Product recommendation using sentiment analysis of product’s review in e-commerce

Dr. Bhrantav Vora, Assistant Professor

LDRP Institute of Research and Technology

MCA Department, KSV University, Gandhinagar.

Dr. Ronak B. Patel, Associate Professor

Shrimad Rajchandra Institute of Management and Computer Application , UkaTarsadiaUniversity,Maliba Campus, Bardoli-Mahuva Road, Gopal Vidyanagar, Barodli,Gujarat.

Dr. Meghna Patel, Associate Professor

A. M. Patel Institute of Computer Studies,

GanpatUniversity, Kherava, India.

Prof. Girish Dave, Assistant Professor

B P College of Computer Studies,

KSV University, Gandhinagar

Prof. Megha Shah, Assistant Professor

B P College of Computer Studies,

KSV University, Gandhinagar

ABSTRACT

Recommendation is very crucial part of human life; we rely on recommendation for almost every stuff in our daily routine. In this research work we worked on sentiment analysis by applying frequent pattern mining on the reviews of products provided by the customers on e-commerce platform. It is very easy to recommend a product based on its rating which is usually out of 5, but to deal with the reviews written in few thousands then it would be not feasible for any individual to go through and then select the best product out of available. It is very essential to identify the polarity of the given review by analyzing the sentiment of that particular review. We are applying BoW model and extract the frequent pattern by filtering positive keyword to check the polarity of the review and finally recommending the product to the customer.

Keywords—Machine Learning; Recommendation System; Artificial Intelligence; NLP; Sentiment Analysis

# INTRODUCTION

Sentiment Analysis also known as Opinion Mining plays a vital role in recommending products online through e-commerce platforms. The huge advancement in E-Commerce worldwide avails this opportunity to mine the frequent pattern and extract the insights from the product’s review through opinion mining or sentiment analysis [1]. Fig. 1 depicts the process of sentiment analysis on review given for a product. While selecting any product in an online e-commerce marketplace, customers always rely on ratings given to that product generally out of 5. This is an attempt to provide extra support to the customer in their decision making to finalize the product selection through recommendation [1].

Product Reviews

Sentiment Identification

Feature Selection

Sentiment Classification

Sentiment Polarity

Fig 1: SA Process

# Related Work

Recommendation System (RS) is a very crucial component for e-commerce giants, for an individual in his/her daily routine he/she heavily depends on recommendation whether it is selecting a school for their kids, buying new car/property, consulting a doctor/lawyer, or even purchasing items online. There are three types of recommendation system 1. Content Based Filtering, 2. Collaborative Filtering and third is the mixing for first two that is 3. Hybrid filtering [2].

## Content based filtering

Content based filtering technique uses features of the product to work with and recommend similar other products to the customer who is most likely to buy the product, It works on observing previous actions of the customer or by using explicit feedback from the customer.

## Collaborative filtering

Collaborative filtering technique uses demographic data of the customer and prepares a customer profile, then it would filter out the product for other customers whose profile matches and based on that recommendation can be done.

## Hybrid filtering

Hybrid filtering is an approach to combine both content based filtering and collaborative filtering techniques.

Recommendation is a three-phase process which includes 1. Information collection Phase, 2. Learning Phase and 3. Prediction or Recommendation Phase [3].

1. **Information collection phase**

In this phase of recommendation all the necessary information is being collected through various means which include web scraping, APIs, surveys, and avail huge datasets from portals like kaggle.

1. **Learning phase**

Learning phase is very important in the recommendation system, where ML algorithms take place in order to learn the frequent pattern and further help in making predictions in later stages.

1. **Prediction/Recommendation phase**

This is the final phase of the recommendation system. In this phase RS will try to predict or recommend the best suitable option from the available options based on the learning using ML algorithms in the previous phase. If needed feedback is also provided to the information/data collection phase to improve the prediction. Fig. 2 depicts the recommendation phases [3].

Information collection phase

Learning phase

Prediction / Recommendation phase

Feedback

Fig. 2 Recommendation Phases.

In this research study, a recommendation model was designed, developed and implemented which analyzes using patterns which are extracted from the reviews of the individual customer and rating given from the individual customer. The recommendation algorithm is used in the proposed system to recommend the product at different levels to the customer. In this research our proposed model accepts users’ review of more than one product and gives a level of recommendation i.e. most likely to purchase, likely to purchase, recommended, less recommended and not recommended at all.

The principle thought behind the Recommendation frameworks for E-Commerce is to fabricate a connection between the products, customers, and settle on the choice to choose the most suitable item to-a particular customer[1]. Gone ahead time, consistently in a hurry, current customers shop in erupts from numerous gadgets, as opposed to set aside the effort to take part in a careful shopping long-distance race[4]. Brands conveying fast, on-point offers to catch the most advantages. As indicated by the Personalization Consumer Survey did by 48% of customers go through additional with an E-Commerce organization conveying customized shopping knowledge[2]. ML frameworks enable you to catch information from past and current shopping sessions and change them into dynamic offers[13]. ML can pinpoint which product is in stock to feature[14]. Keen recommender frameworks can look over your whole item index and line up the best items for individual purchasers. You can make it a stride further and show the closest blocks[3]. E-Commerce defines the scope of products is accessible, to peruse the information, focusing on the prospect with nearby stock promotions later in the day[15]. A few out of every unique possibility is prepared to purchase from you at this very moment[16]. When a particular product is absent on an E-Commerce site, many customers will go looking somewhere else. Indeed, even graphic inquiries like 'a white shirt with brilliant catches' or 'nutrients for winter' may not lead customers to the items they need. On location, web search tools are somewhat inadequate when given random questions[17].

AI calculations can manage this issue. Aside from being prepared to perceive a more extensive scope of equivalent words, they can likewise help you naturally arrange your items dependent on their highlights[18]. Profound learning calculations are as of now equipped for dissecting item pictures and separating them into specific traits, for example, Slipover, A-line skirt, knee-length, and so on[19]. Recommender frameworks are utilized by E-business locales to propose items to their customers. The items can be prescribed depending on top general dealers on-site, in light of the socioeconomics of the customer, or dependent on an investigation of the past purchasing conduct of the customer as an expectation for future purchasing conduct[20]. Extensively, these procedures are a piece of personalization on a site since they help the site adjust to every customer. Recommendation Systems give the experience of personalization on the Web,which engages the customer for personalization experience[21]. To understand as personalization to this degree is one approach to understand Pine's thoughts on the Web. In this manner, Pine would presumably concur with Jeff Bezos, CEO of Amazon.com, when he said: "On the off chance that I have 2 million customers on the Web, I ought to have 2 million stores on the Web[22]. " Recommender frameworks upgrade E-business deals in three different ways: Browsers into purchasers: Visitors to a Web website regularly investigate the webpage while never obtaining anything[23]. Recommender frameworks can enable customers to discover items they wish to buy. Strategically pitch: Recommender frameworks improve strategically pitch by proposing extra items for the customer to buy. If the proposals are great, the normal request size should increment. For example, a site may suggest extra items in the checkout procedure, in light of those items as of now in the shopping basket. Dedication: In reality, as we know it where a site's rivals are just a tick or two away, picking up customer faithfulness is a basic business system[24]**.** Recommender frameworks improve reliability by making a worth included connection between the site and the customer. Destinations put resources into finding out about their customers, use recommender frameworks to operation that learning, and present custom interfaces that match customer needs. Customers reimburse these destinations by coming back to the ones that best match their needs. The more a customer utilizes the proposal framework – instructing it what they need – the more faithful they are to the site. "Regardless of whether a contender was to assemble precisely the same capacities, a customer would need to invest an unnecessary measure of time and vitality instructing the contender what the organization knows" [25].  A clever framework like recommender takes care of the data over-burden issue on the Web by offering decisions that consider customers' needs or interests[26]. The job of recommender frameworks isn't irrelevant in different significant level sites, for example, Amazon, YouTube, Netflix, Yahoo, and so on[27].

# Proposed framework

Proposed framework can be divided into four phases, phase 1 is identifying products in the e-commerce website, phase 2 is extracting user reviews and rating of identified products in phase 1, phase 3 is implementing machine learning algorithm on collected information from phase 2 and finally in phase 4 by applying improved steepest ascent hill climbing algorithm recommending the product. Fig. 3 demonstrates the proposed framework.

Products in E-Commerce Websites

**Phase 1**

User Rating and Reviews

**Phase 2**

Improved Frequent Pattern Mining

**Phase 3**

Artificial Intelligence

**Phase 4**

Fig. 3 Proposed framework

## **Phase1. Products in e-commerce**

This is the very first phase in product recommendation framework, in this phase we have identified six product (i.e. Smart Phone, Television, Refrigerator, Laptop, Washing Machine and Air Conditioner) in the electronic segment in e-commerce websites to collect data. We wanted to collect ample amount of data in terms of rating and reviews of customer to provide enough amount of data to our proposed model, further we have observed that products in electronic segment are having more number of reviews and rating compare to others and hence we have identified six different product category namely fridge, television, laptop, smart phones, air condition and washing machine.

## **Phase2. User Rating and Reviews**

In the second phase of the recommendation framework we have developed a python script to scrap product detail from e-commerce websites. In an attempt to extract required data of the product we scraped the following product detail using python script, Brand, Product, Model, Users’ Review, Users’ Rating, Price. We have manually selected features like users’ rating and reviews and took it forward in the next phase of recommendation framework.

## **Phase3. Improved Frequent Pattern Mining**

In this phase, we apply NLP on collected reviews in previous phase to process the natural language, in order to find appropriate information from the collected reviews we have applied Bag of Words model and try to find frequent pattern in reviews and converted categorical information into numerical to process it further in machine learning algorithm

## **Phase4. Artificial intelligence**

In this phase of recommendation framework, Artificial intelligence is used to recommend the product to the customer based on the received information from the previous phase. Steepest ascent hill climbing algorithm is used to find the best neighboring node to meet the goal state and that product is being recommended, further in improvement we have evolved steepest ascent hill climbing algorithm to the next level and provide level of recommendation i.e. Most recommended, Most likely recommended, Recommended, Less recommended and not recommended at all.

# Experimental Procedure

In our research work, we can divide our experimental procedure into three major phases to get a clear understanding of implemented research. Phase 1 is Data or Information collection phase in which we created a python script in Google colab and implemented web scraping to collect targeted data from the e-commerce websites. In Phase 2 we are performing pre-processing and learning tasks, and again we have created a separate python script to extract knowledge from the data that we have scraped from the previous python script created in phase 1. Lastly in phase 3 which is the recommendation or prediction phase we have created one more python script in which we have implemented the steepest ascent hill climbing algorithm of artificial intelligence and achieved the recommendation.

P

H

A

S

E

1

Data, Information Collection

P

H

A

S

E

2

Data Refining

P

H

A

S

E

3

Recommendation

Fig. 4 Phases in Experimental Procedure

## **Data Collection**

Data collection is one of the most basic and necessary tasks to be completed. In order to move forward in research work, a researcher needs to collect data, and data can be available from various sources. In our research work we have collected data from e-commerce websites.

### **Web Scraping**

Web scraping is the technique to extract large amounts of structured data from the web. In the era of information and knowledge on the web, it is highly competent to extract data from the web. In order to implement data mining, pattern recognition, forecasting through database drilling one must need ample amount of data to work on and it should also be available in soft form to easily implement decidedly accurate and result oriented machine learning and artificial intelligence algorithms and for the same web is the most suitable platform available. Extracting data from the web through different websites using greatly powerful scripting language i.e. python is web scraping. Fig. 5 illustrates the way we scrap the data from e-commerce websites.

Python script - Web Scraping

Configure selenium framework

Install chromium web driver

Include pandas library files

Write a code with Xpath to scrape required data

Exporting CSV file

Fig. 5 Data collection flow

### **Python Script**

In our research work to obtain customers’ review and rating of product from different e-commerce websites, we have developed a python script in Google colab which is a cloud based data science work space similar like Jupyter notebook, which also provide 13 GB of ram and processors ranging from CPU, GPU to TPU.

### **Selenium Framework**

In python script we have incorporated selenium framework which is an open-source web-based automation tool with python it empowers to connect with the browser and can send python commands to different browsers.

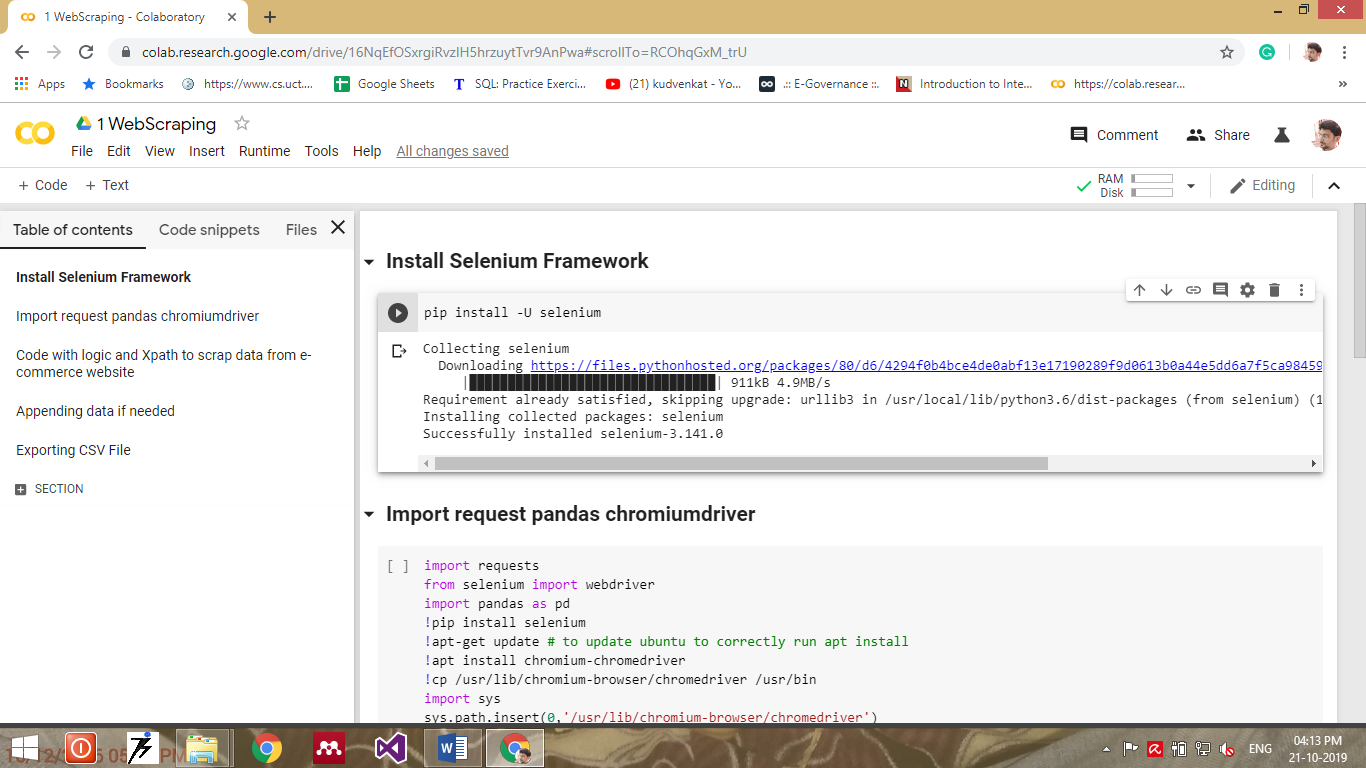


Fig. 6 Selenium configuration in Python notebook

### **Chromium Driver**

We have then configured the ChromeDriver, which is a separate executable that Selenium WebDriver uses to control Chrome. X - Path for XML path language is used for selecting nodes from an XML document of targeted webpage.

