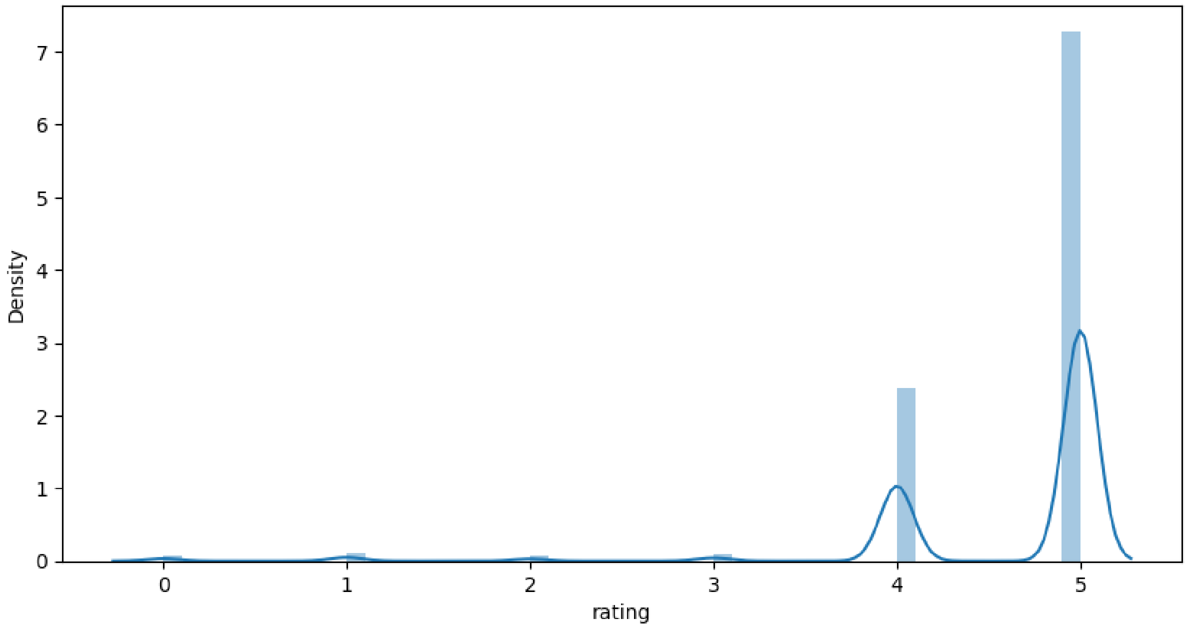


Figure 4. 2: Density per rating of product in dataset

3. Plotting density per price of the product in the dataset:

Below plot (figure 4.3) signifies the price density in the dataset where we can see most of the are in price range of $0 -$200 with highest number of product ranges around 0 -25.

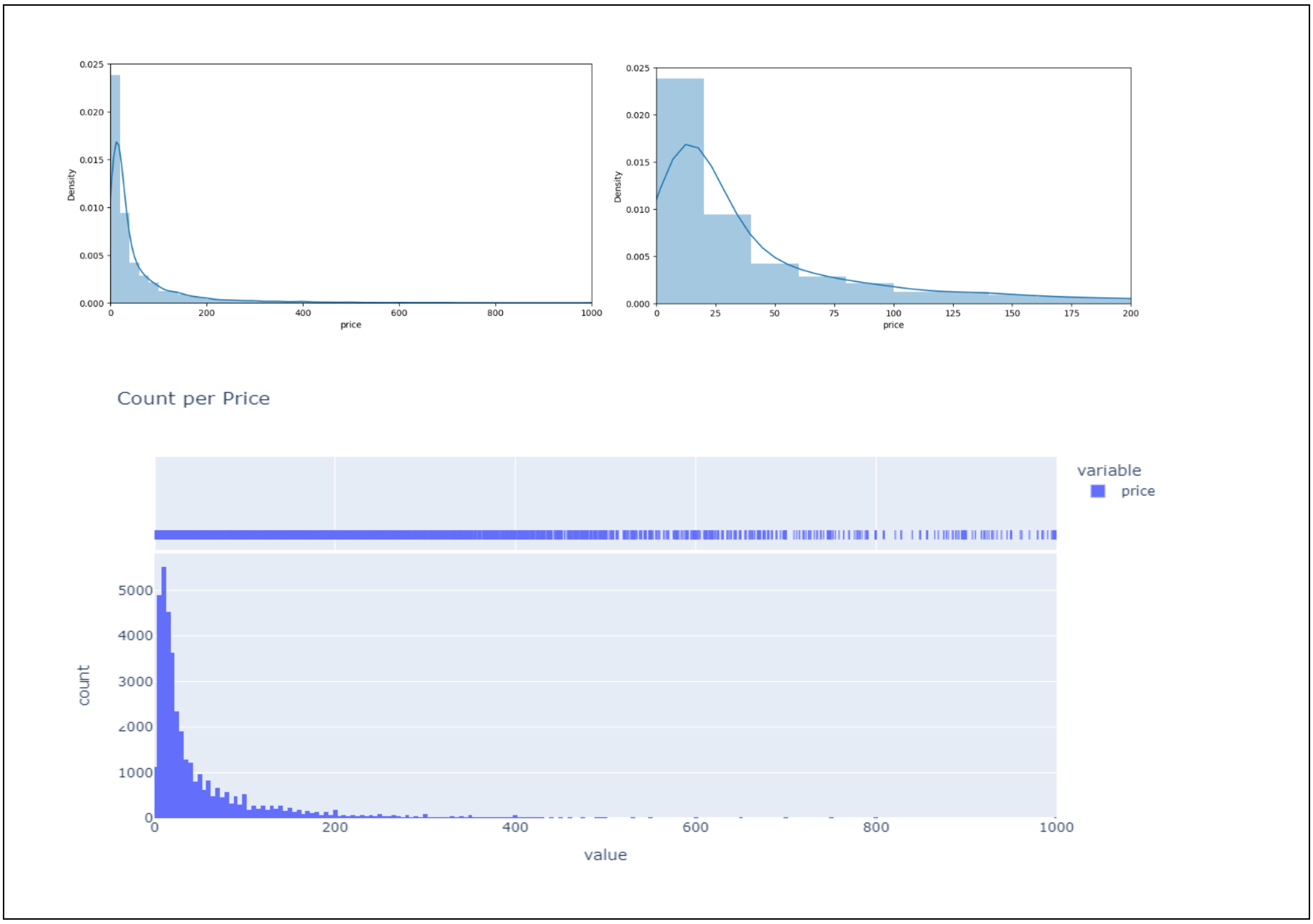


Figure 4. 3 : Density per price of product in dataset

Thus, from numerical data visualization we can infer:

Product Price - There are products starting from 0.0 dollars to 999.99 dollars, so this is a huge range in the dataset. The price mean is $58.54 across different category.

Product Rating - On an average product rating is 4.64. That means the product present in the dataset on an average are quality products.

### 4.4.2 Categorical Data Visualization

Any type of data that commonly depicts groupings is considered categorical data. It merely comprises of categorical variables that serve as a representation of properties. The main goals of categorical data visualisation are to illustrate important concepts, make connections, and provide context.

Below figure 4.4 is the count of product in each category describing that in each six categories i.e., Pet Supplies, Home and Kitchen, Electronics, Cell Phone and Accessories, Watches and Shore there are 5000 products and in Clothing Men and Clothing women there are more than 4500 products.

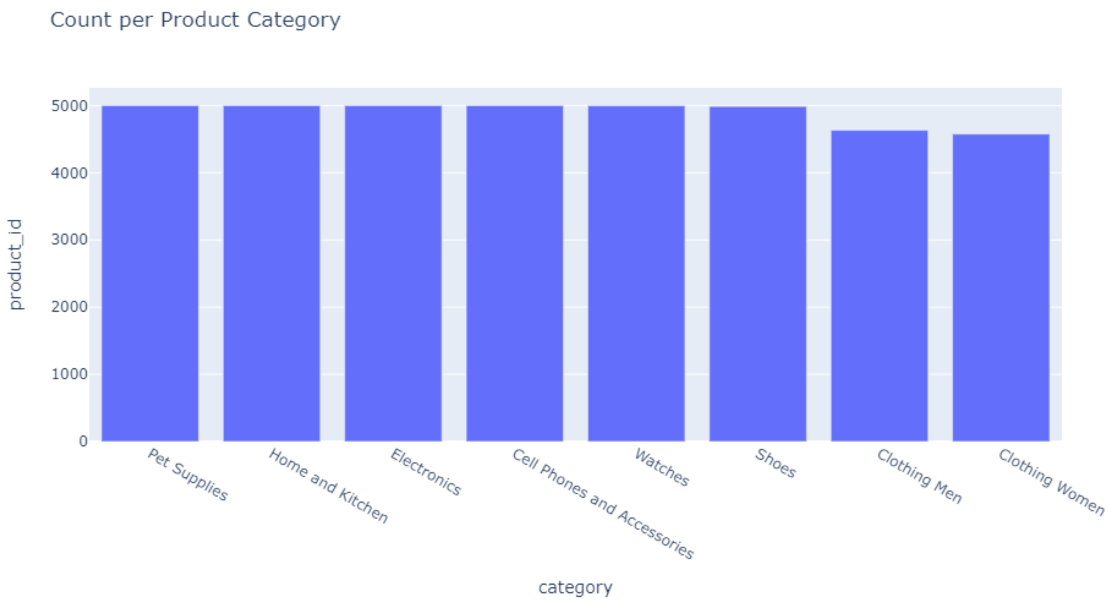


Figure 4. 4: Count of products in each category.

Collaboratively there are more than 39000 products in the dataset, with more than 4500 in each category, figure 4.5 illustrates the product sample of each category and visualisation of a product image of all category present in the dataset.


Figure 4. 5: Sample Products in each category

### 4.4.3 Word Cloud

The Wordcloud library in Python is a popular library for creating word clouds. A word cloud is a visual representation of text data, where the size of each word is proportional to its frequency in the text. It is a data visualisation technique used to depict text data, shows its frequency or relevance. Using a word cloud, significant textual data points may be emphasised.

In this research we captured product tile and displayed like a product image of each category. This help in depicting the frequency of word present in the product tile. To display word cloud, we have used Wordcloud python library.

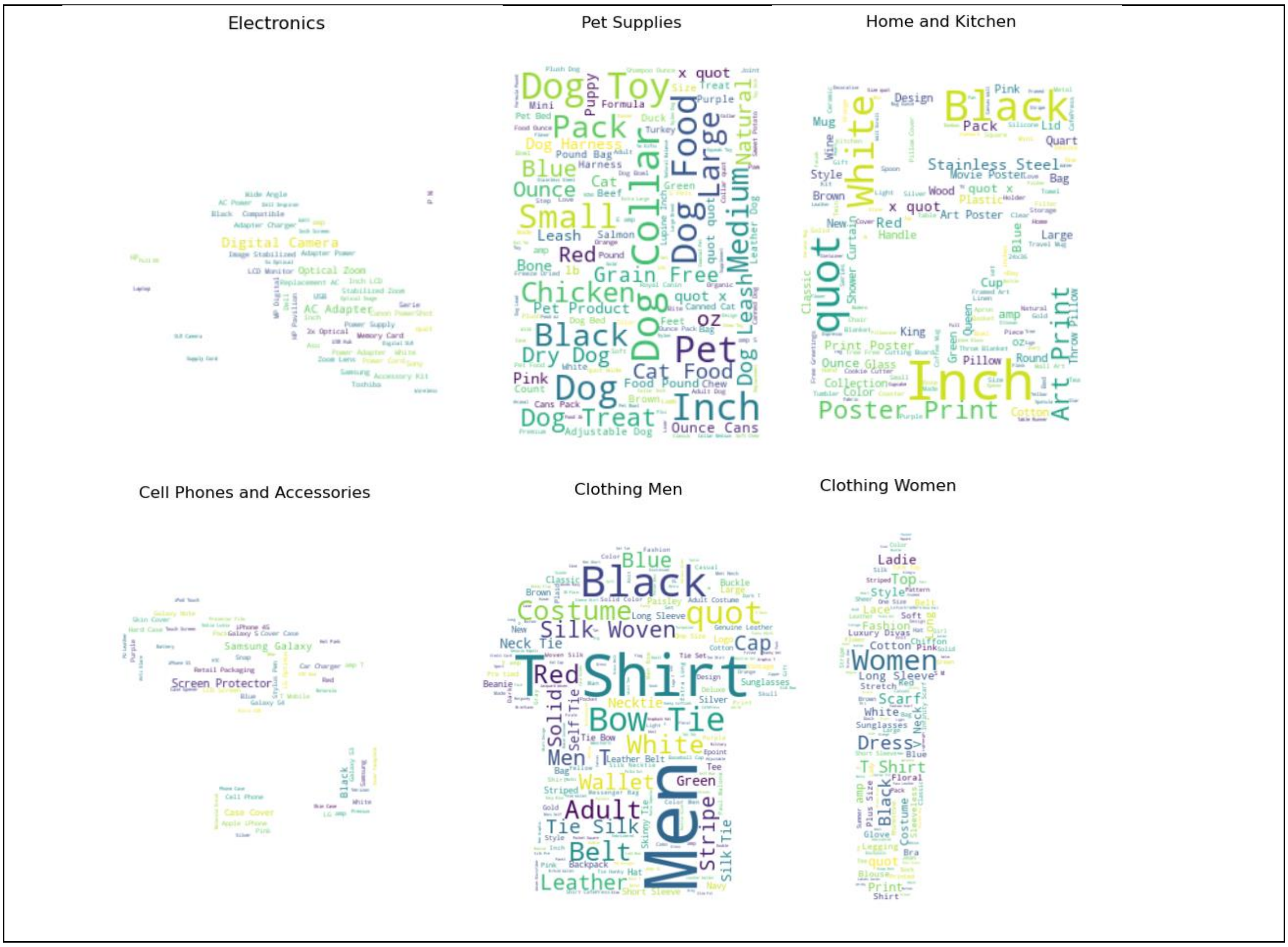


Figure 4. 6 : Wordcloud of product title and product category

**4.5 Summary**

Data sourcing, data cleaning and data visualization is most important and essential part in any machine learning research work. Where data sourcing heles in collecting data from various sources, data cleaning helps in making dataset ready for modelling and data visualization helps in infaring dataset in graph, plot, chart, table.

