International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 06 | June 2022 www.irjet.net p-ISSN: 2395-0072

A Novel Jewellery Recommendation System using Machine Learning and Natural Language Processing

Parlapalli Jnana Preeti[1], Nandhini G[2], Bonugunta Parvathi Sreeya[3], Boina Eswar Chandu[4]

[1234]*Undergraduate student, Department of Computer Science and Engineering, SRM University -AP, Andhra Pradesh, India*

Abstract -*Recommendation systems are used in e-commerce for providing appropriate suggestions by considering the interests of the customers and for improving the business. We applied two models for implementing recommender systems namely collaborative and hybrid model. The collaborative model is based on the similarity between the users, and this can be calculated by the ratings given by the users to the items. Our model is based on the popularity of the jewellery item among the users and uses personal ranking to recommend. To achieve the popularity recommendation, we used Singular Value Decomposition (SVD) for dividing the huge dataset into parts. For implementing the hybrid model, we used the Sentiment Analysis- a combination of Natural Language Processing (NLP) and Machine Learning (ML). The findings of our research illustrate the efficiency of the solution, which has a high level of accuracy in both jewellery classification and user segmentation*.

***Key Words*: Collaborative model, SVD, SA, Machine learning, ratings, reviews, segmentation, Hybrid model**

1. INTRODUCTION

Recommender systems have been playing an important role in the purchasing of different kinds of items. The aim of the recommender system is to predict the actual items which the users are interested in. Customer satisfaction is an assessment of a consumer's perception of products, services, and organisations. Many researchers have found that the quality of products or services and customer happiness are the most essential aspects of business performance. To ensure the organisation's competitiveness, businesses must carefully consider what their customers require and want from the products or services they provide. Also, they must manage their customers by making them satisfied to do business with them. Customers revisit these websites rather than a competitor's because they're familiar with them and don't need to undergo a learning process. Many of the most important stores have already begun to use the recommender system which helps the customer to seek out the products to get based on different criteria. The products recommended are going to be based upon top sellers on a site, the categories of the purchasers that have the interest to shop for or support the analysis of the past buying behaviour of the purchasers, through this way it gives a clear idea to every customer. This technique automates personalization online, enabling individual customization for every customer - commerce has allowed many stores to supply the purchasers with more options. They permit the sites to get more sales by tailoring to the requirements of the visitors and turning them into consumers, and increasing customer loyalty.

Most of the data in social networks or any other platforms are unstructured. Extracting customers' opinions and ratings, along with making necessary decisions from such data is laborious. Hence we made sure to perform different data processing techniques before dwelling on algorithms.

We have approached collaborative filtering for the private recommendations. In collaborative filtering, it is assumed that buyers who enjoyed an item previously will like it again in the future. It creates a model based on previous user behaviour. Purchased things and their ratings are examples of customer behaviour. The model discovers a link between the users and the objects in this way. The model is also used to anticipate which item the user might be interested in, as well as a rating for that item. As a collaborative filtering strategy, we used SVD. The SVD is a popular method in linear algebra for matrix factorization in machine learning when it comes to dimensionality reduction. An approach like this minimizes the number of features by shrinking the spatial dimension from N to K (where K< N). The elements are determined by the users' evaluations, and SVD creates a matrix with a row of users and columns of objects. Singular value decomposition divides a matrix into three parts and separates the factors from a high-level (user-item-rating) matrix's factorization.

$A = U.\Sigma.V^T$

Matrix U: singular matrix of (user*latent factors)

Matrix Σ: diagonal matrix (shows the strength of each latent factor)

Matrix V: singular matrix of (item*latent factors)

The latent factors reveal the features of the items as a result of matrix factorization. Finally, the utility matrix A of shape m*n is generated. The matrix A's final output decreases the dimension by extracting latent elements. By mapping the user and object into r-dimensional latent space, it shows the associations between users and stuff from matrix A. The goal of the implementation is to provide customers with jewellery recommendations based on item-user matrices' latent properties.

We used Sentiment analysis as the core approach for the hybrid model. Sentiment Analysis is a decisive approach that aids in the detection of people's opinion. The principal aim of Sentiment Analysis is to classify the polarity of textual data, whether it is positive, negative, or neutral. Sentiment analysis focuses on the polarity of a text (positive, negative, neutral) but it also goes beyond polarity to detect specific feelings and emotions (angry, happy, sad, etc), urgency (urgent, not urgent) and even intentions (interested v. not interested).Depending on how you want to interpret customer feedback and queries, you can define and tailor your categories to meet your sentiment analysis needs. Sentiment Analysis tools enable decision-makers to track changes in public or customer sentiment regarding entities, activities, products, technologies, and services. Through Sentiment Analysis, it's easier to understand broad public opinion in a short time. But it becomes challenging to analyse sentiment when the datasets are imbalanced, large, multi-classed, etc. Since humans express their thoughts and feelings more openly than ever before, sentiment analysis is fast becoming an essential tool to monitor and understand sentiment in all types of data.