### **Pandas Library**

We have also used pandas library files in python, it is a NumFOCUS sponsored project and an open source BSD-license library, providing high-performance, easy-to-use data structures and data analysis tools for the [Python](https://www.python.org/) programming language.

By using above mentioned tools we have then generated a powerful python script to extract desired data from the web. Figure shows the code implementation of importing pandas library file and chromium driver.

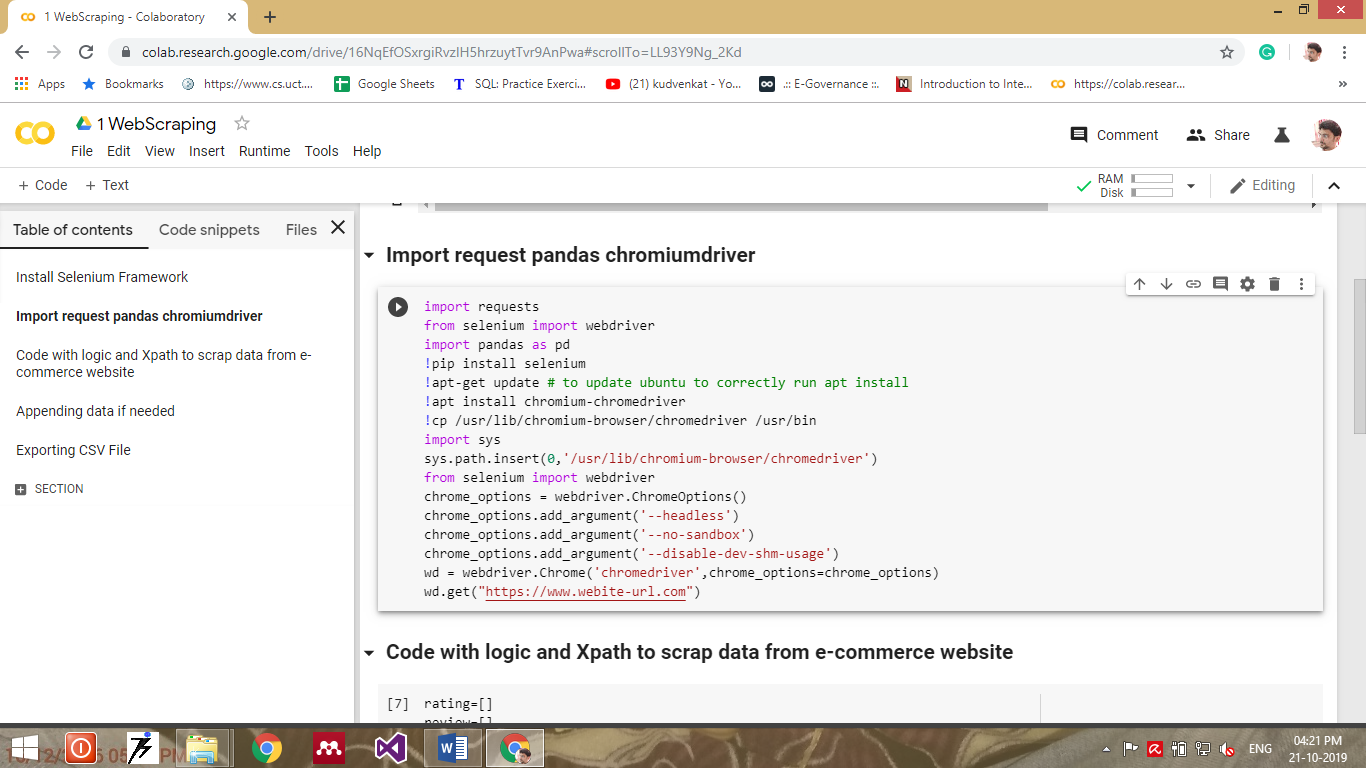


Fig. 7 Importing Pandas & Chromiumdriver

### **Programming with Xpath to scrape data**

After adding necessary library files, installing driver and configured framework it is required to exploit the XML generated from the targeted webpage from the selected e-commerce website. Using the GET request function of the web driver intended link is exploited and to further mine XML file and get the necessary data from it, an Xpath has been used in generated logic to loop through repeated data in XML file. To select the node from the XML document, Xpath is the appropriate query language. Furthermore, to compute values from the XML document content XPath is useful. The World Wide Web Consortium has defined the Xpath. Given figure mentioned the way logic was implemented with Xpath.

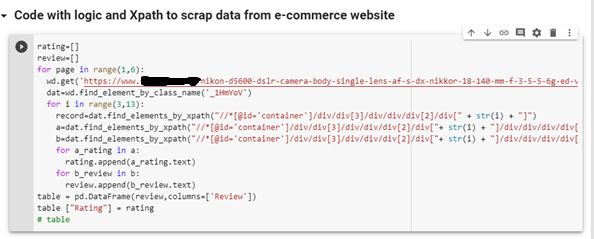


Fig. 8 Xpath usage to scrape data

### **Exporting extracted data**

After applying logic with the help of Xpath, and extracting the required data, it is highly required to convert that data into the tabular form through which further processing could be done. For that we used a feature of pandas library file and exported data into csv format for the use in the next phase. Fig. 9 depicts the exporting data into CSV file format.

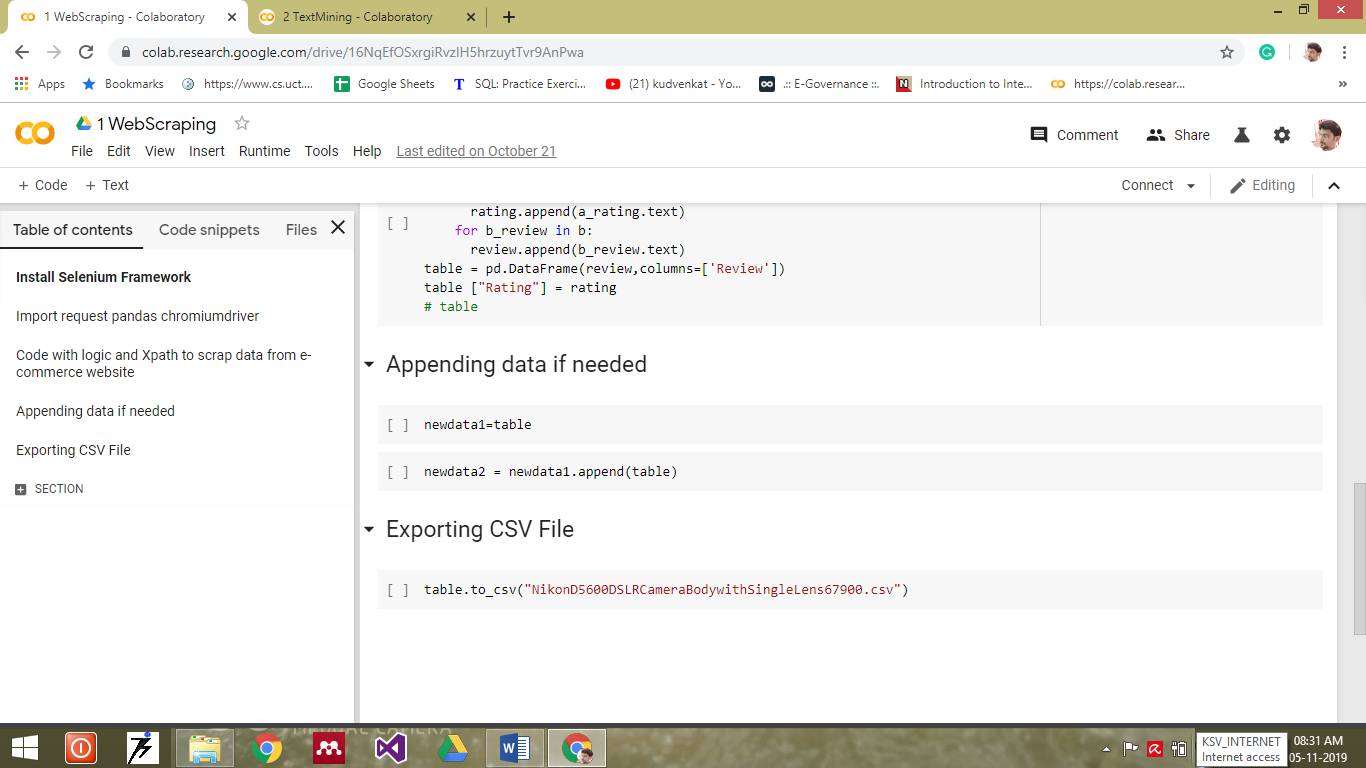


Fig. 9 Exporting scraped data in csv file

### **Primary Data Collection**

Performance evaluation of the system, which takes input of a file containing 75026 records of different products such as Mobile Phones, Refrigerator, Television, Air Conditioner, Washing Machine, and Laptop which has the following attributes Review, Rating. Classification of the extracted data is given below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Category** | **Total Reviews** | **Total Model** | **Product** |
| 1 | AC | 2636 | 6 | 18 |
| 2 | Laptop | 4570 | 2 | 6 |
| 3 | MobilePhone | 31880 | 9 | 27 |
| 4 | Refrigerator | 11090 | 7 | 21 |
| 5 | TV | 13930 | 7 | 21 |
| 6 | WashingMachine | 10920 | 7 | 21 |
| Total | 6 | 75026 | 38 | 114 |

Table 1: Summary of scrapped data

## **Data Refining**

Data preprocessing is very important, any step moving forward without clear understanding and doing proper pre-processing may lead to massive error or we may put ourselves into a mash. Hence it is highly required to perform data pre-processing which includes data cleaning and then implementing a bag of word model to find keywords for the further use. Following figure depicts the way we achieve pre-processing tasks.

Feature Selection

Data Cleaning

Text Transformation NLP

Bag of Words Model

Improved Frequent Pattern Mining Algorithm

Finding Rank of Keywords

Fig. 10 Pre-processing and Learning workflow

### **Feature selection**

In order to apply pattern mining on scraped data, it is required to have implemented feature selection which is one of the core concepts in ML, it impacts hugely the model performance. The data features that we use to train our proposed model have a great impact on the performance we could achieve. Features that are not relevant may influence the model performance negatively. Hence it is highly recommended to clean the data and select appropriate features before moving further.

By applying feature selection, we could avoid the problem of Over fitting and could be benefited by Improves Accuracy and Reduces Training Time. In this process we can automatically or manually select those features which contribute most to our prediction variable or output in which we are interested in. We have selected feature manually and we have selected two crucial feature which could highly impact our model those features are rating and reviews.

### **Import File from previous phase**

After performing feature selection, we are now ready to perform necessary task on the gathered data collected in the data information collection phase. After importing necessary library files i.e. pandas, google-collab files and matplotlib.pyplot, we then imported the csv file which was exported from the previous phase. Fig. 11 shows the basic code to implement the same.