For this research, we have extracted data from [Amazon product data](), in 8 different categories consisting of 233.1 million data. Below is the category for which we have extracted data with 5000 images in each category.

- Electronics
- Pet Supplies
- Home and Kitchen

- Cell Phones and Accessories
- Clothing Men
- Clothing Women
- Watches
- Shoes

Thus, combining up to extract dataset with 40000 records, consisting of below columns:

```
'asin', 'related', 'title', 'price', 'salesRank', 'imUrl', 'brand', 'categories', 'descri
ption', 'rating', 'timestamp', 'imUrl_local', 'user'
```

In this research since we will work on image data, we will download image from imUrl and remove data row for which image is not present in data set resulting in 39194 final records after data cleaning and dimensionality reduction.

After feature selection and feature engineering below columns are selected for further processing:

```
'product_id', 'asin', 'title', 'price', 'imUrl', 'brand', 'category'
```

# CHAPTER 5: IMPLEMENTATION, EXPERIMENTS AND RESULT EVALUATION

## 5.1 Introduction

This chapter presents the result evaluated by various experiments proposed in the Chapter 3 i.e., research methodology section, by passing the data extracted from data source and cleaned as explained in Chapter 4 i.e., Analysis chapter of the research.

In this chapter we will pass the dataset created / extracted and perform experiment to drive the best model suited for image recommendation system bot.

The different experiment involves creating and identifying the best model for creating image recommendation system.

1) Image Classification - Create model to identify the correct category of the product from random image.
2) Text Based Recommendation – Recommend product based on product title.
3) Model Identification – Identify the best model suited for image recommendation system.
4) Image Based Recommendation System – Based on best identified model develop image-based recommendation system.

## 5.2 Image Classification using CNN

Image classification involves extraction of features from the product image to observe the trend / pattern in the dataset, in this research we will use one the most popular algorithm in ML for image classification known as CNN or convolution Neural Network.

In this experiment we created an image classifier that can distinguish whether the given image or random image or the product belong to which category i.e., Electronics, Pet Supplies, Home and Kitchen, Cell Phones and Accessories, Clothing Men, Clothing Women, Watches and to achieve this we will use CNN.

After the analysis done in Chapter 4, the dataset of 39194 image is created. All the images have corresponding label assigned to it. We will first convert all the label / category variable to number this encoding is done for the machine to understand the labels.

```
# Converting the response variable into numbers
data['category'][data['category']=='Electronics' ]=0
data['category'][data['category']=='Pet Supplies' ]=1
data['category'][data['category']=='Home and Kitchen' ]=2
data['category'][data['category']=='Cell Phones and Accessories' ]=3
data['category'][data['category']=='Clothing Men' ]=4
data['category'][data['category']=='Clothing Women' ]=5
data['category'][data['category']=='Watches' ]=6
data['category'][data['category']=='Shoes' ]=7
data['category']=data['category'].astype(int)
```

Once the encoding is done, we will create array of the image by reducing the size to 120. In this experiment we have reduce the size of image to 120 because of the computation power or limitation of the system.

```
# Reading the training images
train_image = []
for i, row in tqdm(data.iterrows()):
    img = image.load_img('Data\Images\Image_'+ str(row['product_id'])+'.jpeg', target_siz
e=(120,120,1), grayscale=True)
    img = image.img_to_array(img)
    img = img/255
    train_image.append(img)
X = np.array(train_image)
```

Create the target variable which would be category of the data and split data into train and test.

```
# Creating the target variable
y = data['category'].values
y = to_categorical(y)
# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=42, test_size=0.25
)
```

Were,

- Random_state: controls how the data is shuffled before the split is implemented. If you want consistent results from several function calls, pass an int
- test_size: reflects the percentage of the dataset to include in the train split and range between 0.0 and 1.0.

Thus, creating train data of 29395 record and test data of 9799.

After splitting the data, we will normalize the train and test data before creating and passing to CNN module which uses filters on raw pixel of product image to learn pattern and compare with the global patten. In this research we have created CNN layer having:

1. Six convolutional Conv2D layers with 32, 32, 64, 64, 128, 128 filters respectively with ReLU activation function.
2. Followed by batch normalisation to improve the performance.
3. Followed by max pooling layer which decreases size of input by taking the max value of sub-matrix.
4. Added dropout regularization of 0.20, to drop few neurons
5. And then flatten the output layer by simply taking (13, 13, 128) output of the previous layer and 'flattens' it into a vector of length 13 x 13 x 128 = 21632.
6. Finally dense layer with 8 neurons represents 8 categories of the products with output of 4104 parameters.

Thus, the total number of parameters are 11,368,424 all of which are trainable and 896 non trainable parameters.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 120, 120, 32)      320

 activation (Activation)     (None, 120, 120, 32)      0

 batch_normalization (BatchN  (None, 120, 120, 32)     128
 ormalization)

 conv2d_1 (Conv2D)           (None, 118, 118, 32)      9248

```

```
 activation_1 (Activation)   (None, 118, 118, 32)      0

 batch_normalization_1 (Batc  (None, 118, 118, 32)     128
 hNormalization)

 max_pooling2d (MaxPooling2D  (None, 59, 59, 32)       0
 )

 dropout (Dropout)           (None, 59, 59, 32)        0

 conv2d_2 (Conv2D)           (None, 59, 59, 64)        18496

 activation_2 (Activation)   (None, 59, 59, 64)        0

 batch_normalization_2 (Batc  (None, 59, 59, 64)       256
 hNormalization)

 conv2d_3 (Conv2D)           (None, 57, 57, 64)        36928

 activation_3 (Activation)   (None, 57, 57, 64)        0

 batch_normalization_3 (Batc  (None, 57, 57, 64)       256
 hNormalization)

 max_pooling2d_1 (MaxPooling  (None, 28, 28, 64)       0
 2D)

 dropout_1 (Dropout)         (None, 28, 28, 64)        0

 conv2d_4 (Conv2D)           (None, 28, 28, 128)       73856

 activation_4 (Activation)   (None, 28, 28, 128)       0

 batch_normalization_4 (Batc  (None, 28, 28, 128)      512
 hNormalization)

 conv2d_5 (Conv2D)           (None, 26, 26, 128)       147584

 activation_5 (Activation)   (None, 26, 26, 128)       0

 batch_normalization_5 (Batc  (None, 26, 26, 128)      512
 hNormalization)

 max_pooling2d_2 (MaxPooling  (None, 13, 13, 128)      0
 2D)

 dropout_2 (Dropout)         (None, 13, 13, 128)       0

 flatten (Flatten)           (None, 21632)             0

 dense (Dense)               (None, 512)               11076096
```

```
 activation_6 (Activation)    (None, 512)              0

 dropout_3 (Dropout)          (None, 512)              0

 dense_1 (Dense)              (None, 8)                4104

 activation_7 (Activation)    (None, 8)                0

=================================================================
Total params: 11,368,424
Trainable params: 11,367,528
Non-trainable params: 896
_________________________________________________________________
```

Model Summary of CNN Model

On this CNN model, we trained our 29395 image and achieved below accuracy and loss in 50 epochs:

| | |
|---|---|
| Training Accuracy | 55.86% |
| Train Loss | 9.71 |
| Validation Accuracy | 52.90% |
| Validation Loss | 9.83 |
| Testing Accuracy | 52.90 |
| Testing Loss | 9.83 |

Table 5. 1: CNN Accuracy and Loss

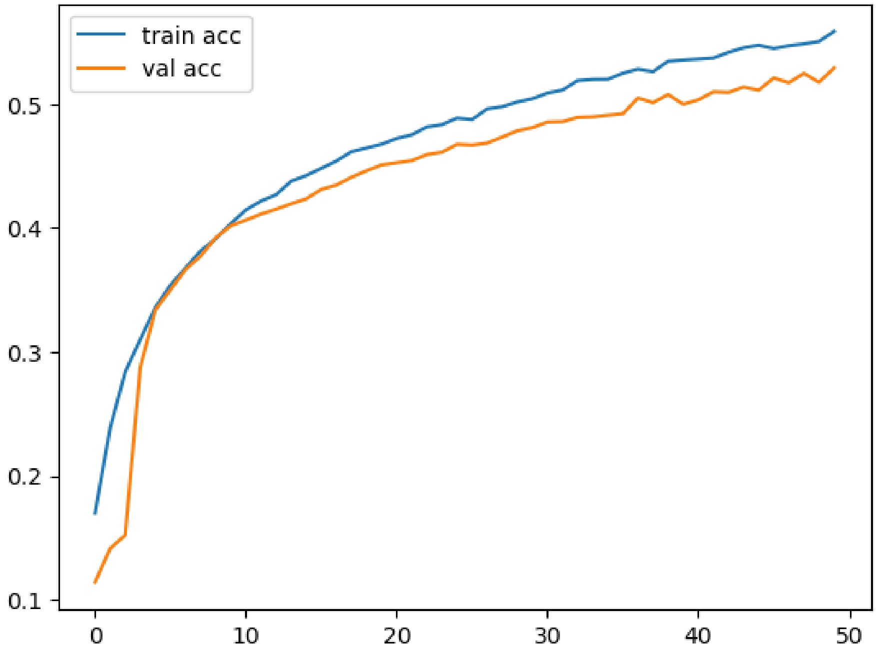


Figure 5. 1: CNN Model Accuracy.

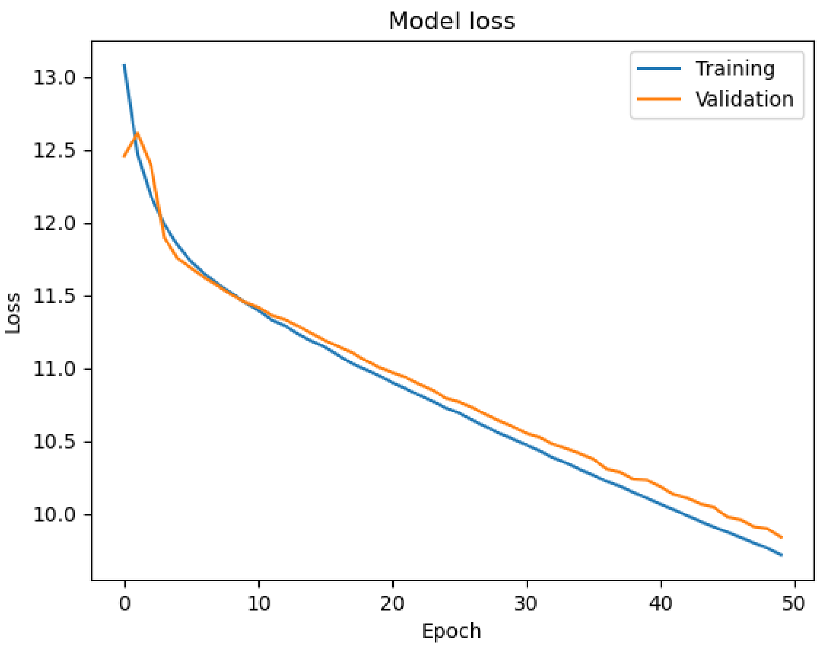


Figure 5. 2: CNN Model loss

Below is the confusion matrix of test data, From the matrix it looks our model is classifying more or less all the categories correctly.

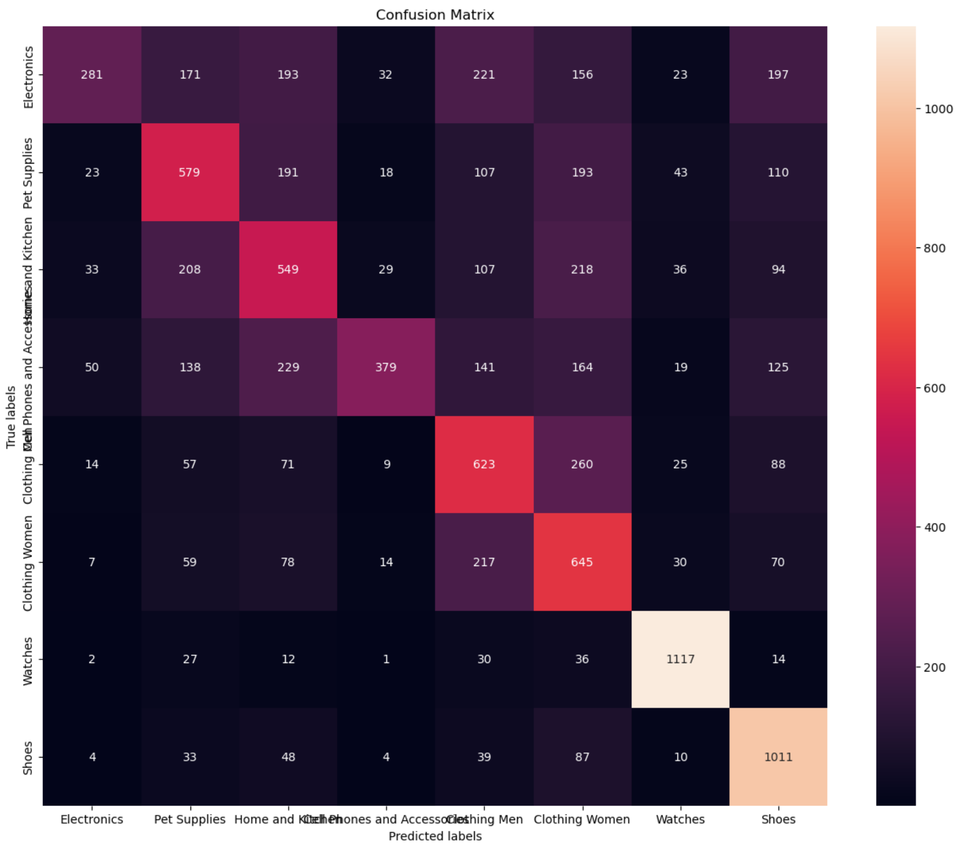


Figure 5. 3: Confusion matrix on test data

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Electronics | 0.68 | 0.22 | 0.33 | 1274 |
| Pet Supplies | 0.46 | 0.46 | 0.46 | 1264 |
| Home and Kitchen | 0.40 | 0.43 | 0.42 | 1274 |
| Cell Phones and Accessories | 0.78 | 0.30 | 0.44 | 1245 |
| Clothing Men | 0.42 | 0.54 | 0.47 | 1147 |
| Clothing Women | 0.37 | 0.58 | 0.45 | 1120 |
| Watches | 0.86 | 0.90 | 0.88 | 1239 |
| Shoes | 0.59 | 0.82 | 0.69 | 1236 |
| accuracy | | | 0.53 | 9799 |
| macro avg | 0.57 | 0.53 | 0.52 | 9799 |
| weighted avg | 0.57 | 0.53 | 0.52 | 9799 |

Table 5. 2: Classification Report for CNN Model

Above is the performance evaluation metrics, it presents the model's precision, recall, F1 score and support and provide the better understanding of overall classification model.

From above classification report we can infer that model is more precisely able to classify product belonging to Electronics, Cell Phones and Accessories, Watches and shoes categories compared to other category of products.

In this experiment, we have created CNN module where we have achieved accuracy of 55.86% but we have seen there is a significant loss. To validate the model prediction, we will pass random 8 image and see the prediction vs actual category of the random product image. Below are the images of product depicting Predicted vs Actual category of product on random 8 image. We can see due to high loss one of the products which is "Pet Supplies", model is predicting it as "Clothing Women".


Figure 5. 4: Predicted Vs Actual category on random 8 images.

### 5.3 Text Based Recommendation

Text based recommendation is content based recommendation system, where in this experiment objective is to recommend n product based on product title.

The data of text-based recommendation system is the same dataset prepared in chapter 4, but we need to some further cleaning to the product tile because in the dataset we have seen some long text. So, we need to reduce the product tile word length to 100 words, this is because in the dataset we have seen some product tile with extra text.

```
data['title'] = data['title'].astype(str).apply(lambda x: x[:100])
```

Split the data to train test and transform the train product title to numerical data using CountVectorizer. CountVectorizer helps text data to use directly in machine learning models by removing all the stop words and convert the text to lowercase.

```
# Initialize the CountVectorizer
count_vectorize = CountVectorizer(stop_words = 'english')
# Run the CountVectorizer on the text
title_vectorized = count_vectorize.fit_transform(train_product['title'])
```

After transforming the product title to vectorized form we got 32560 unique words for 29395 products.

We have used cosine similarity to calculate and determine the similarity between the product tiles.

```
cosine_sim = cosine_similarity(title_vectorized, title_vectorized)
```

Based on this similarity score we have designed recommendation system which takes product id and returns N recommendation. This similarity score is arranged in descending order and results are displayed based on score.

```
def text_recommendation(pid, num_recommend=5):
    recommended_prod = []
    score=[]
    print("---------------------------------------------------------------------")
    print("Original product:")
    print("---------------------------------------------------------------------")
    print("Product ID : " , pid)
    print("Title : ", train_product['title'][train_product['product_id']==pid].tolist()[0
].split(' - ')[0])
    print("Brand : ", train_product['brand'][train_product['product_id']==pid].tolist()[0
].split(' - ')[0])

    # creating a Series with the similarity scores in descending order
    score_series = pd.Series(cosine_sim[pid]).sort_values(ascending = False)

    # getting the indexes and scores of the N most similar products
    top_10_indexes = list(score_series.iloc[1:(num_recommend+1)].index)
    top_10_score=list(score_series.iloc[1:(num_recommend+1)])

    # Displaying the recommended products- PID, Name, Brand and Similarity Score
    print("\n")
    print("--------------------------------------------------------------------")
    print("Most similar products:")
    print("-----------------------------------------------------------------------")

    for i in range(0,len(top_10_score)):
        recommended_prod.append(list(train_product['title'])[i])
        print("\nProduct ID : " , top_10_indexes[i])
        print("Title : ", train_product['title'][train_product['product_id']==top_10_inde
xes[i]].tolist()[0].split(' - ')[0])
        print("Brand : ", train_product['brand'][train_product['product_id']==top_10_inde
xes[i]].tolist()[0].split(' - ')[0])
        print("Similarity score : ",top_10_score[i])
```

Example output of text-based recommendation system:

```
-----------------------------------------------------------------------
Original product:
-----------------------------------------------------------------------
Product ID :  23967
```

```
Title :  KONG Funsters Flip Dog Toy, X-Small
Brand :  KONG

-----------------------------------------------------------------------
Most similar products:
-----------------------------------------------------------------------

Product ID :  14105
Title :  Yonex Women's Tennis Shoe
Brand :  Unknown
Similarity score :  0.6030226891555273

Product ID :  18626
Title :  Obama
Brand :  Unknown
Similarity score :  0.5892556509887895

Product ID :  18087
Title :  Hannspree HL HL161ABB 16-Inch Screen LED-Lit Monitor
Brand :  Hannspree
Similarity score :  0.5892556509887895