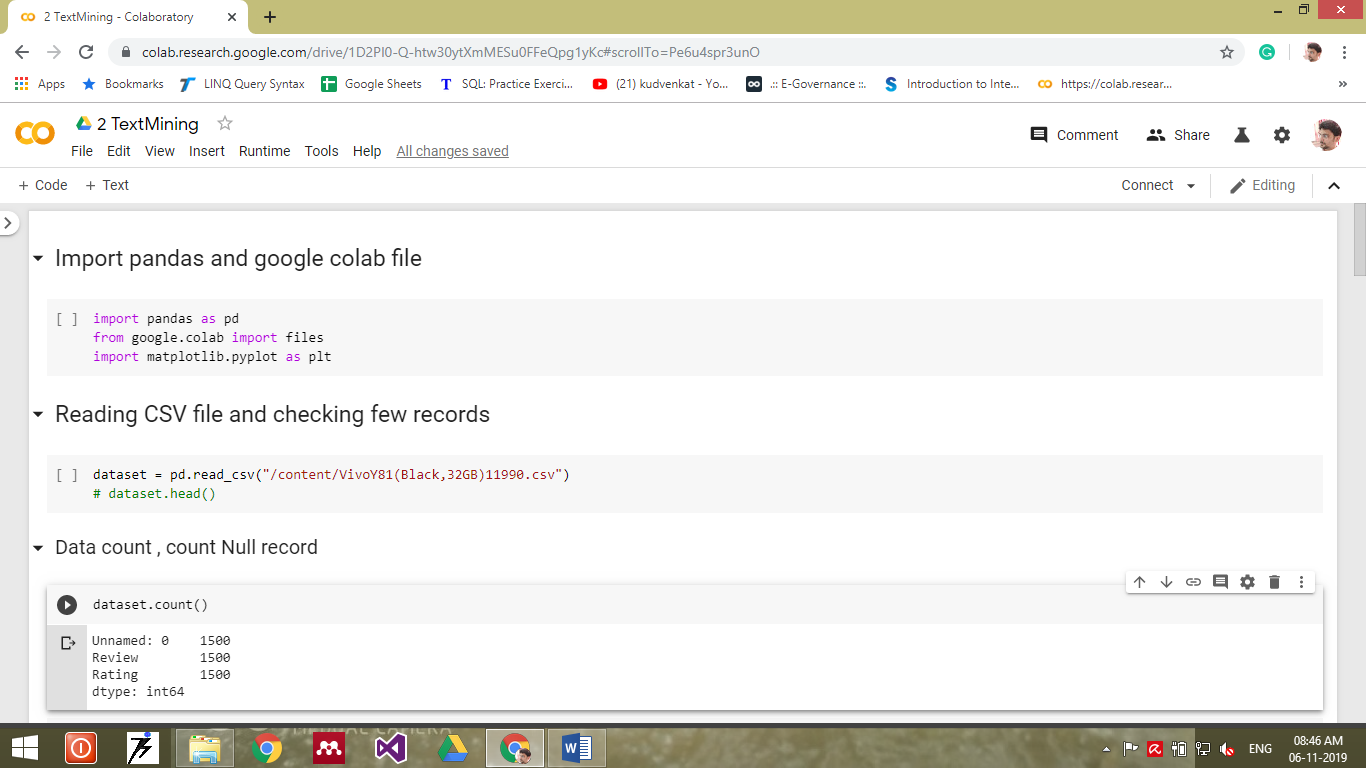


Fig. 11 Importing CSV file

### **Data Cleaning**

Our work is on users’ review and ranking of products received from e-commerce sites. By implementing a successful python script, we gathered enough data to deal with, while dealing with ranking it is not difficult to handle numeric data but when working on reviews, which is text including special symbols, emoji and special characters which are not required in further steps. Hence appropriate data cleaning code in python script was generated and we get rid of unwanted content from the reviews in data cleaning. Following two figures Fig. 12 and Fig. 13 show how cleaning processes happened via python script.

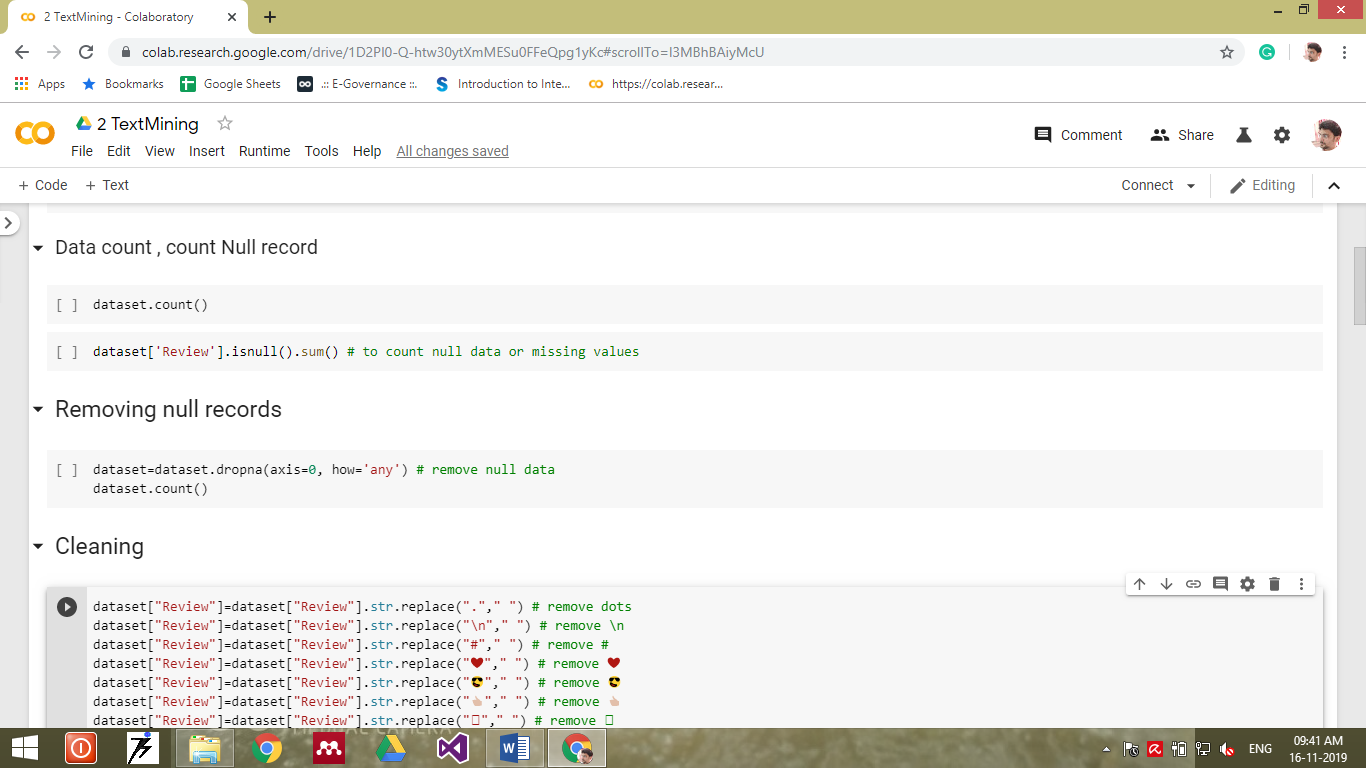


Fig. 12 Removing null records

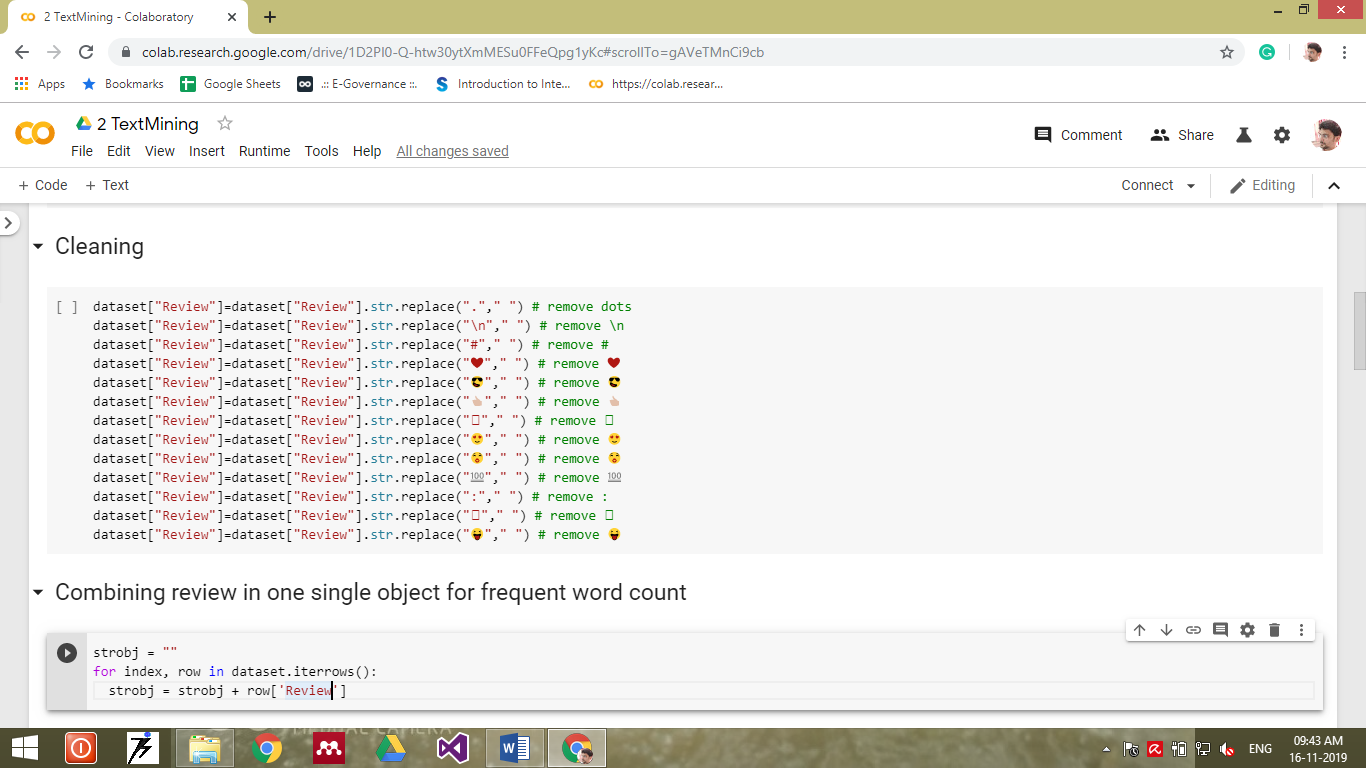


Figure 13: Data Cleaning

### **Text Transformation and NLP**

After implementing feature selection and data cleaning, in our research work at this stage we are readily available with the exact content we want to deal with. The content available with us are users’ rating and reviews. It is quite easy to deal with ratings which are numerical data and could be fitted in machine learning algorithms easily, but the same is not the case with reviews. It is required to transform the text from categorical to numerical so that it could be fitted into a machine learning algorithm and produce the required result. We have processed the natural language using a bag of word models and transformed the text from categorical to numerical. To achieve desired output from the text we required to collect all individual reviews into a single string object. Fig. 14 depicts the implementation of the same.

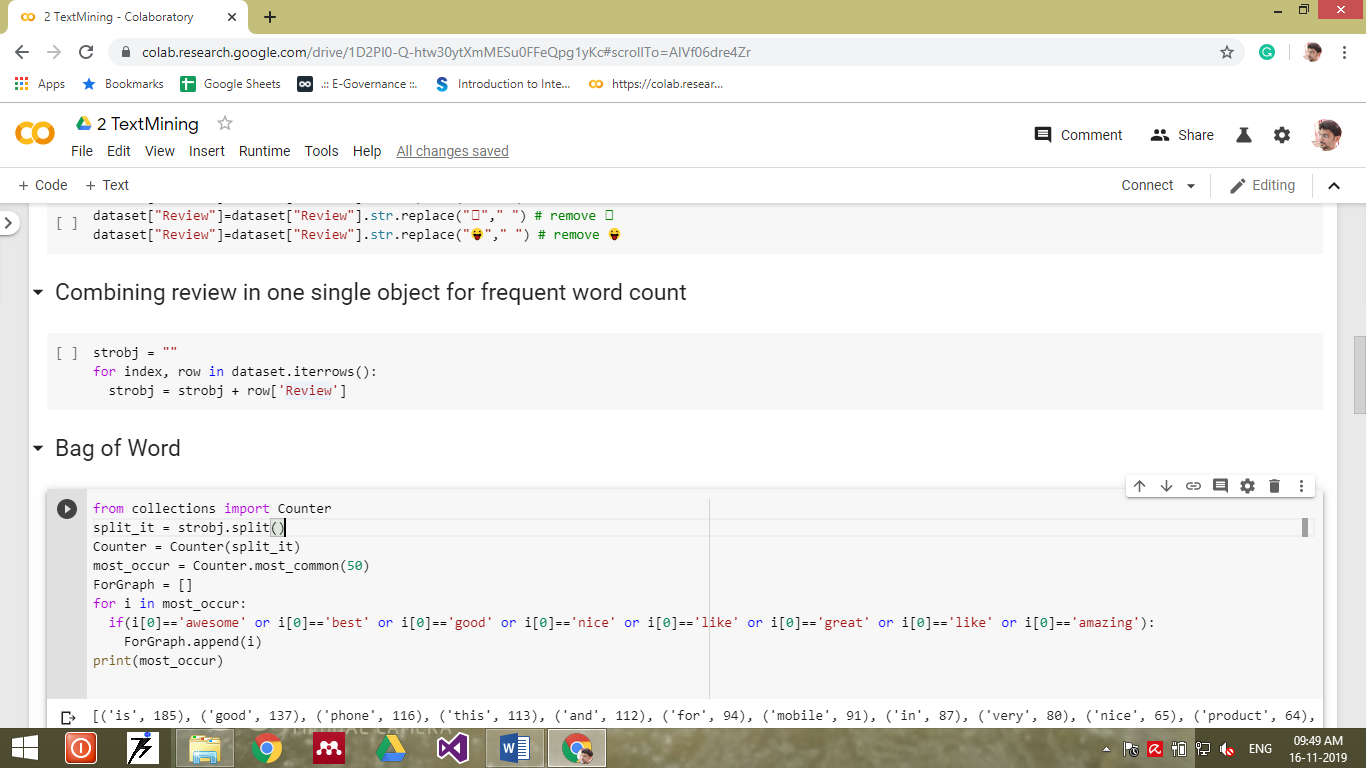


Fig. 14 Combining all reviews in one object

### **Bag of Words model**

Bag of words model is an information retrieval through natural language processing (NLP) to simplify representation. A text document is represented as a bag of words it contains and by ignoring the grammar and word order it gives the multiplicity of words in the document given as an input to it.

In our python script all text content collected in one object called bag of word and then by applying the Bag of Words model to it gives the most frequent word in it, which further helps in finding key words to convert key information from categorical to numerical in order to implement further recommendation algorithms on it. Following figure Fig. 15 shows the implementation of the “Bag of Word model” in our python script.