Product ID :  2649
Title :  Wardley Tropical Flakes
Brand :  Wardley
Similarity score :  0.5892556509887895
Product ID :  826
Title :  Iams Smart Puppy (Formerly Puppy Original Formula)
Brand :  Iams
Similarity score :  0.5555555555555556
```

### 5.4 Models of Recommendation System

Till now we have designed CNN model which can be used for image-based recommendation system but we have seen that accuracy of CNN model is moderate and loss is on the higher side. Thus, we need to explore more and find model with high accuracy and less or minimum loss.

In this section we will validate the best model suited for image recommendation system, we will experiment our dataset (prepared after chapter 4) on various model and identify the appropriate model.

### 5.4.1 ResNet-50

ResNet-50 is a CNN that is 50 layers deep consisting of 5 stages each with a convolution and identity block which each block convolution block has 3 convolution layers and each identity block has 3 convolution layers creating total 24,390,536 parameters with 802,824 trainable parameter and 23,587,712 non-trainable parameters.

In this experiment we have 29395 train data and 9799 test data corresponding to 8 class with image size of (224,224). After training ResNet-50 model on dataset, we have achieved below accuracy and loss in 10 epochs.

| | |
|---|---|
| Training Accuracy | 56.72% |
| Train Loss | 1.76 |
| Validation Accuracy | 45.47% |
| Validation Loss | 2.3372 |
| Testing Accuracy | 53.86% |
| Testing Loss | 2.00 |

Table 5. 3: ResNet-50 Accuracy and Loss

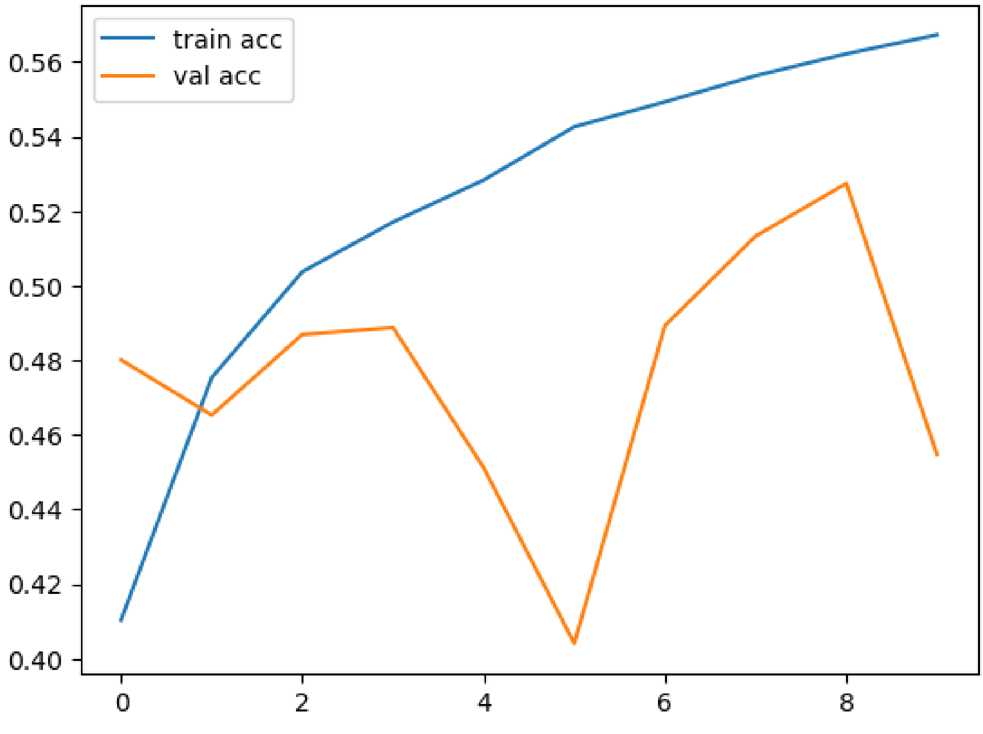


Figure 5. 5: ResNet-50 Model Accuracy.

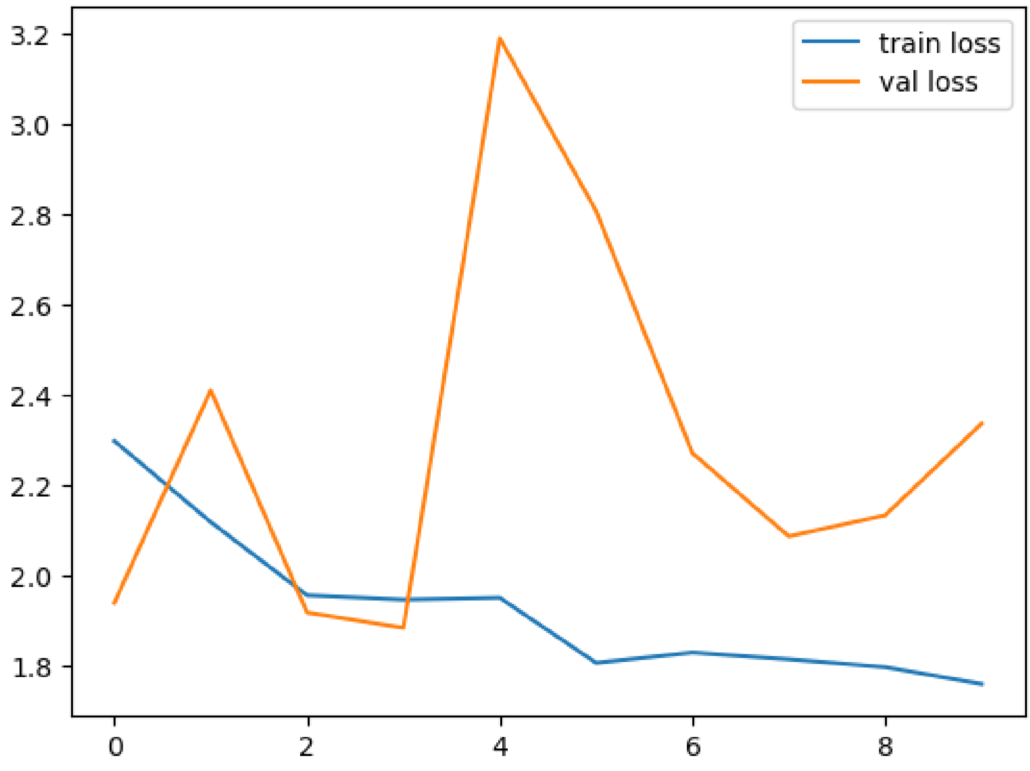


Figure 5. 6: ResNet-50 Model Loss.

### 5.4.2 GoogLeNet

GoogLeNet is different from rest of the CNN architecture, it employs a variety of techniques, including global average pooling and inception model with 1x1 convolution, to build deeper architecture.

Overall GoogLeNet architecture is of 22 layers with:

- Average pooling layer of stride 3 and filter size 5x5
- 128 filters and an 1x1 convolution for dimension reduction and ReLU activation.
- A fully connected layer with ReLU activation and 1025 outputs.
- Regularization of dropouts with a dropout ratio of 0.7.
- A 1000-class softmax classifier that produces results similar to the primary SoftMax classifier.

GoogLeNet Parameters:

Total params: 22,212,392

Trainable params: 409,608

Non-trainable params: 21,802,784

In this experiment we have 29395 train data and 9799 test data corresponding to 8 class with image size of (224,224). After training GoogLeNet model on dataset, we have achieved below accuracy and loss in 5 epochs.

| | |
|---|---|
| Training Accuracy | 82.40% |
| Train Loss | 3.24 |
| Validation Accuracy | 74.79% |
| Validation Loss | 6.26 |
| Testing Accuracy | 81.15% |
| Testing Loss | 3.92 |

Table 5. 4: GoogLeNet Accuracy and Loss

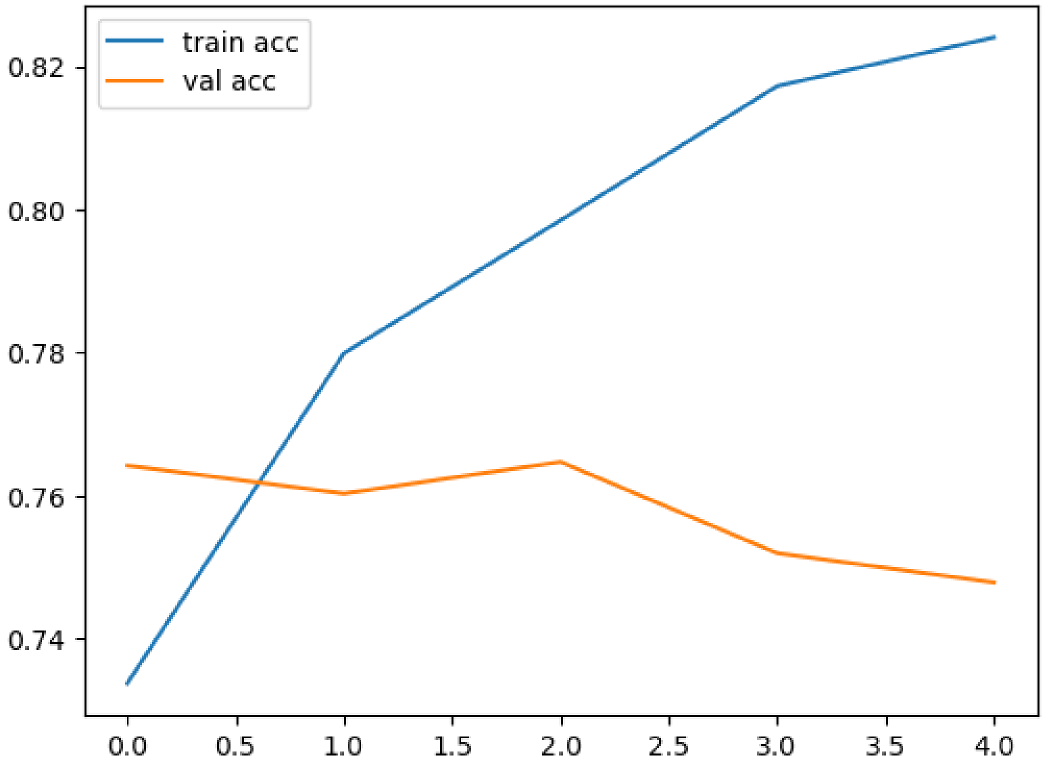


Figure 5. 7: GoogLeNet Model Accuracy.

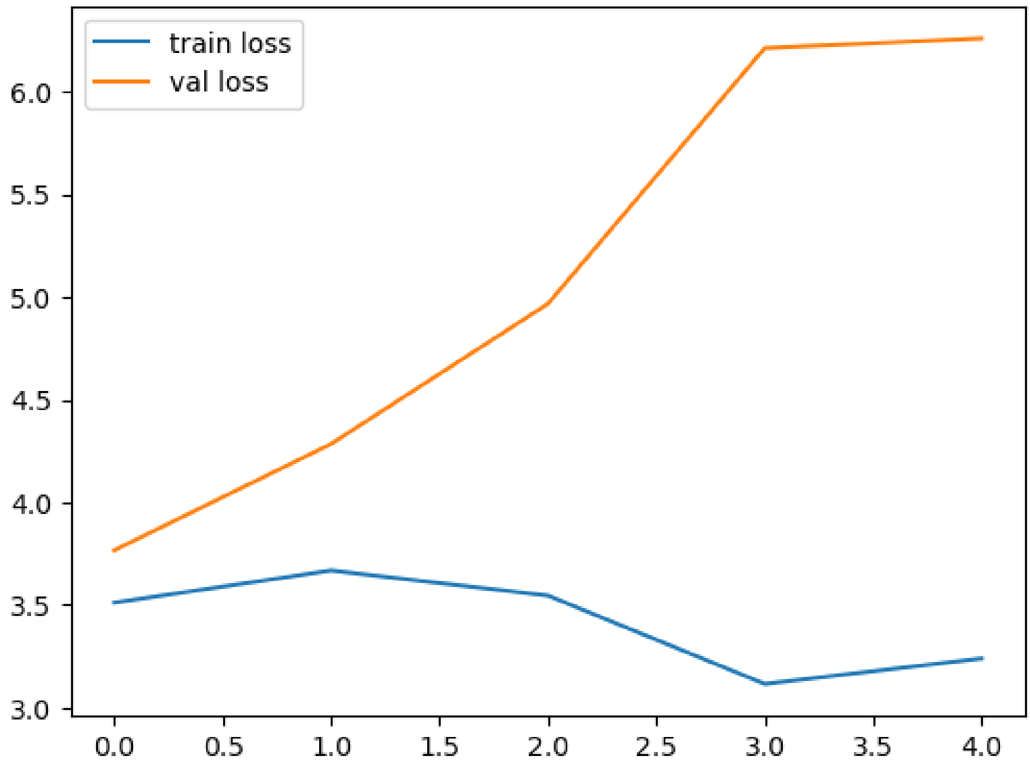


Figure 5. 8: GoogLeNet Model Loss.

### 5.4.3 VGG16

VGG16 is a 16-layer CNN model in which there are thirteen convolution layers, five Max Pooling layers, three dense layers. Though there are twenty-one layers but among which there are sixteen weight layers. In other word there are sixteen learnable or trainable parameters layers due to which it is known as VGG16.

Feature of VGG16:

- Throughout the whole architecture, the convolution and max pool layers are uniformly ordered.
- Input tensor size is 224 by 244 and has three RGB channels.
- Convolution layers have 3x3 filters with stride 1 with the same padding in all thirteen layers.
- Max pool layers have 2x2 filters with stride 2.
- Conv-1 layer have 64 filters, Conv-2 layer have 128 filters, Conv-3 layer have 256 filters, Conv- 4 and Conv-5 layer have 512 filters and final layer is the soft-max layer.

VGG16 parameters:

Total params: 14,915,400

Trainable params: 200,712

Non-trainable params: 14,714,688

In this experiment we have 29395 train data and 9799 test data corresponding to 8 class with image size of (224,224). After training VGG16 model on dataset, we have achieved below accuracy and loss in 10 epochs.

| | |
|---|---|
| Training Accuracy | 85.41% |
| Train Loss | 0.48 |
| Validation Accuracy | 74.80% |
| Validation Loss | 1.1641 |
| Testing Accuracy | 84.98% |
| Testing Loss | 0.60 |

Table 5. 5: VGG16 Accuracy and Loss

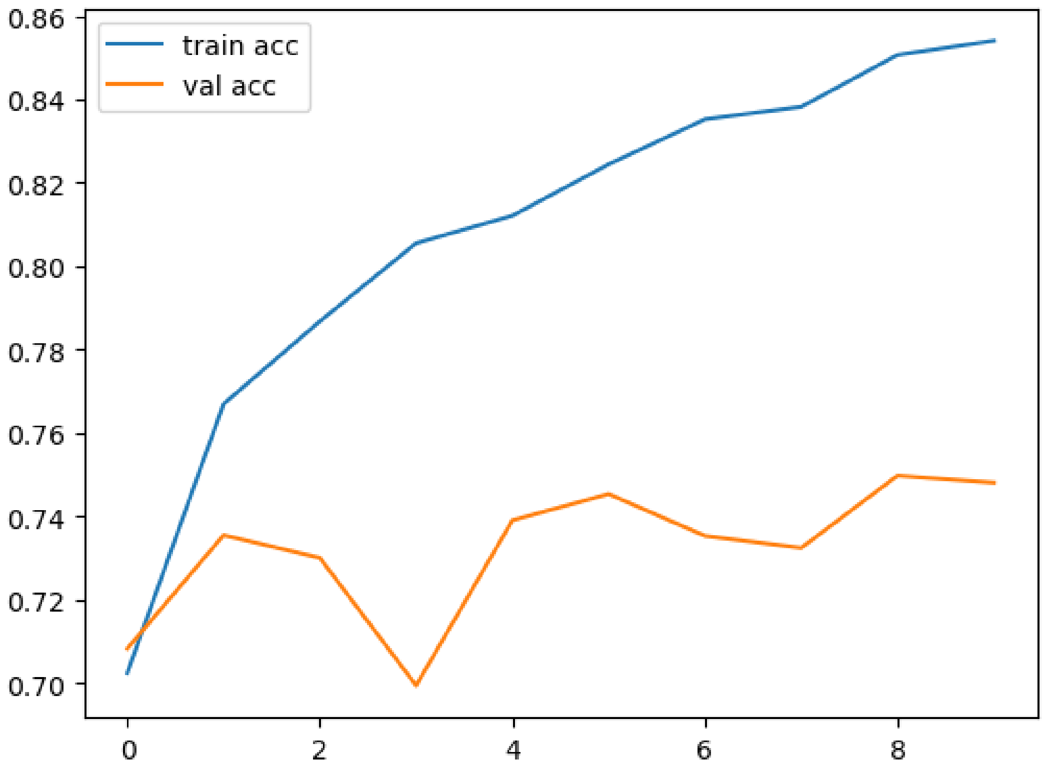


Figure 5. 9: VGG16 Model Accuracy.

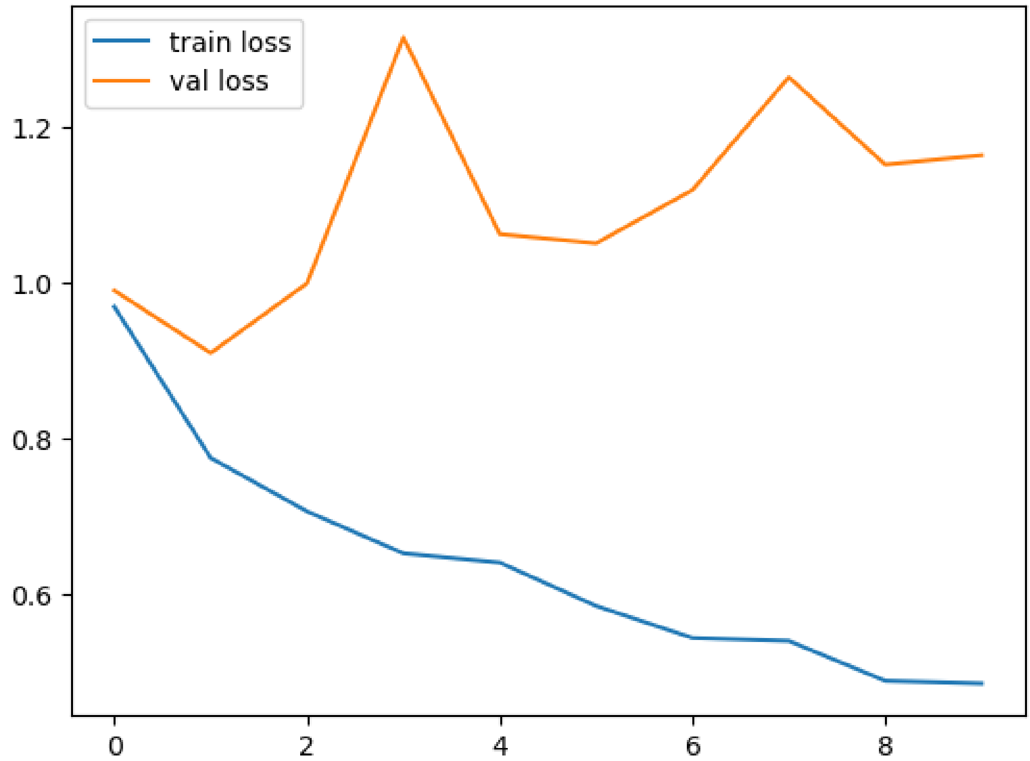


Figure 5. 10: VGG16 Model Loss.

### 5.4.4 Xception

Xception is for “extreme inception” with 71 layers deep CNN with depth wise separable convolution blocks and maxpooling layers. In Xception model applies filters on each depth and compress the input space using 1x1 convolution by applying it across the depth. Also, in Xception model there in both presence and absence of non-linearity after first operation.

The use of depthwise separable convolutions, which enable the network to learn spatial hierarchies more effectively than conventional convolutions, is the main novelty of Xception. This makes the network faster and more effective to employ by enabling it to attain performance similar to Inception with less parameters.

In Xception model data first got through entry flow and then middle flow and keep on repeating eight times before flowing through the exit with batch normalization in all convolution and separable convolution layers.

Xception parameters:

Total params: 21,664,304

Trainable params: 802,824

Non-trainable params: 20,861,480

In this experiment we have 29395 train data and 9799 test data corresponding to 8 class with image size of (224,224). After training Xception model on dataset, we have achieved below accuracy and loss in 10 epochs.

| | |
|---|---|
| Training Accuracy | 87.97% |
| Train Loss | 2.03 |
| Validation Accuracy | 78.87% |
| Validation Loss | 6.30 |
| Testing Accuracy | 87.24% |
| Testing Loss | 3.10 |

Table 5. 6: Xception Model Accuracy and Loss

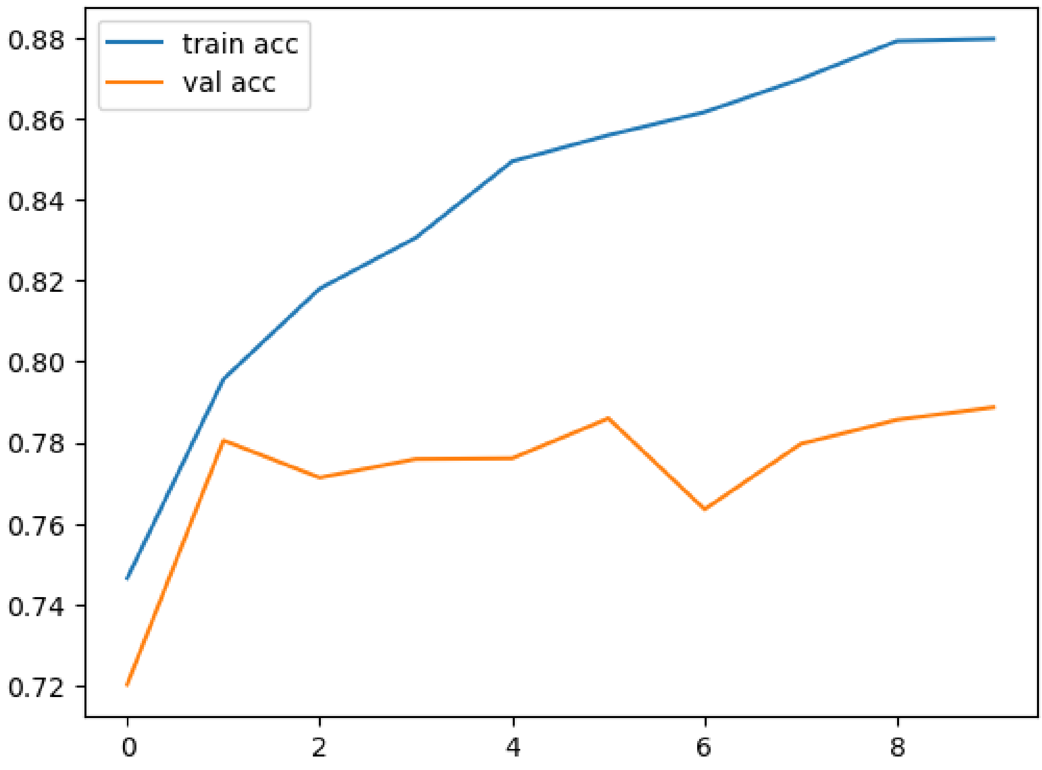


Figure 5. 11: Xception Model Accuracy.

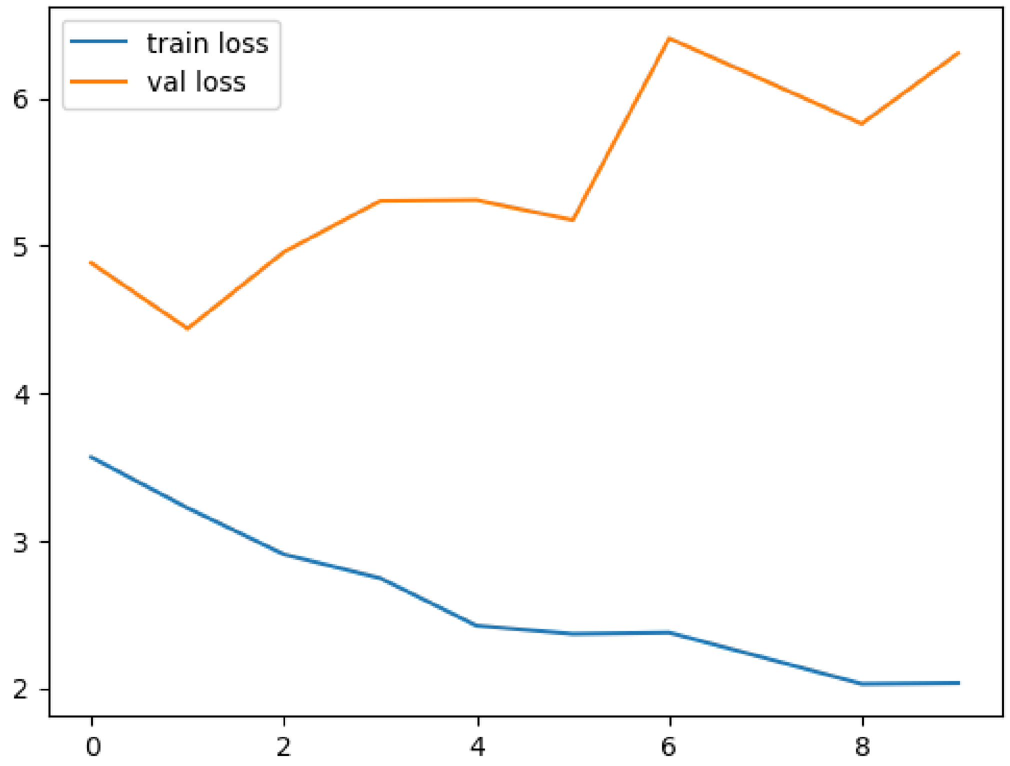


Figure 5. 12: Xception Model Loss.

### 5.5 Image Based Recommendation System

This section involves implementation of image-based recommendation system, aims to recommend N product based on image similarity.

Since we are detailing with images over here, first steps involved in building recommendation system is feature extraction. Where we will compare the feature of given image with the features of all the image in the data set. By model comparison we have seen VGG16 has given best result with the dataset. So, we will be using pre trained VGG16 model for feature extraction and for every image 2048 feature would be extracted. As per chapter 4 we have 39194 image datasets, So, using VGG16 we will extract features of 39194 image. Creating total extracted feature of (39194, 2048).

Once the feature extraction is done, we will export the feature set so the it can be reused may be for other image or telegram bot. To identify the closest image to the random image i.e., closest features to the features of the image provided by user we will be using NearestNeighbors.

To create model, we will be using VGG16 by excluding top 3 fully-connected layers and attached GlobalMaxPooling2D to VGG16 model.