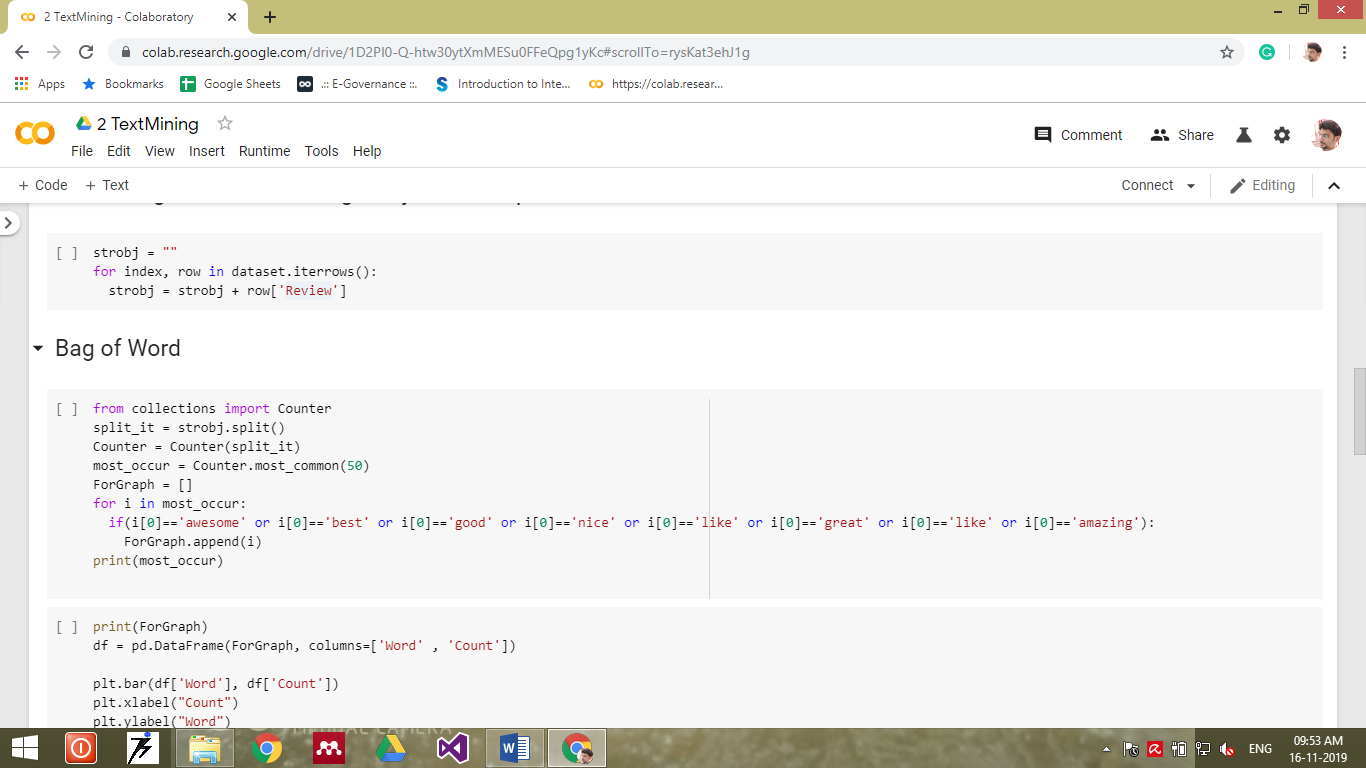


Fig. 15 Bag of Word Model

### **Improved Frequent Pattern Mining**

Frequent patterns are sets of data or item sets which frequently appear in the available data set which has frequency greater than or equal to the specified threshold. Mining frequent pattern is actually data mining which has an objective to extract frequent data or item sets from the given data set. Finding frequent patterns from the data set is very important and it plays a very significant role in different data mining tasks. By applying a bag of word model we have identified frequently appearing positive words from the reviews and based on the frequency we have selected the positive keyword to further assign rank to each positive word.

### **Selecting Keyword**

From the Bag of Word model we could identify the frequent pattern appearing in the data, but it is equally important to select appropriate keywords to assign rank. In order to select the correct key word we have then used a graph from the matplotlib and try to find which keyword is to be selected. Then we get the clear idea to select the following five key words namely “good”, “nice”, “like”, “awesome” and “best”. We also keep the counter updated by negating the positive word used in negative nouns. For example, good is a positive word but when it is preceded by not, it becomes not good. We took care of such occurrences and then we prepared and selected the key word. Following figure Fig. 16 depicts graph implementation and how we select the keyword.

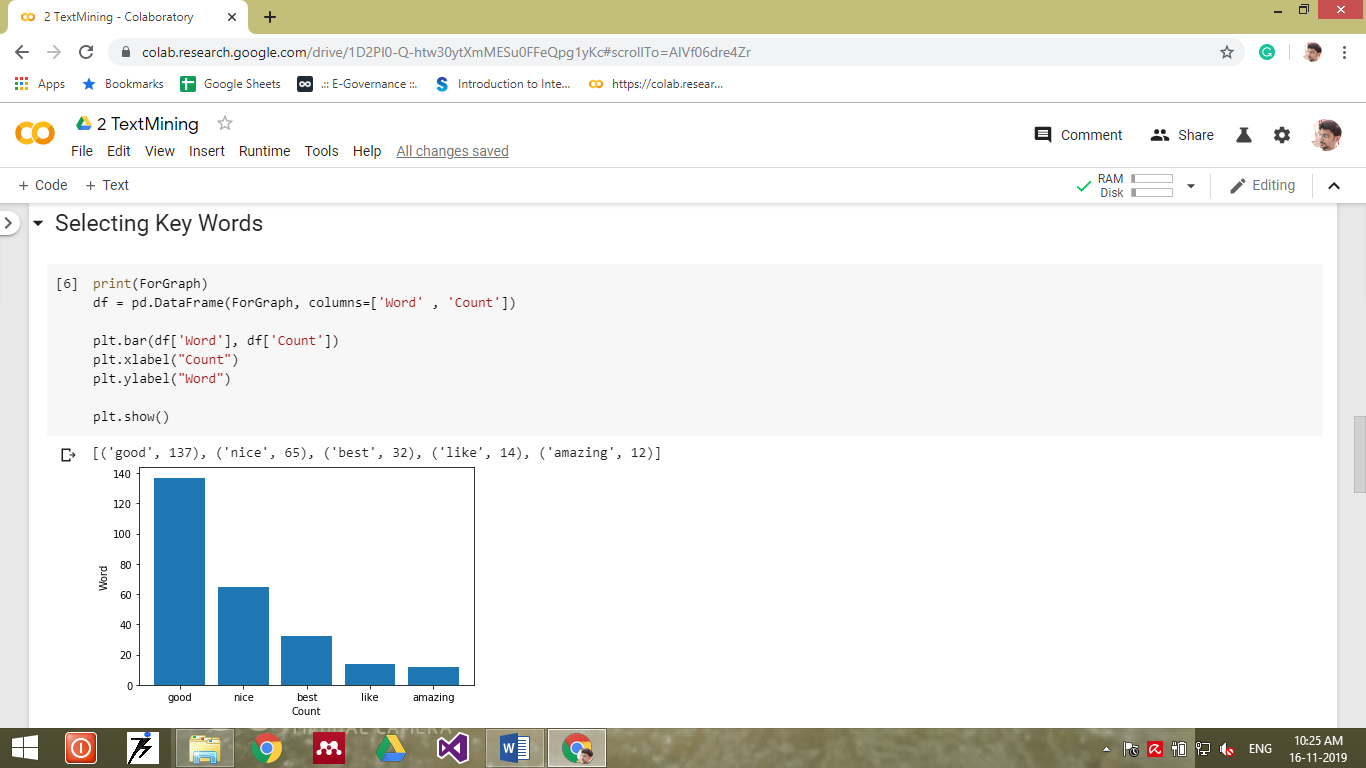


Fig. 16 Selecting Keywords

### **Finding Rank of Keywords**

In our research work as discussed in the previous paragraph, we found a few keywords which are frequently used in most of the users’ reviews by applying a bag of word model, to convert key information from categorical to numerical it is highly important to maintain appropriate weightage of each product review we gathered. In our observation while extracting data from the web one product might have 80 numbers of reviews and another product might have 2200 reviews and hence to give equal weightage to each key word, we took average mean of the number of reviews and then ranked keyword accordingly, which finally convert categorical data to numerical. Figure depicts the code implementation in python script for filtering the positive word count by eliminating the positive word preceded by natation.