```
model = VGG16(weights='imagenet',include_top=False,input_shape=(224,224,3))
model.trainable = False
model = Sequential([
    model,
    GlobalMaxPooling2D()
])
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 7, 7, 512)         14714688

 global_max_pooling2d (Globa  (None, 512)              0
 lMaxPooling2D)

=================================================================
Total params: 14,714,688
Trainable params: 0
Non-trainable params: 14,714,688
_________________________________________________________________
```

Function to extract the feature from image path and create a list of feature list of all the image of the dataset.

```
def extract_features(img_path,model):
    img = image.load_img(img_path,target_size=(224,224))
    img_array = image.img_to_array(img)
    expanded_img_array = np.expand_dims(img_array, axis=0)
    preprocessed_img = preprocess_input(expanded_img_array)
    result = model.predict(preprocessed_img).flatten()
    normalized_result = result / norm(result)
    return normalized_result


feature_list = []
for file in tqdm(filenames):
    feature_list.append(extract_features(file,model))
```

NearestNeighbors implements the algorithm which compares the closest distance of the training sample i.e., feature_list to the new point and predict the label from these.

```
neighbors = NearestNeighbors(n_neighbors=6,algorithm='brute',metric='euclidean')
neighbors.fit(feature_list)
```

Creating function for image recommendation which will identify the nearest neighbours and recommend the closes image.

```
def image_recommendation(item_id,num=5):

    image_path = 'Data\Images\Image_'+ str(item_id) + '.jpeg'

    print("------------------------------------------------------------------------")
    print("Original product:")
    print("------------------------------------------------------------------------")
    print("Product ID : " , data.loc[data['product_id'] == item_id]['asin'].tolist()[0].s
plit(' - ')[0])
    print("Title : ", data.loc[data['product_id'] == item_id]['title'].tolist()[0].split(
' - ')[0])
    print("Brand : ", data.loc[data['product_id'] == item_id]['brand'].tolist()[0].split(
' - ')[0])
    print("Cost : ", '$'+str(data.loc[data['product_id'] == item_id]['price'].tolist()[0]
))
    print("Rating : ", str(data.loc[data['product_id'] == item_id]['rating'].tolist()[0])
 + ' out of 5.0' )
    load_img(image_path)

    test_img_norm_img = extract_features(image_path,model)
    distances,indices = neighbors.kneighbors([test_img_norm_img])

    print("\n")
    print("------------------------------------------------------------------------")
    print("Most similar products:")
    print("------------------------------------------------------------------------")

    for file in indices[0][1:num+1]:
        rec_id = int(remove_prefix(os.path.splitext(filenames[file])[0],'Data\Images\Imag
e_'))
        print("Product ID : " , data.loc[data['product_id'] == rec_id]['asin'].tolist()[0
].split(' - ')[0])
        print("Title : ", data.loc[data['product_id'] == rec_id]['title'].tolist()[0].spl
it(' - ')[0])
        print("Brand : ", data.loc[data['product_id'] == rec_id]['brand'].tolist()[0].spl
it(' - ')[0])
        print("Cost : ", '$'+str(data.loc[data['product_id'] == rec_id]['price'].tolist()
[0]))
```

```
        print("Rating : ", str(data.loc[data['product_id'] == rec_id]['rating'].tolist()[
0]) + ' out of 5.0' )
        print("Url: ", "https://www.amazon.com/dp/"+ str(data.loc[data['product_id'] == r
ec_id]['asin'].tolist()[0].split(' - ')[0]))
        load_img('Data\Images\Image_'+ str(rec_id) + '.jpeg')
```

### 5.6 Evaluating the Image Recommendation Model

To evaluate the image recommendation model, we will pass random product id see the recommended product.

Output of the recommendation model:

```
-----------------------------------------------------------------------
Original product:
-----------------------------------------------------------------------
Product ID :  B00ANCGVRG
Title :  Casio G-Shock Black Dial Men's Quartz Watch
Brand :  Casio
Cost :  $80.0
Rating :  5 out of 5.0
Url : https://www.amazon.com/dp/B00ANCGVRG
```


```
-----------------------------------------------------------------------
Most similar products:
-----------------------------------------------------------------------
Product ID :  B00BF2CUYG
Title :  Casio G-Shock Garish Color Super Illuminator
Brand :  Casio
Cost :  $61.61
Rating :  5 out of 5.0
Url:  https://www.amazon.com/dp/B00BF2CUYG
```

Product ID :  B00791YX10
Title :  Casio Men's G6900KG-3CR G-Shock Military Green Multi-Function Digital Watch
Brand :  Casio
Cost :  $93.0
Rating :  4 out of 5.0
Url:  https://www.amazon.com/dp/B00791YX10

Product ID :  B00BU6RUU6
Title :  Casio Men's GD350-1B G Shock Black Watch
Brand :  Casio
Cost :  $112.0
Rating :  5 out of 5.0
Url:  https://www.amazon.com/dp/B00BU6RUU6

Product ID :  B00EBBC3XG
Title :  Casio G Shock Digital Black Resin Mens Watch GR8900A-1CR
Brand :  Casio
Cost :  $115.0
Rating :  4 out of 5.0
Url:  https://www.amazon.com/dp/B00EBBC3XG


```
Product ID :  B00IF3K7MK
Title :  Casio GD-X6900-1ER Mens G-Shock World Time All Black Digital Watch
Brand :  Casio
Cost :  $140.0
Rating :  5 out of 5.0
Url:  https://www.amazon.com/dp/B00IF3K7MK
```


Figure 5. 13: Output of Image based Recommendation System

### 5.7 Recommendation system with telegram bot.

In chapter 3, we discussed the steps to register and create a bot. Once bot is created, we will get a token that can used to access the Telegram API.

User can access bot get recommendation using three simple steps.

1. To open product recommendation bot use can access below link or search for @ml_product_recommendation_bot on telegram. After the user will land on Product Recommendation bot as shown in step 1.

Link - http://t.me/ml_product_recommendation_bot

2. User can start the application with command /start, and bot will respond with the request to upload image as shown in step 2 below.
3. User need to upload image to find the similar product as shown in step 3


Figure 5. 14: Steps to access product recommendation bot.

Once the image is uploaded by user, we first remove the background of the image using ‘rembg’ library and then extract the feature of the image without background and find the nearest neighbours of this image feature to the features extracted from the dataset and recommend top 5 product to the user with details like product id, title, brand, price, rating and url as shown in below figure 5.15.

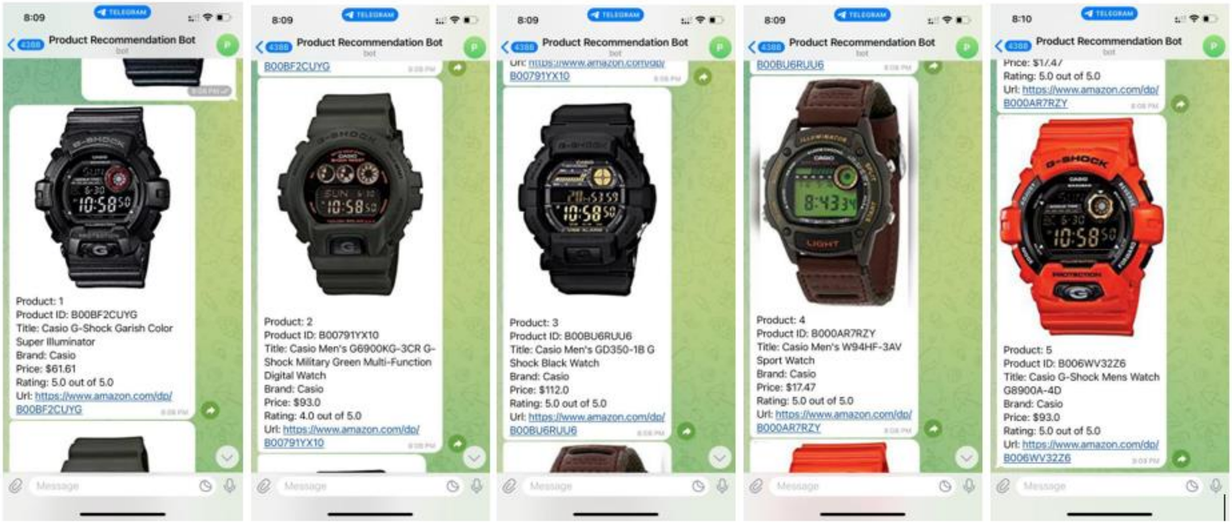


Figure 5. 15: Top 5 recommendation provided by recommendation bot.

### 5.8 Summary

In this chapter firstly we have implemented image classification model using CNN which involve extraction of features from the product image to observe the trend / pattern in the dataset based on this trend/ pattern the image classifier distinguishes whether the given image or random image or the product belong to which category i.e., Electronics, Pet Supplies, Home and Kitchen, Cell Phones and Accessories, Clothing Men, Clothing Women, Watches.

We also tried to implement recommendation system based on text i.e., content-based recommendation system. In this recommendation system, recommendation made is based on product tile, in other words system of the given product item co-sign similarity from all the product in the dataset is calculated and the product with the high co-sign similarity is recommended by the system.

While implementing image classification model using CNN, we have seen accuracy achieved on test data was 52.90% with loss of 9.83, which might result in inaccurate result. Due to which we have done experiment by passing extracted dataset on different CNN models like:

- ResNet-50
- GoogLeNet
- VGG16

- Xception

For all the model we did experiment where we passed 29395 train data and 9799 test data corresponding to 8 class. After training all model on dataset, we have achieved below accuracy and loss across all models.

| | Accuracy % | | | Loss | | | Epochs |
|---|---|---|---|---|---|---|---|
| | Training | Validation | Testing | Training | Validation | Testing | |
| CNN-SVM | 55.86 | 52.9 | 52.9 | 9.71 | 9.83 | 9.83 | 50 |
| ResNet-50 | 56.72 | 45.47 | 53.86 | 1.76 | 2.34 | 2 | 10 |
| GoogLeNet | 82.4 | 74.79 | 81.15 | 3.24 | 6.26 | 3.92 | 5 |
| VGG16 | 85.41 | 74.8 | 84.98 | 0.48 | 1.16 | 0.6 | 10 |
| Xception | 87.97 | 78.87 | 87.24 | 2.03 | 6.3 | 3.1 | 10 |

Table 5. 7: Comparing Accuracy and Loss across all Models.

We can see from above table accuracy of CNN-SVM is comparatively low and losses is high comparatively to another models. The accuracy achieved by Xception model is high but we can see there is significant loss.

So, comparing all the models VGG16 has provided better result as compared to other model where accuracy is high and loss is less. So, we have picked VGG16 as model for image-based recommendation system.

In the later section of this chapter implemented image-based recommendation system using VGG16 model where first we extracted features of all the image present in the dataset. Also, for random image provide by user we extracted the features and find the nearest neighbours of this image feature to the features extracted from the dataset to recommend the top 5 or top n similar product. Thus, using this mechanism, we can design the telegram bot to recommend 5 products based on similarity.

# CHAPTER 6: CONCLUSIONS AND RECOMMENDATION

## 6.1 Introduction

In this chapter we will conclude our research in area of Ai bot image-based recommendation system and work related to same. We will be discussing the lessons we learned work carried out toward achievement of the objectives we achieved in the research.

We will be also discussing the future recommendation in this area and challenges we face while building recommendation system.

## 6.2 Conclusion

To conclude, in this research we have covered image classification model using CNN, text recommendation based on co-sign similarity and implementation of image-based recommendation system based on best identified model.

In this research we have built a smart recommendation system based on image. Which can be integrated with instant messaging system using bot, in our research we have integrated the image recommendation model with telegram bot. Where user upload image in the telegram which internally called recommendation system over internet and recommendation system analyse the uploaded image looks for similar images in the dataset the return the recommended /similar product to the user with details like product id, title of the product, brand, price, rating and url from where user can navigate and buy the product.

For this research we have collected data from Amazon product data which comprises of 233.1 million. Since working with such large dataset would be difficult, we have extracted 5000 records across eight different categories like Electronics, Pet Supplies, Home and Kitchen, Cell Phones and Accessories, Clothing Men, Clothing Women, Watches, Shoes. Thus, combining up to 40000 records and reducing to 39194 record datasets after data cleaning.

We than build CNN image classifier on this dataset and also saw some images where not predicted accurately in classification due to high losses. Due to this we did experiment in different standard module like ResNet-50, GoogLeNet, VGG16 and Xception and found VGG16 as best module with high accuracy and less loss.

| | Accuracy % | | | Loss | | | Epochs |
|---|---|---|---|---|---|---|---|
| | Training | Validation | Testing | Training | Validation | Testing | |
| VGG16 | 85.41 | 74.8 | 84.98 | 0.48 | 1.16 | 0.6 | 10 |

Table 6. 1: Accuracy Loss of VGG16 Model

And then with VGG16 module we build image recommendation system where first we extracted features of all the image present in the dataset. Compared with the random image provide by user and find the nearest neighbours of this image feature to the features extracted from the dataset to recommend the top 5 or top n similar product and integrate this recommendation system with the telegram bot to build AI bot for image-based recommendation system.

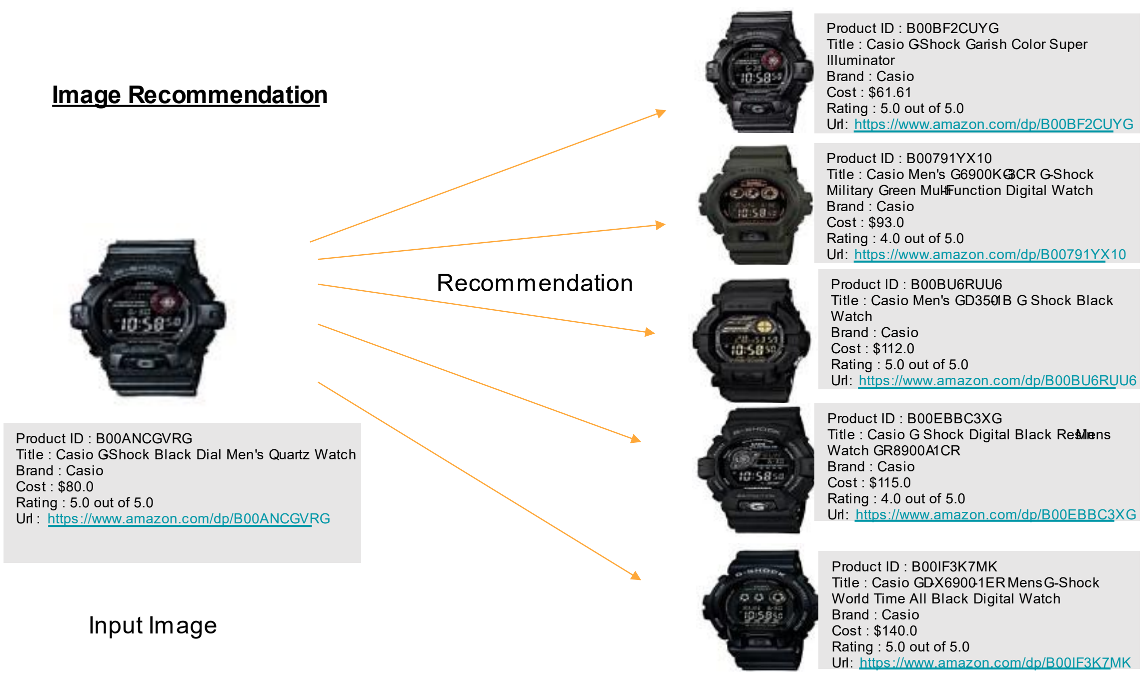


Figure 6. 1: Output of image recommendation system

## 6.3 Future Recommendations

In this research we did not combine CNN classification model and text-based recommendation to image-based recommendation to keep code open to accept any data. But while working with this dataset. We can combine all the three modules where first the input image can be categorised and then the image extracts the feature of the image and fine the nearest neighbours of this image feature to the features of the images of same categories and then look for text similarity with title and band for better recommendation.

Also, for future research we can:

- Explore GoogLeNet model by training the dataset on more epochs.
- If building a product replace the dataset by ecommerce Api.
- Data scraping can be done from multiple ecommerce website to create large dataset the try to recommend product.
- Incorporate user feedback: Another way to improve image-based recommendation systems is to incorporate user feedback into the recommendation process. This could involve allowing users to rate or provide feedback on the recommendations they receive, and using this feedback to improve the accuracy of the system over time.
- Increase scalability: As the dataset increases, increasing the scalability of system using distributed computing techniques, such as parallel processing, to enable the systems to handle larger datasets and serve more users.

## 6.4 Challenges

There are several challenges that can arise when building an image-based recommendation system. These challenges include:

1. Data collection and labelling: One of the biggest challenges in building an image-based recommendation system is collecting and labelling the data that the system will use to make recommendations. This can be a time-consuming and labour-intensive process, particularly if the system is being trained on a large dataset.

2. Feature extraction: In order for an image-based recommendation system to make accurate recommendations, it must be able to extract useful features from the images it is analysing. This can be difficult, particularly if the images are complex or have a lot of noise or clutter.

3. Scalability: Image-based recommendation systems often require a large amount of data and computational power to train and operate. This can make it difficult to scale the system to handle a large number of users or a large number of images.

4. Privacy concerns: Image-based recommendation systems can raise privacy concerns, as they may be able to extract personal information from the images they are analysing. This can be particularly concerning if the system is being used for sensitive applications, such as medical imaging or surveillance.

5. Adversarial attacks: Image-based recommendation systems can be vulnerable to adversarial attacks, in which attackers manipulate the images to trick the system into making incorrect recommendations. This can be difficult to defend against, and can undermine the reliability of the system.

## 6.5 Summary

This chapter summaries the whole thesis and the research around the same. This is thesis we have built an image-based recommendation system which can take images from the user using bot and image recommendation system process the image and find the best nearest feature present the dataset which is closest to the feature of the image provided by the user and finally recommend N products to the user where user can see name, cost, rating, brand, url of the product and user and navigate the product page to buy the product. In process to build the image-based recommendation system we have done model comparison to find the best possible model for recommendation system.

In this section we have also discussed future recommendation while working with image base recommendation system or working with similar dataset.

Also, in this chapter have discussed the challenges faced while working in recommendation system and the issue face while working with the dataset.

## REFERENCES


Chen, L., Yang, F. and Yang, H., (2015) Image-based Product Recommendation System with CNN. *Http://Cs231N.Stanford.Edu/Reports/2017/Pdfs/105.Pdf*, [online] p.2015. Available at: http://cs231n.stanford.edu/reports/2017/pdfs/105.pdf.

Chollet, F., (2017) Xception: Deep learning with depthwise separable convolutions. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-January, pp.1800–1807.

Dwivedi, R., Anand, A., Johri, P., Banerji, A. and Gaur, N., (2020) Product Based Recommendation System On Amazon Data. *Researchgate*, [online] June 2021. Available at: https://www.researchgate.net/publication/352815845_Product_Based_Recommendation_System_On_Amazon_Data.

Elsayed, S., Brinkmeyer, L. and Schmidt-Thieme, L., (2022) End-to-End Image-Based Fashion Recommendation. [online] pp.1–11. Available at: http://arxiv.org/abs/2205.02923.

Gonzalez, T.F., (2007) Handbook of approximation algorithms and metaheuristics. *Handbook of Approximation Algorithms and Metaheuristics*, pp.1–1432.

He, K., Zhang, X., Ren, S. and Sun, J., (2016) Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem, pp.770–778.

Huang, G., (2022) E-Commerce Intelligent Recommendation System Based on Deep Learning. *2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers, IPEC 2022*, pp.1154–1157.