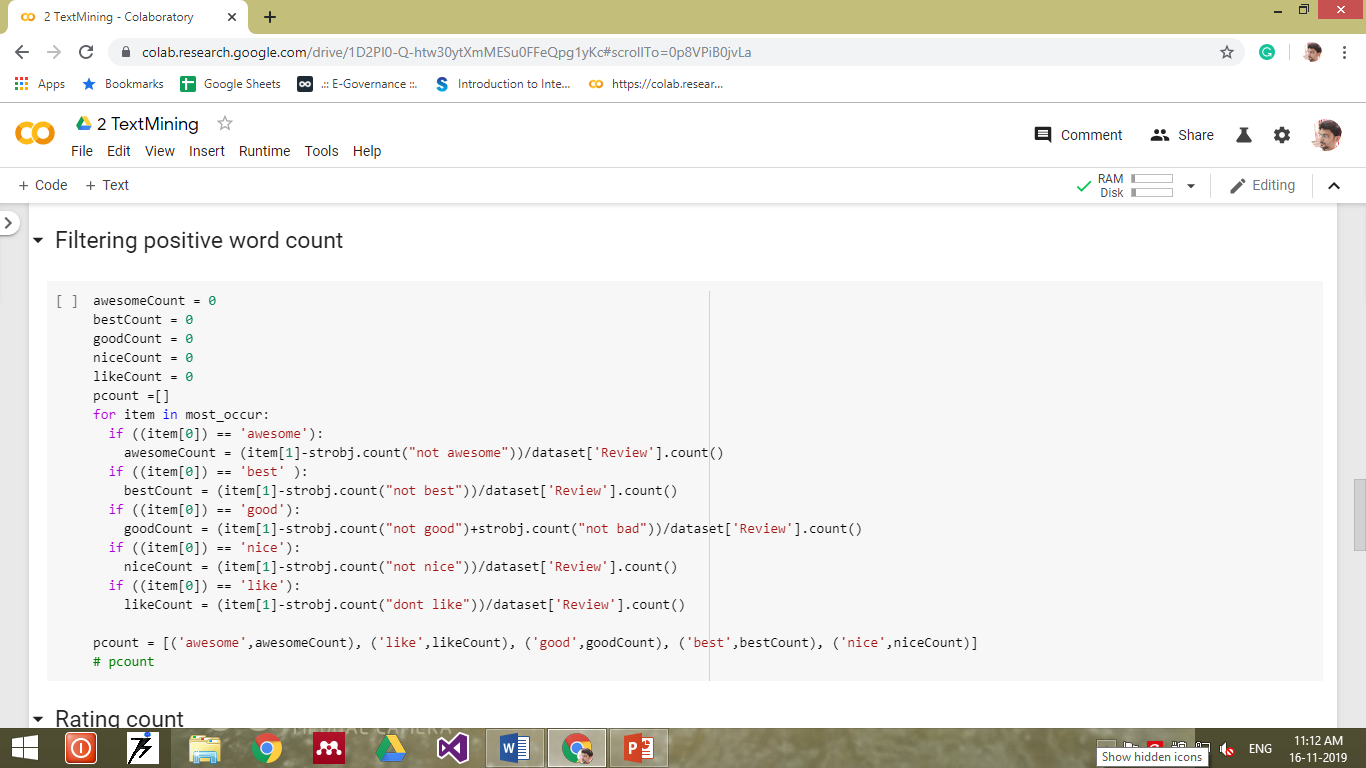
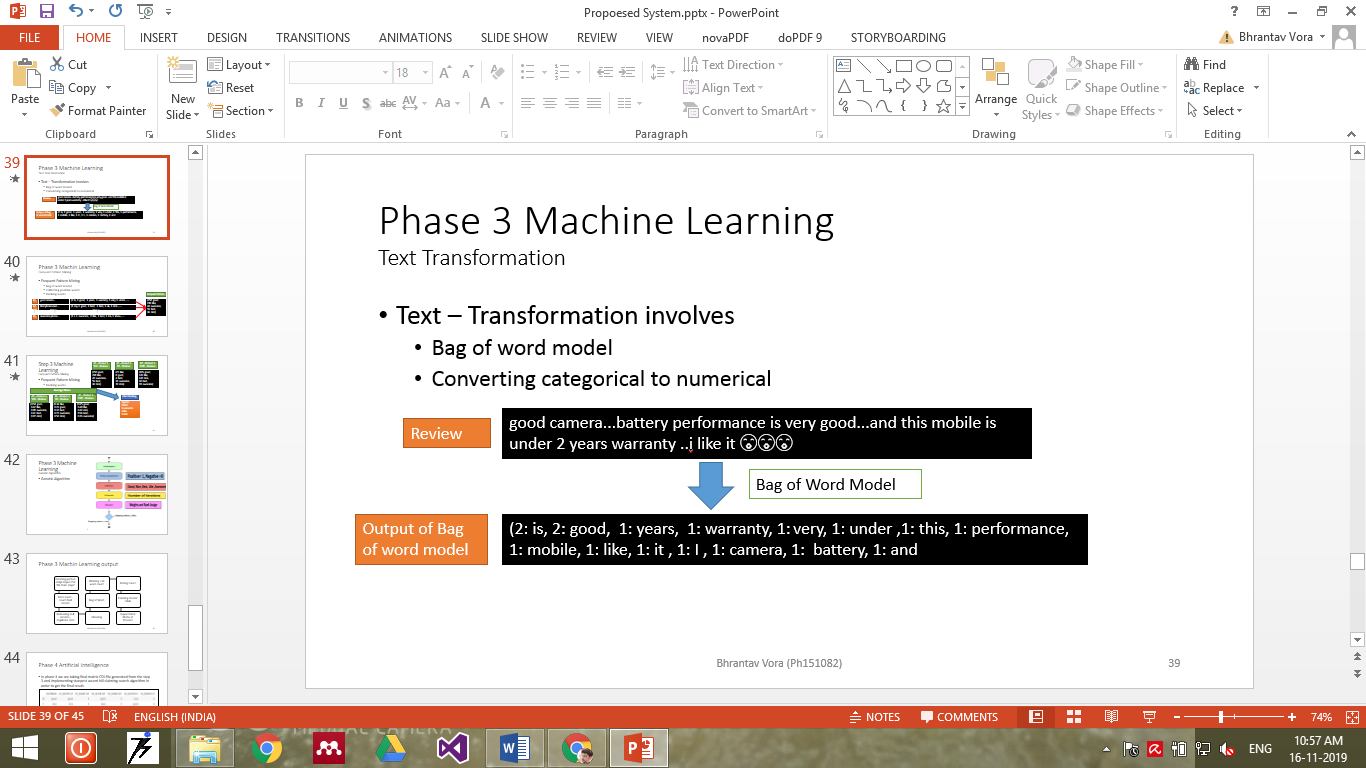
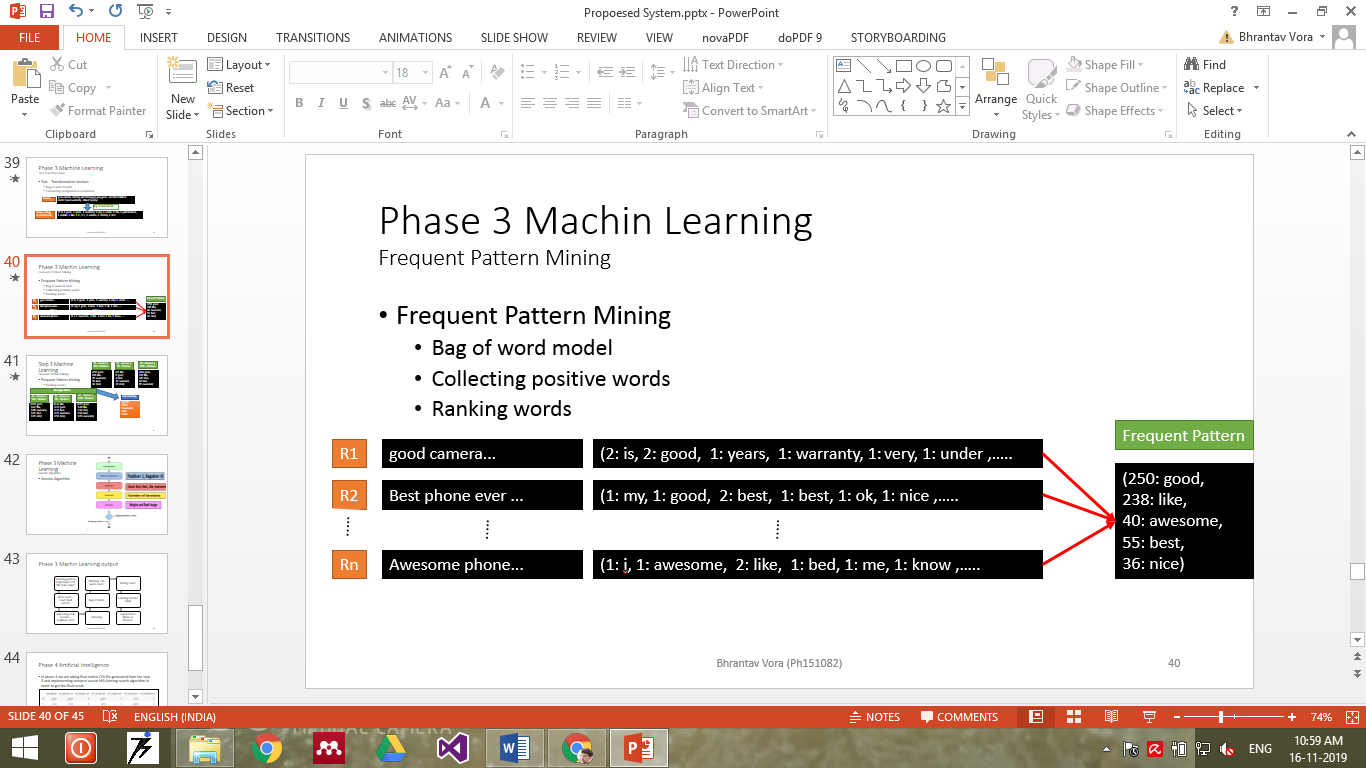
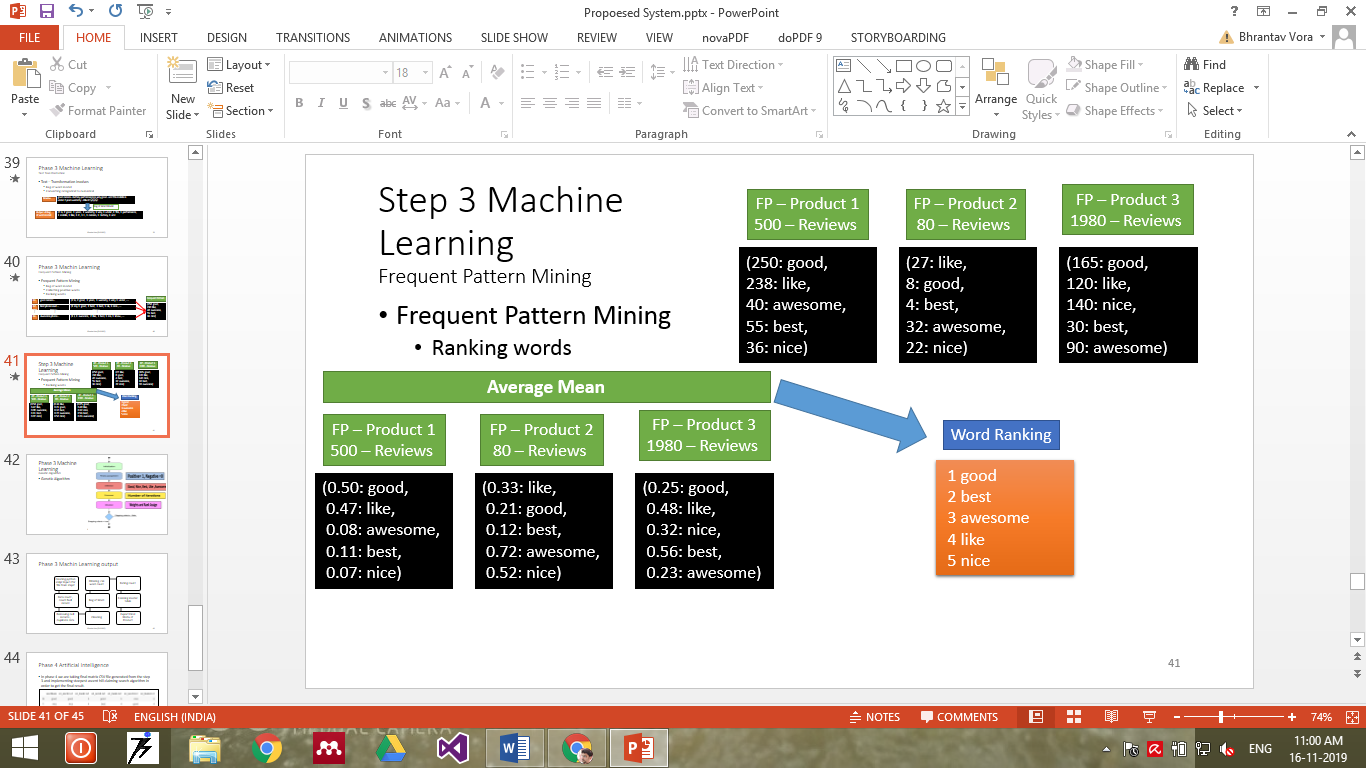


Fig. 17 Filtering positive word count

Following is the pictorial representation of find keyword and given them rank form the row data that is reviewed.







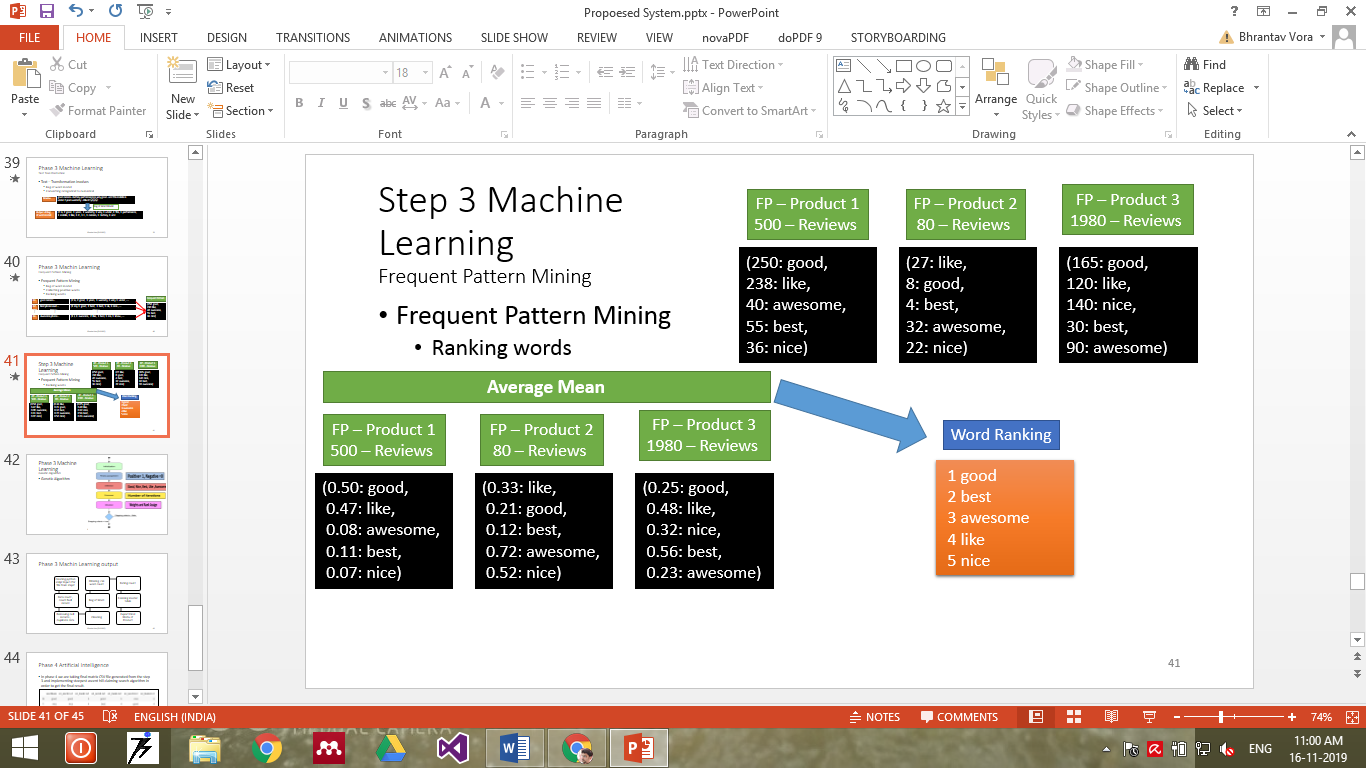


Fig. 18 Process of finding Word Rank

### **Data Transformation**

We are preparing and formatting a master table for each product which contains selected keywords with its weightage and rate with its weightage. This master table for each product will be an input for the next phase of recommendation and prediction. Following figure illustrates the coding work to generate the master file and then export it into a csv format.