Jouyandeh, F. and Zadeh, P.M., (2022) IPARS: An Image-based Personalized Advertisement Recommendation System on Social Networks. *Procedia Computer Science*, [online] 201C, pp.375–382. Available at: https://doi.org/10.1016/j.procs.2022.03.050.

Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P., (1998) A B7CEDGF HIB7PRQTSUDGQICWVYX HIB edCdSISIXvg5r ` CdQTw XvefCdS. *proc. OF THE IEEE*. [online] Available at: http://ieeexplore.ieee.org/document/726791/#full-text-section.

McAuley, J., Targett, C., Shi, Q. and Van Den Hengel, A., (2015) Image-based recommendations on styles and substitutes. *SIGIR 2015 - Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.43–52.

Rashed, A., Elsayed, S. and Schmidt-Thieme, L., (2022) *Context and Attribute-Aware Sequential Recommendation via Cross-Attention*. *RecSys 2022 - Proceedings of the 16th ACM Conference on Recommender Systems*, Association for Computing Machinery.

Simonyan, K. and Zisserman, A., (2015) Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp.1–14.

Stewart, I., (2012) Cups and downs. *College Mathematics Journal*, 431, pp.15–19.

Sulthana, A.R., Gupta, M., Subramanian, S. and Mirza, S., (2020) Improvising the performance of image-based recommendation system using convolution neural networks and deep learning. *Soft Computing*, [online] 2419, pp.14531–14544. Available at: https://doi.org/10.1007/s00500-020-04803-0.

Taneja, A. and Arora, A., (2018) Recommendation research trends: Review, approaches and open issues. *International Journal of Web Engineering and Technology*, 132, pp.123–186.

Thakur, A., (2021) Trends and Analysis of E-Commerce Market : A Global Perspective. 61, pp.11–22.

Ullah, F., Zhang, B. and Khan, R.U., (2020) Image-Based Service Recommendation System: A JPEG-Coefficient RFs Approach. *IEEE Access*, 8Dl, pp.3308–3318.

Wang, Y., Li, S. and Kot, A.C., (n.d.) JOINT LEARNING FOR IMAGE-BASED HANDBAG

RECOMMENDATION Yan Wang , Sheng Li , and Alex C . Kot Rapid-Rich Object Search ( ROSE ) Lab School of Electrical and Electronic Engineering Nanyang Technological University , Singapore 639798.

Yu, L., Han, F., Huang, S. and Luo, Y., (2018) A content-based goods image recommendation system. *Multimedia Tools and Applications*, 774, pp.4155–4169.

Zeng, G., He, Y., Yu, Z., Yang, X., Yang, R. and Zhang, L., (2016) Preparation of novel high copper ions removal membranes by embedding organosilane-functionalized multi-walled carbon nanotube. *Journal of Chemical Technology and Biotechnology*, 918, pp.2322–2330.

# APPENDIX A: RESEARCH PROPOSAL

AI BOT FOR IMAGE-BASED PRODUCT RECOMMENDATION

SHASHANK PRASAD

Research Proposal

AUGUST 2022

**Abstract**

The online shopping industry is booming day by day, and they largely depend on text filtering to find the required product. And for customers, it is also very inconvenient to download multiple applications and search for desired products manually. This is a research implementation where we will address and propose a new way to filter products via image filtering and recommend related products to customers through an instant messaging application like Telegram and WhatsApp.

In this research, to interact with the instant messaging application we will create a bot and this bot will call a recommendation system to get the recommended product for the given product image provided by the customer/user.

In this recommendation system, first, we will classify and categorize the product by the image classification model. This image classification model will be just the CNN model which will take the input image and categorize the product it belongs to. In the CNN model, we will also explore different CNN architecture/models like SVM (Support Vector Machine), VggNet, GoogleNet, and ResNet and validate their training, validation, and test accuracy.

In the second part of the recommendation engine, we help to identify the recommended product based on the input image, for which we will calculate cosine similarities between the feature extracted from the pre-trained model and image. Then, images are suggested based on the calculated cosine similarity, listed in descending order. So, higher the cosine similarity, the recommended image will be more similar to the input image, and to get a better recommendation of the product we will again calculate cosine similarity between the text like title and brand to get the exact/similar product what customer/user is looking for.

## Table of Contents


| | | |
|---|---|---|
| Abstract | | 2 |
| 1. | Background | 4 |
| 2. | Problem Statement OR Related Research OR Related Work | 6 |
| 3. | Aim and Objectives | 7 |
| 4. | Significance of the Study | 8 |
| 5. | Scope of the Study | 9 |
| 6. | Research Methodology | 10 |
| 7. | Required Resources | 15 |
| 8. | Research Plan | 16 |
| References | | 17 |

## 1. Background

In recent few years, we have seen growth in online shopping and most of businesses are trying to establish their presence online by creating their applications and selling most the products at their online store to conduct ecommerce marketing.

According to research, in 2022, global retail online sales will surpass US $5 trillion and by 2025 it is expected to US $7 trillion, despite slow growth.

.

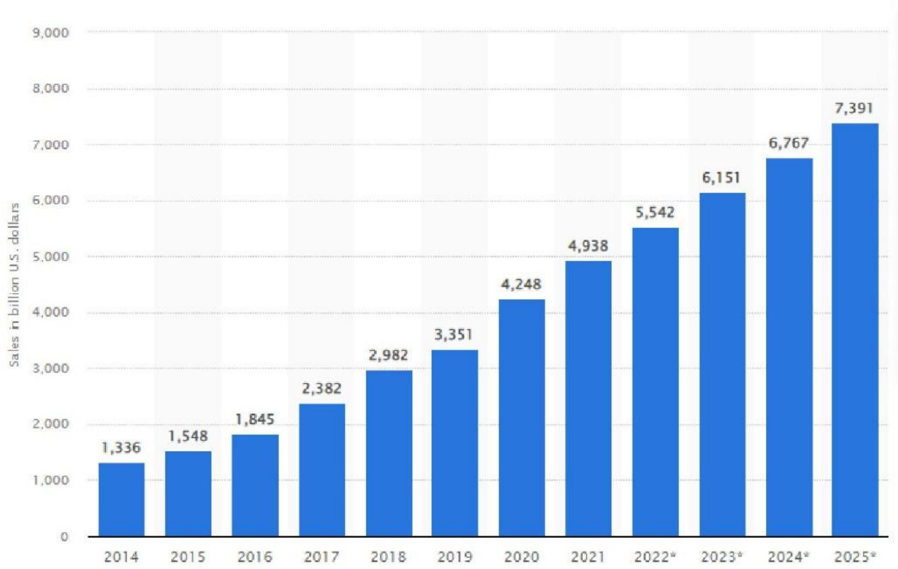


Figure 1: Global Sales in Ecommerce Industry

As online shopping is becoming more and more popular due to its convenience. However, to find the right product, the user needs to download multiple applications and search the resided product, filer it and select the right product. This is time-consuming and with increase in mobile applications it becomes a very tedious task for the user.

The other industry which has seen growth in mobile application area is instant messaging application like WhatsApp and Telegram where alone WhatsApp has 2 billion active users and is the most popular messaging service in over 100 countries.

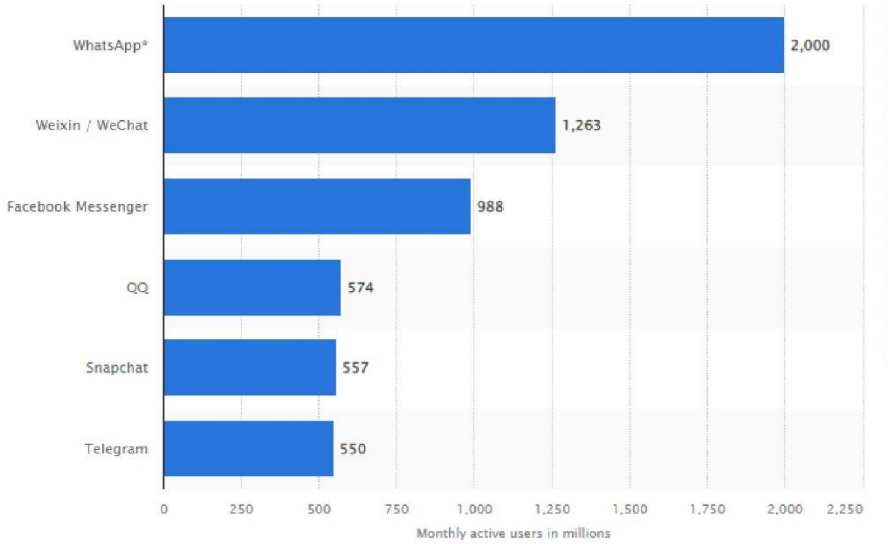


Figure 2: Users in Instant Messaging application.

One solution to solve this problem is to provide user more intuitive way to find relevant products by instant messing application like WhatsApp or Telegram where user can get the list of recommended products based on given image. In other words, An AI bot will help user to identify and recommend online product based on user image for which use can use instant messaging application.

Based on this idea, we will create a bot and run it inside an instant messaging application. User can interact with the bots through the instant messaging application and the bot will return the recommended based on user input. So, the bot here basically would be an image-based recommendation system that will take an image as input. Then this image input would be passed through the CNN model (Convolutional Neural Network) which will help to categorically classify the image to the object it might probably belong. Then the identified product or item is passed as input to the recommendation system. The recommendation system will use the product id or any id and will recommend the suitable product using content-based filtering i.e., products having the same content as the given input image to the recommendation system will be recommended and the user will be able to see the recommended product/ products in the instant messaging application.

## 2. Problem Statement OR Related Research OR Related Work

Problem Statement

Most of the ecommerce website / application like Amazon, Flipkart largely depends on text filtering to find the required product. And for customer it is inconvenient to download multiple application and search for desired product manually. This is very time consuming and can result in selection of wrong product with inferior quality and high price.

In this paper, we are proposing a new way to filter product via image and recommend related products to customers through instant messaging applications like telegram and WhatsApp. Image base recommendations will help user to filter exact or similar products.

Related Research or Related Work

(Wang et al., n.d.) The paper is specific to handbags and proposes methods through which handbags can be recommended to each customer. This paper uses join learning attribute projection along with one class SVM classification for the images clicked by user/customer. Through feature extraction feature of bag images clicked by customer is extracted and mapped to projection matrix. Then projection matrix is used along with SVM classifier to get the new recommendation.

(Chen et al., 2015) Convolutional neural networks and image-based product recommendations are topics covered in the paper. The paper uses CNN to determine the category that this thing most likely falls under before passing via an SVM, AlexNet, and VGG layer to determine the suggested purchase.

(Ullah et al., 2020) Paper proposes recommendation system based on content-based image retrieval where first class/type of the product is identified using Random Forests (RF) classifier and for feature extraction JPEG coefficients is used. In second phase, the proposed recommendation system retrieves closely matched similar products.

(Sulthana et al., 2020) Paper focuses on Improvising the performance of image-based recommendation system using convolution neural networks and deep learning. Improvising is done with the help of a deep architecture and a series of ‘‘convolution’’ operations that cause the overlapping of edges and blobs in images.

(Stewart, 2012) The paper discusses Modelling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering in which novel models are built for the One-Class Collaborative Filtering setting, where our goal is to estimate users’ fashion-aware personalized ranking functions based on their past feedback.

(Dwivedi et al., 2020) Paper discusses different types of recommendation system though which products can be recommended by recommendation system precisely on amazon data. Also, in this research popularity-based recommendation engine and recommendation using collaborative filtering is discussed and evaluated best model based on rmse, mean square error, mean absolute error.

(McAuley et al., 2015) This paper discusses recommendation model for style and substitutes. With this research model is discussed and developed which recommend compatible product which goes with each other or not like which clothes and accessories goes together and which do not go together which helps in styling. In this research using F-dimensional feature vector is calculated using CNN model which is used to group the product which can go along with each other.

## 3. Aim and Objectives

In this paper, we aim to address the recommendation system problem by analysing the feature of the image itself rather than the rating provided by the seller. Basically, we aim to create a recommendation system where customers tend to prefer products with visual similar attributes, such as shape, size, colour, style, or pattern.

To connect to this recommendation system, customer can use instant messaging platforms like WhatsApp or Telegram where the user/customer can upload image of prefer products and through our recommendation system they will get the desired link/URL of the product from various e-commerce websites like Amazon.

To develop an image-based product recommendation system used primarily in the e-commerce domain. This engine will take image of certain product as input via instant messaging application and get the recommendation of similar products that resemble closely to the input image.

We aim to achieve this goal by implementing a combination of dimensionality reduction, clustering, modelling, and recommendation system techniques. And integrating whole recommendation engine on real time with instant messaging application.

Objective of this research is:

- Based on customer input image of product, recommend good product to customer based on visual/ feature similarity.
- Achieve high accuracy in recommendation using image-based recommendation system for e-commerce like Amazon.
- Run recommendation system as a bot on instant messaging applications.

## 4. Significance of the Study

In modern world, the recommendation system plays a key role in defining business. It always has been in demand in every domain and in every industry consider it be in food industry, dating business, online gaming, online streaming media, metaverse or the e-commerce industry.

Personalized product recommendation helps several eCommerce businesses suggest relevant products to their end-users at multiple touchpoints. Quick and instant suggestions will make every user feel valued and give them a personalized shopping experience. Recommendations help the eCommerce industry offer a personalized shopping experience to the end-user and thus witness a rapid boost in user engagement and revenue flow.

Recommendation systems are of different types:

1. Collaborative filtering –

   Collaborative filtering focusses more on gathering, filtering and analysing data based on user's behaviour. For example, if user x likes pizza, pasta, Italian bread, and user y likes pasta, pasta, mac and cheese. So, it would be likely that user x will like mac and cheese and user y will like Italian bread.

2. Content based filtering -

   Content based filtering focusses more on user profile or user's regular/ preferred choices and product profile like description / title of the product. With this type of recommendation system, products are detailed with titles or described using keywords and customer profile will show the product of customer choices/ likes.

3. Hybrid filtering:

   In hybrid filtering products are recommended using both content-based filtering technique and collaborative filtering technique and user /customers are suggested with wide range of product.

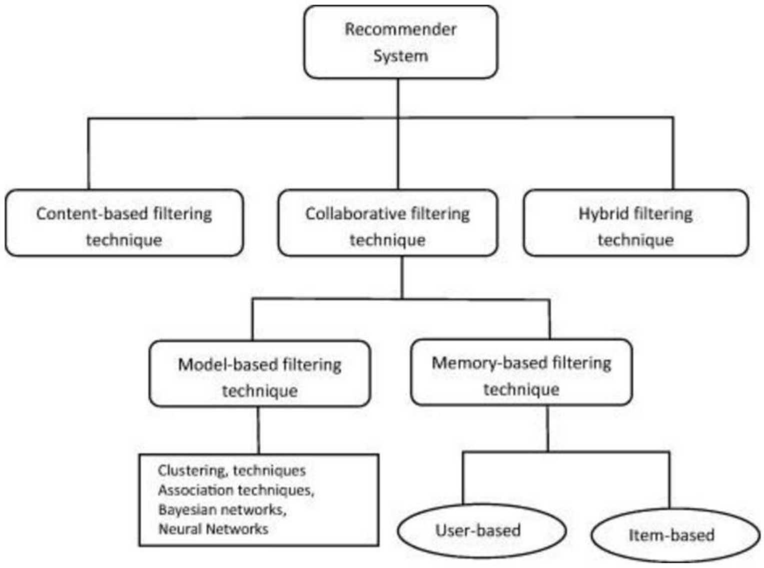


Figure 3: Types of Recommendation System.

With this study we will implement a content-based filtering technique based on user/customer input, we will gather and analyse product visual similar attributes, such as shape, size, colour, style, or pattern and recommend product to the user/customer.

Overall, this study will help to implement end to end recommendation system where user provide product image of their choice into the instant messaging application and get the recommended product based on content-based filtering.

## 5. Scope of the Study

In Score:

- Recommendation to the user will be provided based on user input image.
- Recommendation system can work with and without bot i.e., WhatsApp bot or Telegram bot.
- Recommendation system can work with either WhatsApp bot or Telegram bot.
- For data collection, data can be scrapped from Amazon website using tool example parse hub.
- If data is large then recommendation can be done from specific categories like T-shirt, Shoes, Bags, Hats, Watches.
- For WhatsApp bot new Twilio account needs to be created.

- Will try to explore cloud service as part of this research.
- We can explore the possibility of text-based recommendation engine along with image-based recommendation to get the better result.

Out of Scope:

- User should have smartphone with active phone number and instant messaging applications like WhatsApp or Telegram installed.
- Connection to the bot via instant messaging application should be done manually and will be out of scope of recommendation system.
- WhatsApp bot is intended for testing purpose and not for production usage.
- Product with English description will be considered for the research and other languages will be out of scope of the research.

## 6. Research Methodology

This research involves complete end to end implementation, where in first phase we need to create bot and second phase we will create recommendation system which will take input from the bot and respond to bot with the result.

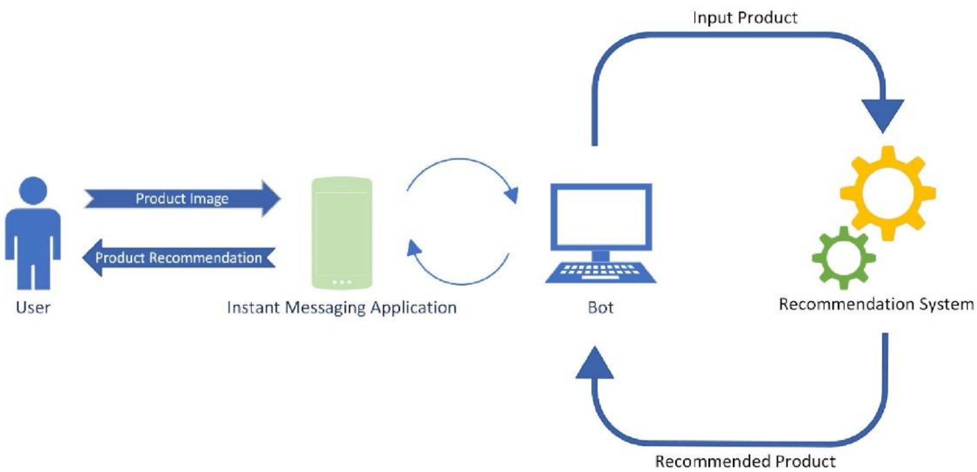


Figure 4: Overview for implementing image-based recommendation system

From the above architecture we can see, user through instant messaging application like WhatsApp and Telegram will connect to bot and provide image as input to the bot which will connect to recommendation system and recommendation system will return the recommended product to the user.

Methodology to setup and create new Telegram bot:

- To create a new bot, we need to login into the Telegram application and search for "BotFather". (BotFather is the bot provided by telegram application through which we can create new bot account and manage existing bots)

- On "BotFather" chat create new bot by "/newbot" command.

- "BotFather" will ask for a bot name and username and will generate authentication token for newly create bot.

- This token value is unique for every bot, and we will use this token in our python code and connect to the recommendation system.

- Along with token, name provided will be displayed in contact details section through which user/customer can connect to bot-based recommendation system.

Methodology to setup and create new WhatsApp bot:

- To create new bot in WhatsApp we need to setup Twilio account using one of the api called "Twilio WhatsApp API".

- After login into Twilio account, we will configure sandbox environment for WhatsApp.

- On python code install Twilio, flask and request and create flask chatbot service and connect the bot with the recommendation system.

Recommendation System:

In Recommendation system, we will implement two major tasks:

1. Classify and categorise input image provided by customer.
2. Get the list of similar products and recommend the most similar product.

We will classify and categorise the input product using a convolution neural network as the initial step of the recommendation system. Convolution Neural Networks are a deep learning algorithm and are specialised particularly working in the neural network field of visual data like images and videos.

In Convolution Neural Networks, image classification is done by complex network with multiple layers and this layer will help to get the feature and identify the product, Example, consider a complex neutral network.

1. Horizontal and vertical edges are extracted in the first layer as raw features.
2. More abstract elements, such textures, are extracted in the second layer (utilizing the attributes that the first layer extracted)
3. Based on the textures, the subsequent layers may be able to distinguish specific aspects of the product image, such as belt, cloth, material, shape, size, etc.
4. Further layers may reveal type, colour, style, or pattern, etc.
5. And last layers finally, classify and categories the image as T-shirt, Shoes, Bags, Hats, Watches.

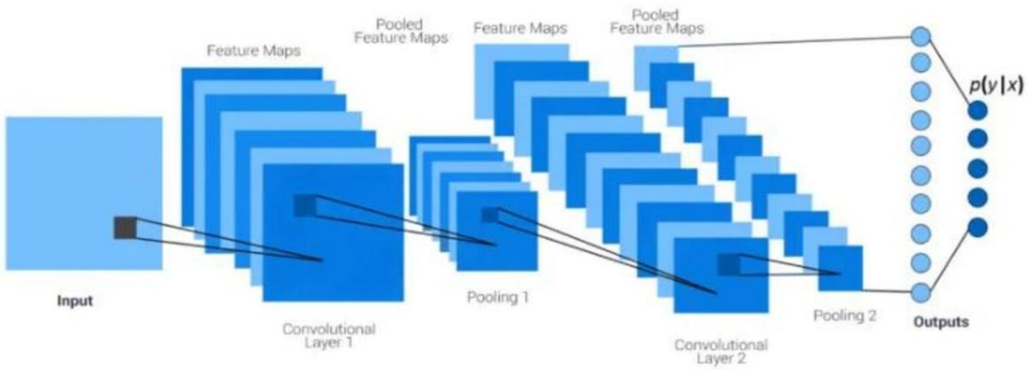


Figure 5: Example of CNN Architecture

The above figure is a simple architecture of CNN, where we have used multiple Convolutional Layers and multiple pooling layers (Pooling layers in typical CNN architecture signifies the layer which is statistical aggregate of previous layers, and these layers are used to reduce the dimensions of the feature).

In this paper, we will try to classify the product using the different CNN Architecture models to compare it and for better training, validation, and test accuracy. Different architecture we will try to explore are:

- SVM (Support Vector Machine)
- VggNet
- GoogleNet
- ResNet

After, the first step i.e., of classification using CNN module we will categorize the product provided by customer under its respective category. After this we will get the list of similar products and recommend the most similar product.

In the second part of recommendation system, our recommendation engine will recommend product based on image, for which we will calculate cosine similarities between the feature extracted from pre trained model and image. Then, images are suggested based on the calculated cosine similarity, listed in descending order. So, higher the cosine similarity, the recommended image will be more similar to the input image.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Figure 6: Cosine similarity formula


Figure 7: Figure illustrating similar product tends to high cosine value or group together.

For better result, we can calculate cosine similarity between the text i.e., text feature like title and brand. In other words, for effective recommendation we can include product recommendation based on text for which we will calculate cosine similarity score using weighted average formula which is individual cosine similarity score along with weight of each type.

$$\text{Similarity score} = \frac{\text{Weight}_p * \text{Simarity}_p + \text{Weight}_b * \text{Simarity}_b}{\text{Weight}_p + \text{Weight}_b}$$

Figure 8: Formula for the weighted average

Approach

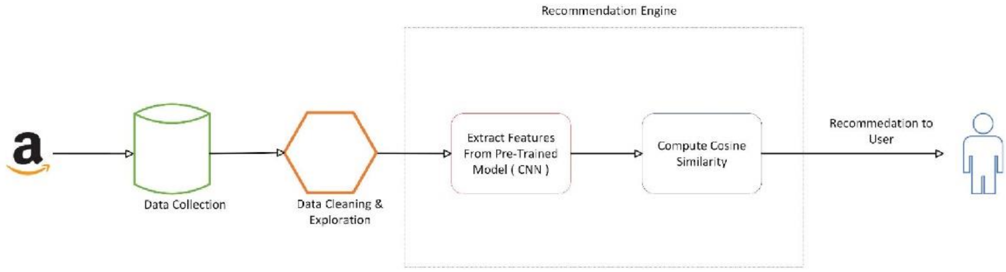


Figure 9: Recommendation engine flowchart

Above figure 9 describes the steps involved in building the recommendation engine.

- In the first step we will get the data from amazon and store it in CSV file.
- After data loading, we will clean the data and categories the data. In data cleaning we will also remove duplicates and perform exploratory data analysis on clean data.
- After cleaning the data and performing exploratory analysis we more to recommendation system where first we will work on image classification with aim to categories the random image using CNN model.
- Post categorisation we, will compute cosine similarity, and products with higher cosine value for the random product (or product provided by user) can be recommended.

## 7. Required Resources

As part of research implementation below software and hardware resources would be required:

- Software Resources
  - Python latest version installed on development environment.
  - Libraries required for development:
    - Pandas
    - Numpy
    - Seaborn
    - Sklearn
    - Tensorflow
    - Wordcloud
    - Scipy
    - Matplotlib
    - Pathlib
    - OS
    - Flask
    - Ngrok
    - Python-telegram-bot
    - Twilio
    - Requests

  - Twilio account to create a sandbox environment for WhatsApp bot.
  - For data extraction parse hub tool can be used on Amazon website

- Hardware Resources:
  - User should have a smartphone with an active phone number and instant messaging application like WhatsApp or Telegram installed.
  - Desktop/Laptop with anaconda installed on machine.
  - System with hardware accelerator GPU/TPU and high ram.

## 8. Research Plan

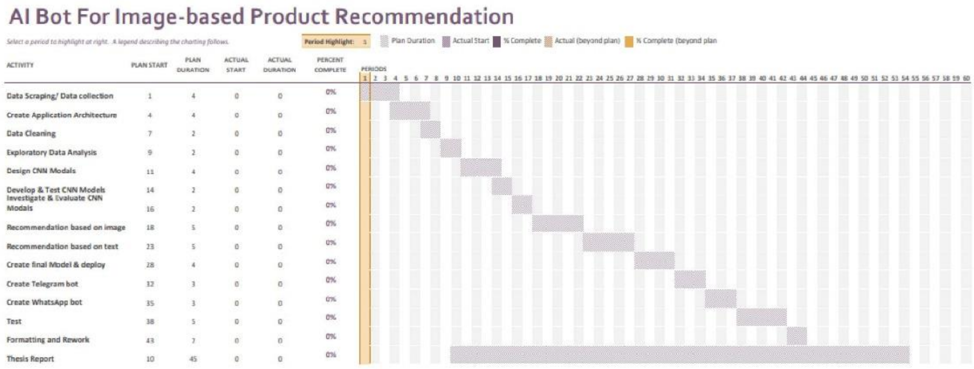


Figure 10: Research plan in Gantt chart

<u>Risk & Contingency Plan</u>:

- Lack of data or too much data:
  - While implementing there might be a possibility that the data required for whole project is not sufficient or size of data is very large.
  - To tackle this issue, we can pick and look for few categories of product like T-shirts, Hats, Watches, Shoes, and bag etc.

- Image provided by user is improper or is not a product or the model is unable to identify the product. In such case, we will show/send user with proper response message.

- While working with bots there is a risk involved around WhatsApp implementation because there is a time limit for which we can create sandbox environment and for production environment we need to buy credits. To mitigate this issue, we will consider Telegram bot for implementation.

**References**

Chen, L., Yang, F. and Yang, H., (2015) Image-based Product Recommendation System with CNN. *Http://Cs231N.Stanford.Edu/Reports/2017/Pdfs/105.Pdf*, [online] p.2015. Available at: http://cs231n.stanford.edu/reports/2017/pdfs/105.pdf.

Dwivedi, R., Anand, A., Johri, P., Banerji, A. and Gaur, N., (2020) Product Based Recommendation System On Amazon Data. *Researchgate*, [online] June 2021. Available at: https://www.researchgate.net/publication/352815845_Product_Based_Recommendation_System_On_Amazon_Data.

McAuley, J., Targett, C., Shi, Q. and Van Den Hengel, A., (2015) Image-based recommendations on styles and substitutes. *SIGIR 2015 - Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.43–52.

Stewart, I., (2012) Cups and downs. *College Mathematics Journal*, 431, pp.15–19.

Sulthana, A.R., Gupta, M., Subramanian, S. and Mirza, S., (2020) Improvising the performance of image-based recommendation system using convolution neural networks and deep learning. *Soft Computing*, [online] 2419, pp.14531–14544. Available at: https://doi.org/10.1007/s00500-020-04803-0.

Ullah, F., Zhang, B. and Khan, R.U., (2020) Image-Based Service Recommendation System: A JPEG-Coefficient RFs Approach. *IEEE Access*, 8Dl, pp.3308–3318.

Wang, Y., Li, S. and Kot, A.C., (n.d.) JOINT LEARNING FOR IMAGE-BASED HANDBAG RECOMMENDATION Yan Wang , Sheng Li , and Alex C . Kot Rapid-Rich Object Search ( ROSE ) Lab School of Electrical and Electronic Engineering Nanyang Technological University , Singapore 639798.