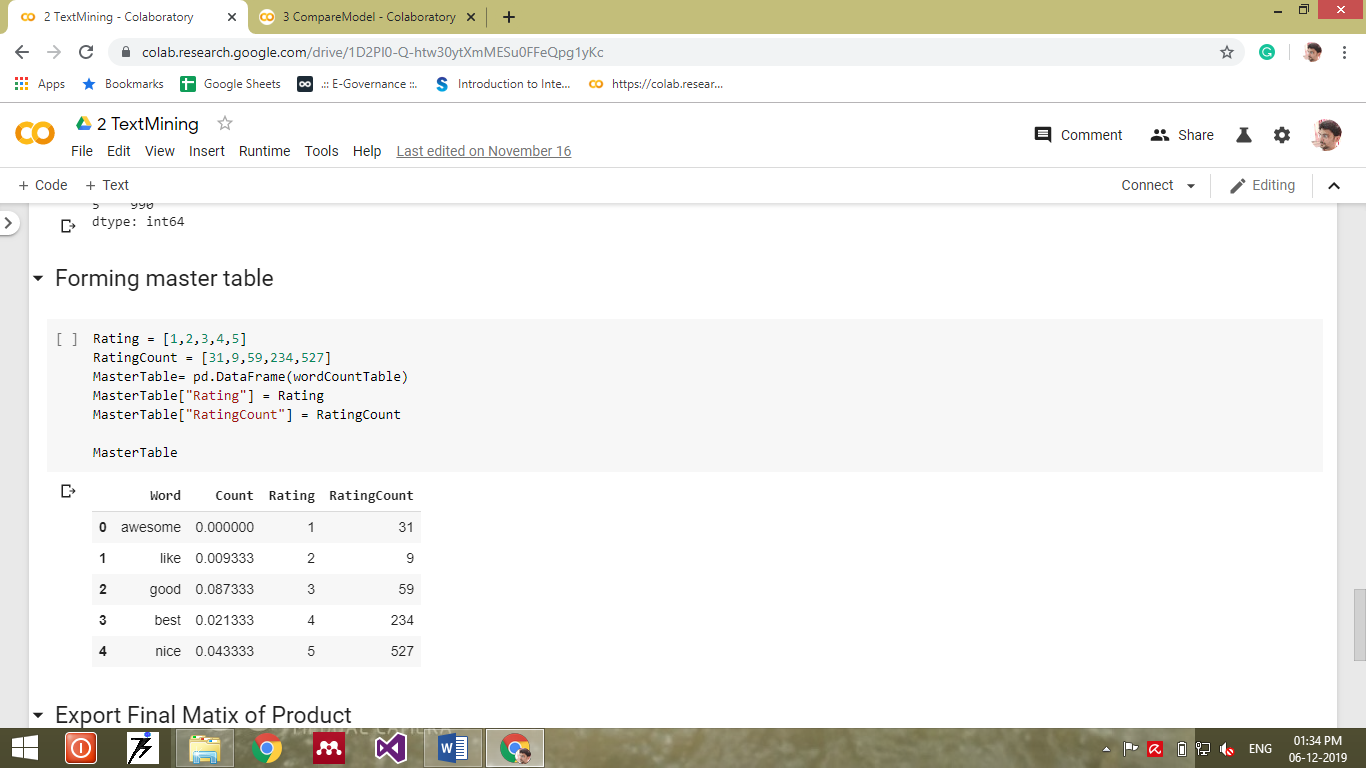


Fig. 19 Forting Master Table

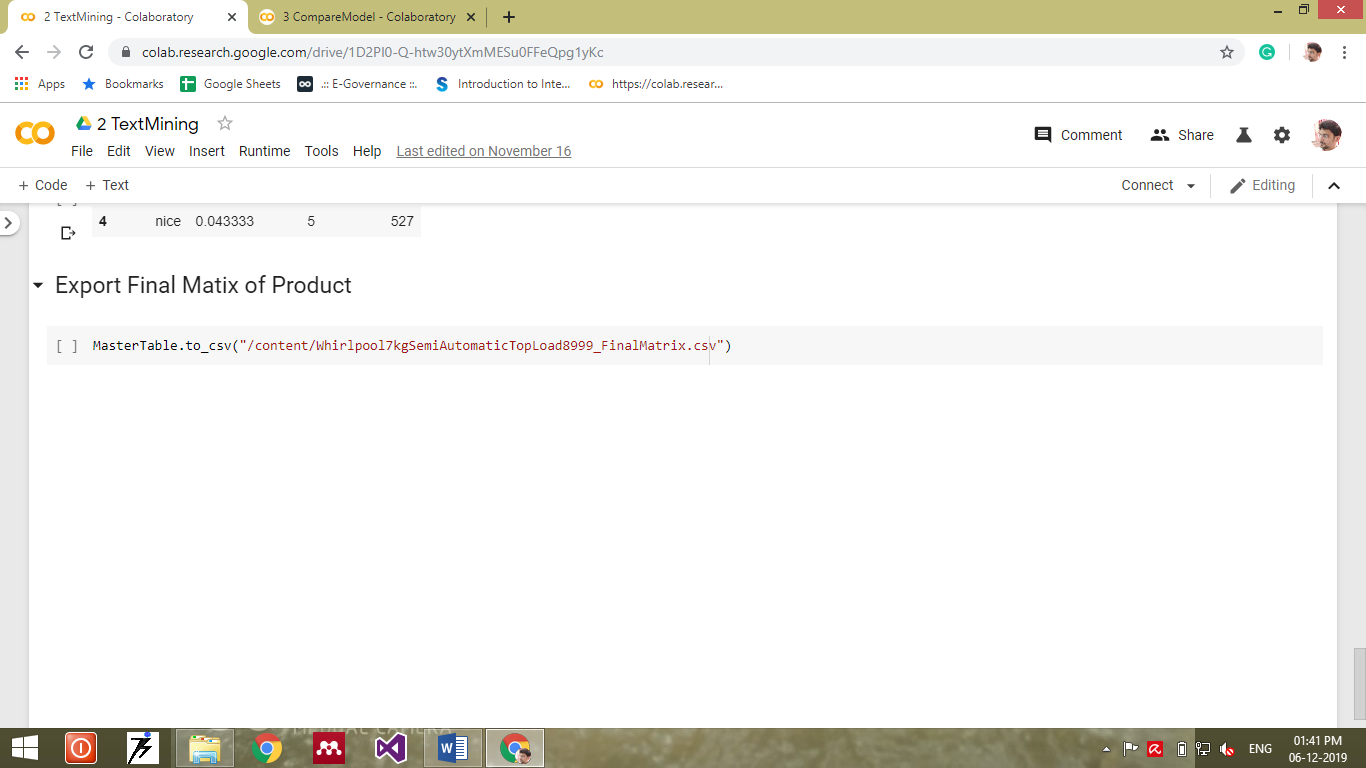


Fig. 20 Exporting final matrix of product in CSV file

## **Recommendation Algorithm**

In experimental procedure, after data collection and preprocessing, it is predicting or recommending the product a user will be interested in or most likely to purchase. For the final recommendation we have prepared a product final matrix from the learning or pre-processing phase and given it as an input to the steepest ascent hill climbing algorithm to predict the final product recommendation. Fig. 21 illustrates the steps of including the recommendation algorithm.

Input products final matrix

Implementing word count and assigning rankt to the words

Converting word rank from categorical to numerical

Calculating Rate Rank

Preparing final list of products with RateRank and WordRank

Implementing steepest accent hill climbing alogorithm on final list

Receiving recommendation

Fig. 21 Recommendation/ Prediction Workflow

### **Importing necessary library files**

To work with the data generated in previous phase we are creating new python script in which we are importing pandas and numpy library files to generate a graph and form data in tabular format, we are also using and importing files Google colab to deal with files in buffer area which keep file in buffer for next 12 hours. Figure Fig. 22 shows the panda, numpy and Google collab files getting imported.

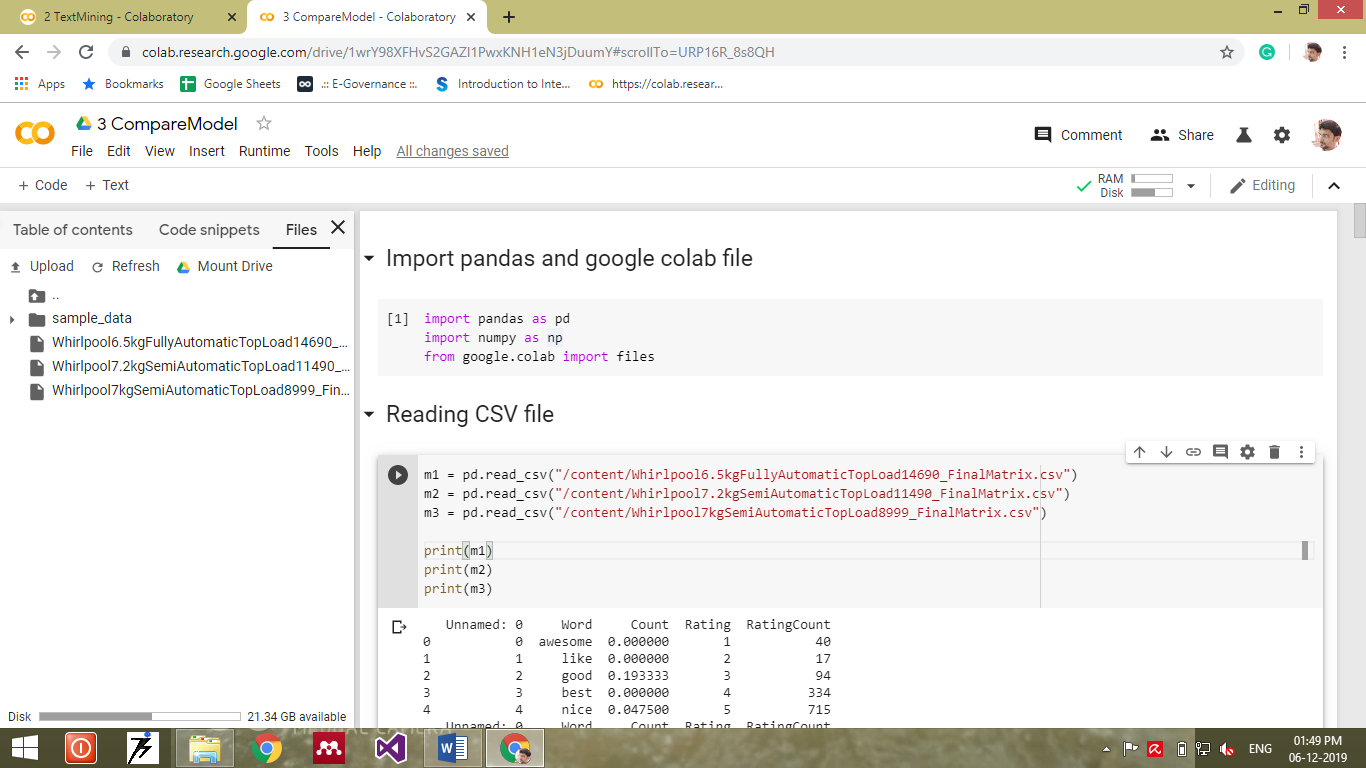


Fig 22: Importing pandas and Google FIle

### **Reading data from the CSV files**

In the recommendation phase, we took more than one different product’s details generated in the previous phase and worked on them for further processing and coming out with the most suitable recommended product out of given products. Figure Fig. 23 shows the reading of three different models’ data reading.

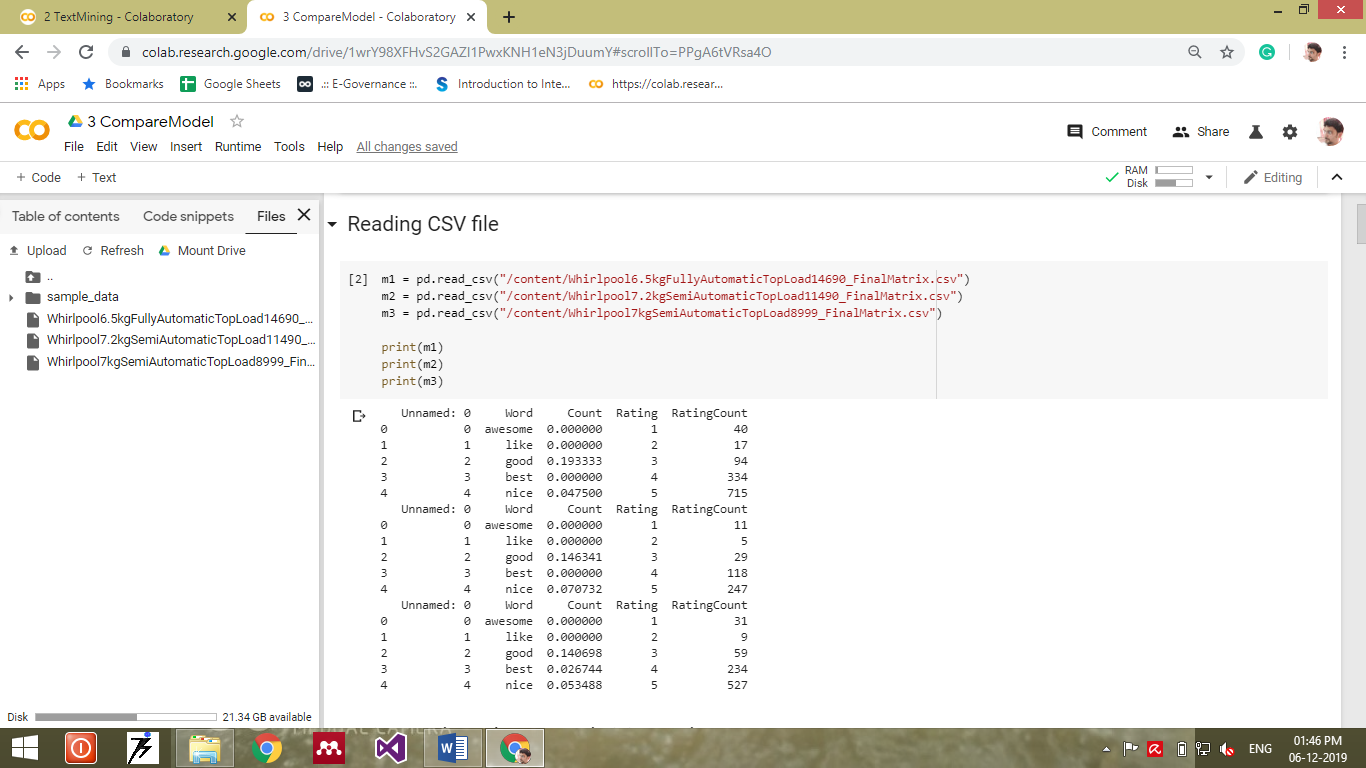


Fig. 23 Reading CSV containing final product matrix

### **Giving Rank to the word**

It is very important to give rank to the word to do unified analysis among the different products. After finding keywords by applying a bag of word model and pattern identifying and finding weighted average mean in the previous phase, we are now giving appropriate rank to the key word. This process helps in finding word rank, which converts categorical data into numerical data. Given figure Fig. 24 shows the code of finding rank of the keyword and converting it from categorical to numerical



Fig. 24 Ranking word

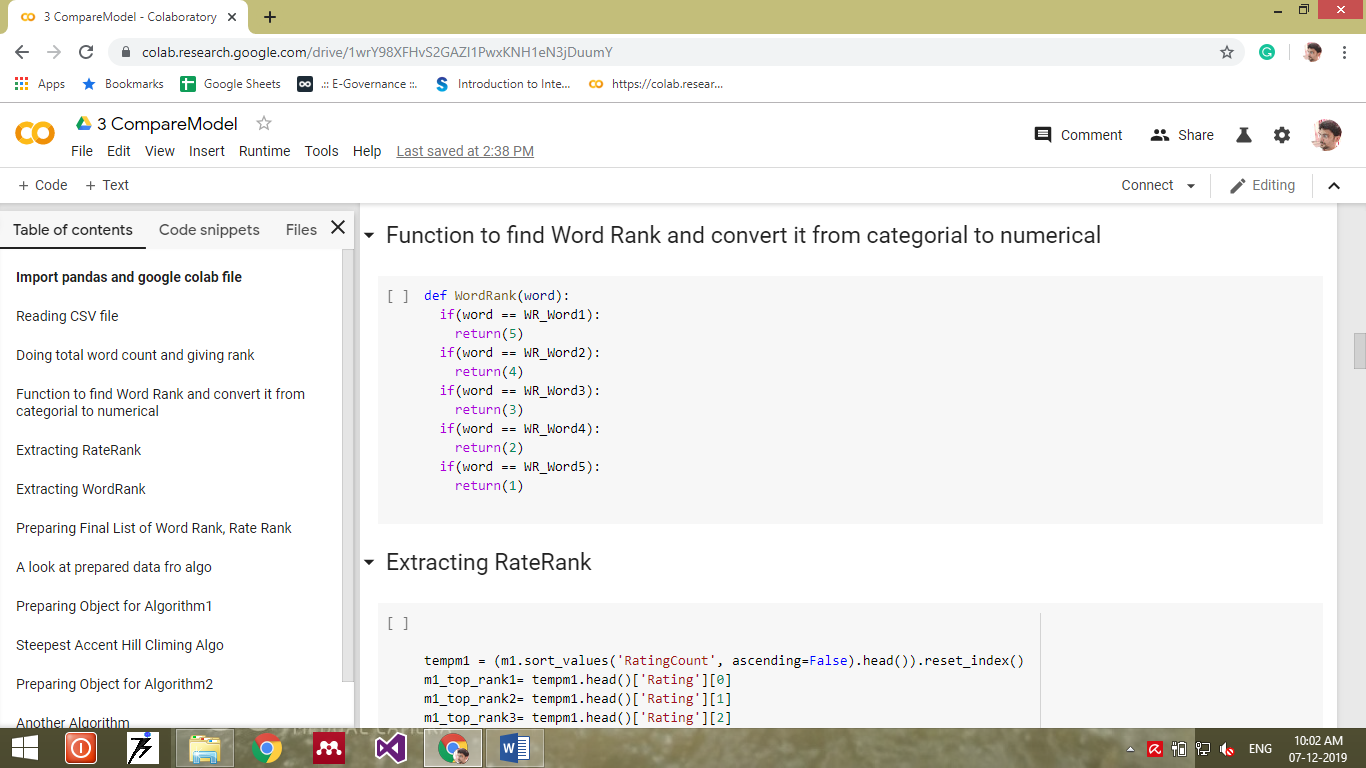


Fig. 25 Function to find word rank

### **Preparing Product Final Matrix**

In the process, further we accumulate the detailed review and rating of the product, we arrange the rating by calculating the sum of the particular rating given by the users which is an easy task and then we work on the review. As mentioned in the above explanation, by converting categorical information to numerical we assign rank to each keyword and arrange them accordingly. This will produce the final matrix of the individual product containing its importance by rating and review wise in the final product matrix. Following figure is just a look at the prepared data which will generate the final product matrix on which the Steepest-Ascent Hill climbing algorithm is applied to find the best suitable product to recommend.

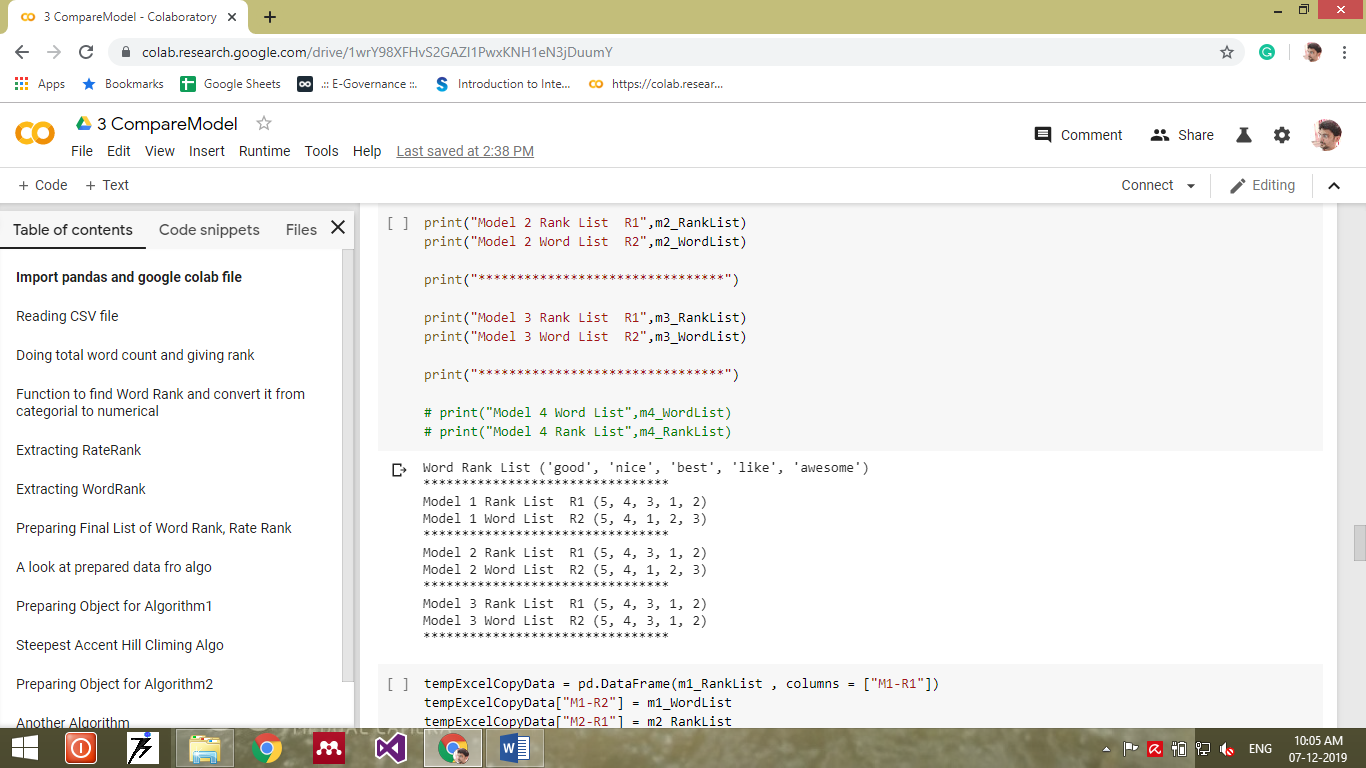


Fig. 26 Product Final Matrix

### **Steepest-Ascent Hill climbing algorithm**

Heuristic search used by the Hill Climbing for mathematical optimization snags in the arena of Artificial Intelligence. In Steepest-Ascent Hill climbing technique, before selecting any node, it first inspects all the neighboring nodes and then selects the node nearest to the goal state as of next node. In our algorithm we used the same technique to find the goal state.

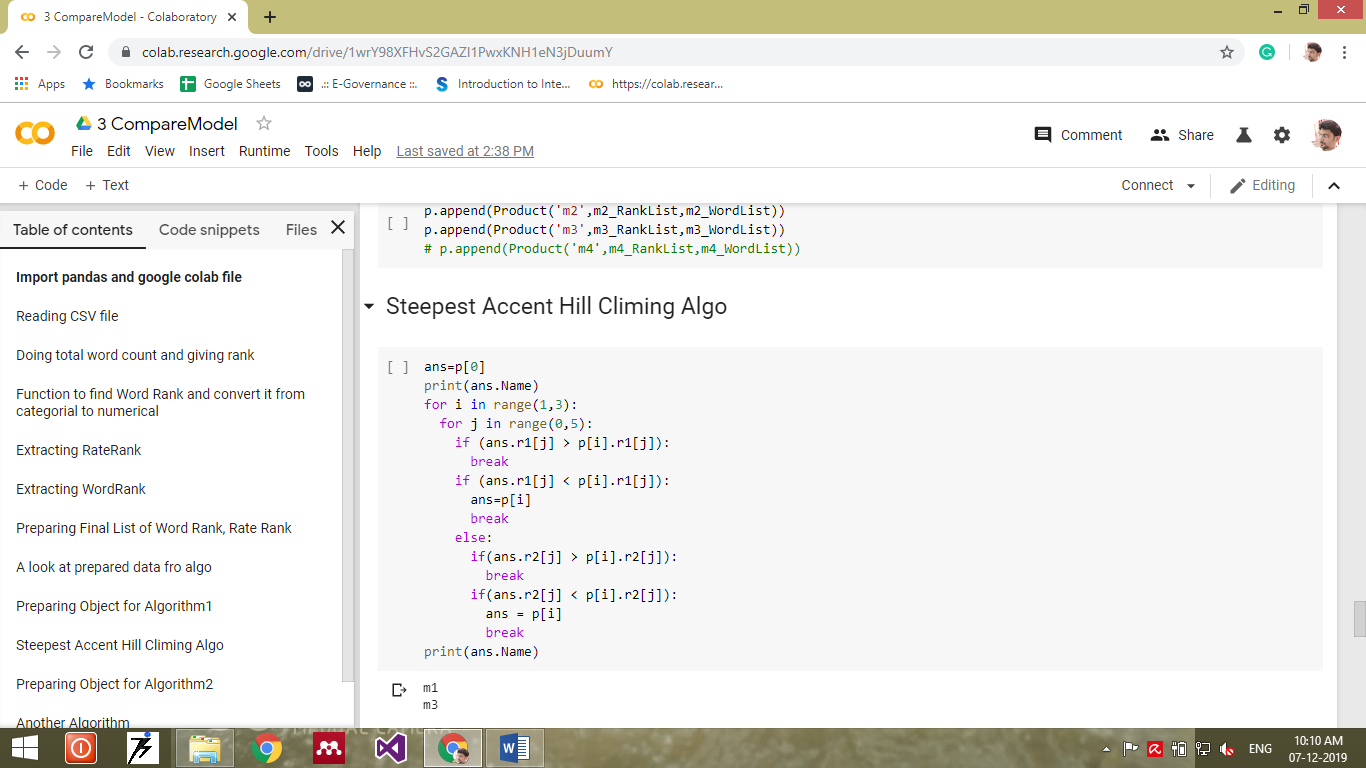


Fig. 27: Steepest Ascent Hill Climbing Algorithm - I

In further advancement, once we have the final product matrix of more than one product we can then give them as an input to our Steepest-Ascent Hill climbing algorithm. This model is an enhanced version of an artificial intelligence algorithm named steepest ascent hill climbing algorithm. In steepest ascent hill climbing algorithm we can find for the best neighboring node which is nearer to the goal state and finally we get the best available node, this is how we could recommend the best product out of given products, in enhancement our developed Improved Steepest-Ascent Hill climbing algorithm, we provide level of recommendation and give freedom to the user to select based on the level of given recommendation. Given figure depicts the code implementation for the Steepest-Ascent Hill climbing algorithm - II.

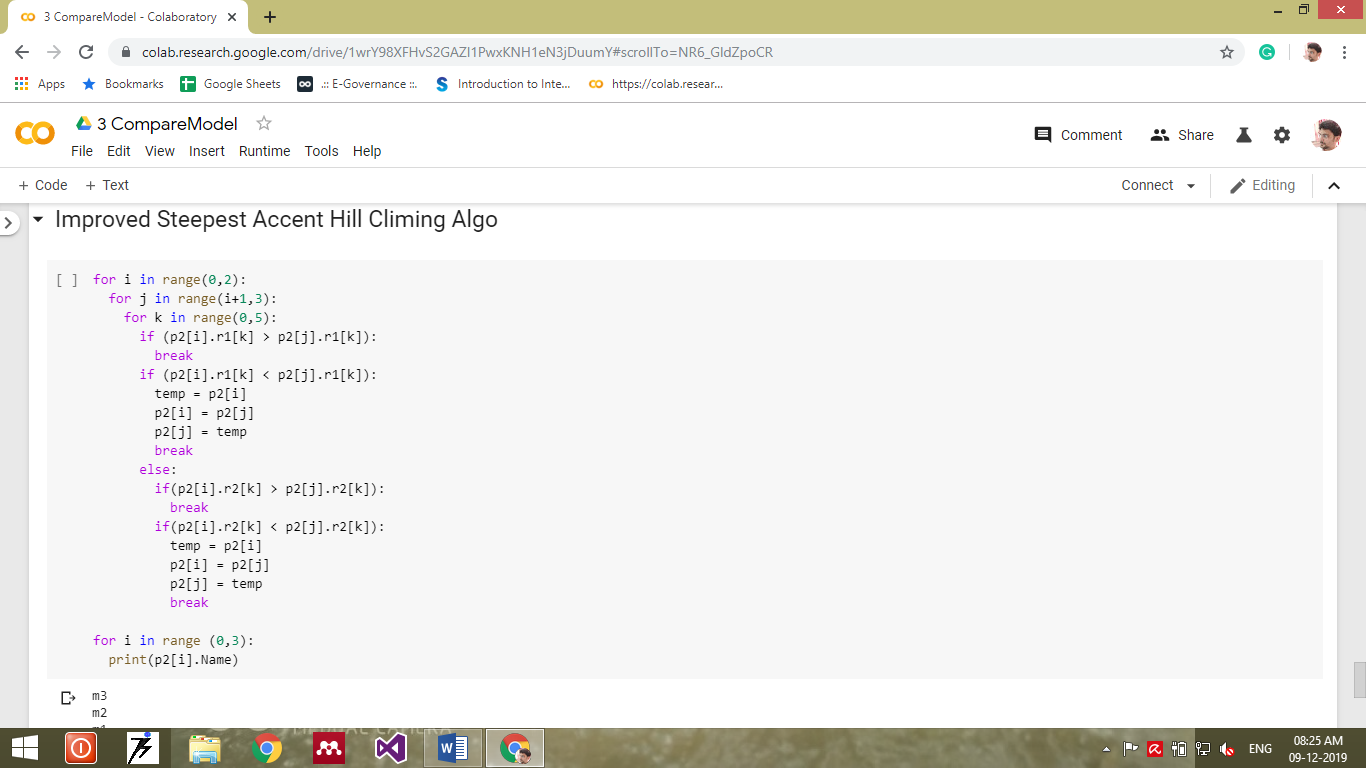


Fig. 28: Improved Steepest Ascent Hill Climbing Algorithm - II

# Result

In the above-mentioned experimental procedure we have identified six different products in the electronic segment and we have divided products under the category of different brands, hence under each brand we have collected data for at least three different models and work on that. In the final attempt we collected data i.e. customers’ review and rating of different 114 products which can be divided into 38 different models and total reviews and ratings collected was 75026. After doing pre-processing on gathered data and further passing it to the recommendation or prediction phase we received results to recommend the product at different levels which further passed as input to Improved Steepest Ascent Hill Climbing algorithm to find the proof of confidence.

## **Evaluation Process**

The recommendation system developed has been evaluated by two different Proofs of Concept Proofs of Concept were tested on the developed recommendation system with different hypotheses and tested against the actual result [28].

### **Proof of Concept**

POC 1: If *rating and ranking of the product duplicates in the final level* while performing the recommendation algorithm as in simultaneously extracting data from the e-commerce portal making the algorithm reach a bottleneck situation.

The rating rank of two product duplicates then the algorithmic iterated, the next following result will enhance the recommendation followed by the next level search

POC2: If *ranking and rating duplicity persists in all levels* of searching the highest rank, while performing the recommendation algorithm as in simultaneously extracting data from the e-commerce portal making the algorithm reach a bottleneck situation. If the search result is duplicated in all levels then the algorithm is shifted for the next feature extraction.



Fig. 29 Proof of Concept Justification

### **Performance evaluation**

Wide range of available metrics are used to measure the accuracy of the developed recommendation system. Two classified categories of metrics used to estimate the recommendation quality and evaluate the accuracy of the recommendation system. RMSE, MAE, RAE are among the few popular prediction metrics to evaluate the quality and accuracy of the developed recommendation system.

Random tree algorithm has been tested against the generated result from the recommendation system

|  |  |
| --- | --- |
| **Criteria** | **Unit / Result** |
| Cross validation fold | 10 |
| Correctly classified instances | **94.8148%** |
| Incorrectly classified instances | 5.1852% |
| Kappa statistic | 0.9222 |
| MAE | 0.0352 |
| RMSE | 0.1868 |
| RAE | 7.9189% |
| RRSE | 39.6186% |

Table 2 : Performance Evaluation

Root Mean Square Error (RSME) is a variant of Mean Square Error (MSE), and before summing an error it squares the errors. Root Mean Square Error method has been used quite frequently as a performance evaluator. The recommended and observed rating difference is processed through this method. In our observed result in Weka 3.8.4 the RSME is 0.1868. MEA is Mean Absolute Error measures, the difference between recommendation and actual rating and its absolute standard deviation is measured by MEA. The interpretation and implementation of this method is very easy and hence it is used most frequently. In our observation the MEA given is 0.0352 [29].

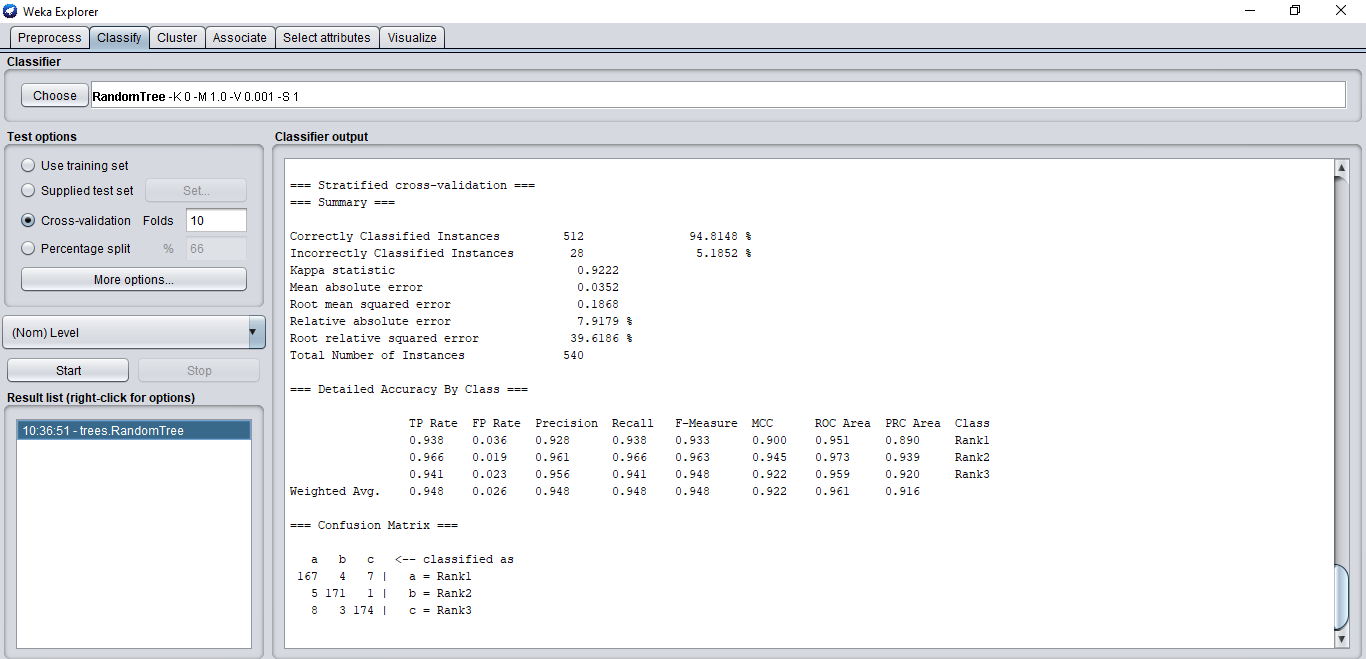


Fig. 30 Weka screen shot - performance evaluation

# Conclusion and Future Scope

This research instigated the product recommendation system for the e-commerce website, which recommends the most likely to purchase product or product at different level of recommendation to the customer using Steepest-Ascent Hill climbing and Improved Steepest-Ascent Hill climbing algorithm respectively and content-based filtering, based on the customers’ rating and review. For the customers’ convenience we developed scripts and scraped data directly from the e-commerce websites. NLP is used to process customers’ reviews and extract relevant information from the reviews, further based on the rating and reviews both collectively we have developed two algorithms namely Steepest-Ascent Hill climbing and Improved Steepest-Ascent Hill climbing. Our algorithm has a limitation of the cold start problem in the recommendation system. For the newly launched product or the product which has no reviews and rating given from the customer is not covered in this research. For the future scope we could focus on the cold start problem in the recommendation system with our proposed Improved Steepest-Ascent Hill climbing algorithm.

##### REFERENCES

[1] C. Y. Li and Y. C. Ku, “The power of a thumbs-up: Will e-commerce switch to social commerce?,” *Inf. Manag.*, vol. 55, no. 3, pp. 340–357, 2018, doi: 10.1016/j.im.2017.09.001.

[2] N. Boysen, R. de Koster, and F. Weidinger, “Warehousing in the e-commerce era: A survey,” *Eur. J. Oper. Res.*, vol. 277, no. 2, pp. 396–411, 2019, doi: 10.1016/j.ejor.2018.08.023.

[3] P. D. I. Hernando, “Electronic commerce,” *Comput. Law Secur. Rev.*, vol. 19, no. 5, pp. 363–374, Sep. 2003, doi: 10.1016/S0267-3649(03)00504-1.

[4] E. Turban, D. King, J. K. Lee, T.-P. Liang, and D. Turban, *Electronic Commerce: a Managerial and social Networks Perspective 2012*. 2012.

[5] T. Garín-Muñoz, R. López, T. Pérez-Amaral, I. Herguera, and A. Valarezo, “Models for individual adoption of eCommerce, eBanking and eGovernment in Spain,” *Telecomm. Policy*, vol. 43, no. 1, pp. 100–111, 2019, doi: 10.1016/j.telpol.2018.01.002.

[6] J. Jooa, S. Bangb, and G. Parka, “Implementation of a Recommendation System using Association Rules and Collaborative Filtering,” *Procedia - Procedia Comput. Sci.*, vol. 91, no. Itqm 2016, pp. 944–952, 2019, doi: 10.1016/j.procs.2016.07.115.

[7] M. Narayanan and A. Kumar, “IMR-DCAL SPECIAL PAPER A study and analysis of recommendation systems for location-based social network ( LBSN ) with big data,” *IIMB Manag. Rev.*, vol. 28, no. 1, pp. 25–30, 2016, doi: 10.1016/j.iimb.2016.01.001.

[8] L. M. De Campos, J. M. Fernández-luna, J. F. Huete, and M. A. Rueda-morales, “International Journal of Approximate Reasoning Combining content-based and collaborative recommendations : A hybrid approach based on Bayesian networks q,” *Int. J. Approx. Reason.*, vol. 51, no. 7, pp. 785–799, 2010, doi: 10.1016/j.ijar.2010.04.001.

[9] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Knowledge-Based Systems Recommender systems survey,” *Knowledge-Based Syst.*, vol. 46, pp. 109–132, 2013, doi: 10.1016/j.knosys.2013.03.012.

[10] R. Katarya and O. P. Verma, “An effective collaborative movie recommender system with cuckoo search,” *Egypt. Informatics J.*, vol. 18, no. 2, pp. 105–112, 2017, doi: 10.1016/j.eij.2016.10.002.

[11] F. O. Isinkaye, “Recommendation systems : Principles , methods and evaluation,” pp. 261–273, 2015, doi: 10.1016/j.eij.2015.06.005.

[12] B. Vora Assit, N. J. Patel Professor, A. Motibhai Patel, N. N. Jani Dean, and G. Dave Assit, “Review Paper introducing E-commerce, Data Mining and Pattern Recognition to recommend the product,” *Res. HUB-International Multidiscip. Res. J.*, no. 03, 2018.

[13] S. Redhu, S. Srivastava, B. Bansal, and G. Gupta, “Sentiment Analysis Using Text Mining : A Review,” *Int. J. Data Sci. Technol.*, vol. 4, no. 2, pp. 49–53, 2018, doi: 10.11648/j.ijdst.20180402.12.

[14] Y. Zhao, G. Kou, Y. Peng, and Y. Chen, “Understanding influence power of opinion leaders in e-commerce networks: An opinion dynamics theory perspective,” *Inf. Sci. (Ny).*, vol. 426, pp. 131–147, 2018, doi: 10.1016/j.ins.2017.10.031.

[15] P. D. I. Hernando, “Electronic commerce,” *Comput. Law Secur. Rev.*, vol. 19, no. 5, pp. 363–374, Sep. 2003, doi: 10.1016/S0267-3649(03)00504-1.

[16] L. Labuschagne and J. Eloff, “E-commerce strategy formulation.”

[17] Y. Guo, M. Wang, and X. Li, “Application of an improved Apriori algorithm in a mobile e-commerce recommendation system,” *Ind. Manag. Data Syst.*, 2017, doi: 10.1108/IMDS-03-2016-0094.

[18] P. Mehta *et al.*, “A high-bias , low-variance introduction to Machine Learning for physicists,” *Phys. Rep.*, vol. 810, pp. 1–124, 2019, doi: 10.1016/j.physrep.2019.03.001.

[19] X. M. T. Vu *et al.*, “A Shared Vision for Machine Learning in Neuroscience,” *J. Neurosci.*, vol. 38, no. 7, pp. 1601–1607, 2018, doi: 10.1523/JNEUROSCI.0508-17.2018.

[20] F. Hutter, *Automated Machine Learning*. .

[21] M. Nilashi, O. Ibrahim, and K. Bagherifard, “A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques,” *Expert Syst. Appl.*, vol. 92, pp. 507–520, 2018, doi: 10.1016/j.eswa.2017.09.058.

[22] P. Alencar and D. Cowan, “The use of machine learning algorithms in recommender systems : A systematic review,” *Expert Syst. Appl.*, vol. 97, pp. 205–227, 2018, doi: 10.1016/j.eswa.2017.12.020.

[23] T. Cunha, C. Soares, and A. C. P. L. F. De Carvalho, “Metalearning and Recommender Systems : A literature review and empirical study on the algorithm selection problem for Collaborative Filtering,” *Inf. Sci. (Ny).*, vol. 423, pp. 128–144, 2018, doi: 10.1016/j.ins.2017.09.050.

[24] S. Sivapalan, A. Sadeghian, H. Rahnama, and A. M. Madni, “Recommender systems in e-commerce,” *World Autom. Congr. Proc.*, pp. 179–184, 2014, doi: 10.1109/WAC.2014.6935763.

[25] C. S. Echt and C. D. Nelson, “Linkage mapping and genome length in eastern white pine (Pinus strobus L.),” *Theor. Appl. Genet.*, vol. 94, no. 8, pp. 1031–1037, 1997, doi: 10.1007/s001220050511.

[26] K. Zhang, R. P. Lively, C. Zhang, W. J. Koros, and R. R. Chance, “Investigating the Intrinsic Ethanol/Water Separation Capability of ZIF-8: An Adsorption and Diffusion Study,” *J. Phys. Chem. C*, vol. 117, no. 14, pp. 7214–7225, Apr. 2013, doi: 10.1021/jp401548b.

[27] R. Bell, Y. Koren, and C. Volinsky, “Matrix Factorization Techniques for Recommender Systems,” *Computer (Long. Beach. Calif).*, vol. 42, no. 08, pp. 30–37, 2009, doi: 10.1109/MC.2009.263.

[28] C. K. Pereira, F. Campos, V. Ströele, J. M. N. David, and R. Braga, “BROAD-RSI – educational recommender system using social networks interactions and linked data,” *J. Internet Serv. Appl.*, vol. 9, no. 1, 2018, doi: 10.1186/s13174-018-0076-5.

[29] D. Agalya and V. Subramaniyaswamy, “Group-Aware Recommendation using Random Forest Classification for Sparsity Problem,” *Indian J. Sci. Technol.*, vol. 9, no. 48, pp. 1–10, 2016, doi: 10.17485/ijst/2016/v9i48/107960.