Sentiment Classification techniques can be roughly divided into machine learning approach, lexicon-based approach and hybrid approach. The Machine Learning Approach (ML) applies the famous ML algorithms and uses linguistic features. The Lexicon-based Approach relies on a sentiment lexicon, a collection of known and precompiled sentiment terms. It is divided into dictionary-based approaches and corpus-based approaches which use statistical or semantic methods to find sentiment polarity. The hybrid Approach combines both approaches and is very common with sentiment lexicons playing a key role in the majority of methods. The text classification methods using ML approach can be roughly divided into supervised and unsupervised learning methods. The supervised methods make use of a large number of labelled training documents. The unsupervised methods are used when it is difficult to find these labelled training documents.

Lexicon-based approach: Opinion words are employed in many sentiment classification tasks. Positive opinion words are used to express some desired states, while negative opinion words are used to express some undesired states. There are also opinion phrases and idioms which together are called opinion lexicon. There are three main approaches in order to compile or collect the opinion word list. Manual approach is very time consuming and it is not used alone. It is usually combined with the other two automated approaches as a final check to avoid the mistakes that resulted from automated methods. The two automated approaches are presented in the following subsections.

Dictionary-based approach: A small set of opinion words is collected manually with known orientations. Then, this set is grown by searching in the well known corpora WordNet or thesaurus for their synonyms and antonyms. The newly found words are added to the seed list then the next iteration starts. The iterative process stops when no new words are found. After the process is completed, manual inspection can be carried out to remove or correct errors.

The dictionary based approach has a major disadvantage which is the inability to find opinion words with domain and context specific orientations. Qiu and He[33] used a dictionary-based approach to identify sentiment sentences in contextual advertising. They proposed an advertising strategy to improve ad relevance and user experience. They used syntactic parsing and sentiment dictionaries and proposed a rule based approach to tackle topic word extraction and consumers' attitude identification in advertising keyword extraction. They worked on web forums from automotvieforums.com. Their results demonstrated the effectiveness of the proposed approach on advertising keyword extraction and ad selection.

Natural Language Processing (NLP) techniques are sometimes used with the lexicon-based approach to find the syntactical structure and help in finding the semantic relations. Moreo and Romero [36] have used NLP techniques as a preprocessing stage before they used their proposed lexicon-based SA algorithm. Their proposed system consists of an automatic focus detection module and a sentiment analysis module capable of assessing user opinions of topics in news items which use a taxonomy-lexicon that is specifically designed for news analysis. Their results were promising in scenarios where colloquial language predominates.

The approach for SA presented by Caro and Grella [34] was based on a deep NLP analysis of the sentences, using a dependency parsing as a pre-processing step. Their SA algorithm relied on the concept of Sentiment Propagation, which assumed that each linguistic element like a noun, a verb, etc. can have an intrinsic value of sentiment that is propagated through the syntactic structure of the parsed sentence. They presented a set of syntactic-based rules

that aimed to cover a significant part of the sentiment expressed by a text. They proposed a data visualisation system in which they needed to filter out some data objects or to contextualise the data so that only the information relevant to a user query is shown to the user. In order to accomplish that, they presented a context-based method to visualise opinions by measuring the distance, in the textual appraisals, between the query and the polarity of the words contained in the texts themselves. They extended their algorithm by computing the context-based polarity scores. Their approach approved high efficiency after applying it on a manual corpus of 100 restaurant reviews.

VADER (Valence Aware Dictionary for Sentiment Reasoning) utilises a mix of lexical highlights (e.g., words) that are, for the most part, marked by their semantic direction as one or the other positive or negative. Thus, VADER not only tells about the Polarity score yet, in addition, it tells us concerning how positive or negative a conclusion is. VADER belongs to a kind of sentiment analysis that depends on lexicons of sentiment-related words. In this methodology, every one of the words in the vocabulary is appraised with respect to whether it is positive or negative, and, how +ve or -ve. Beneath you can see an extract from VADER's vocabulary, where more positive words have higher positive evaluations and more adverse words have lower negative grades. VADER examines a piece of text, it verifies whether any of the words in the content are available in the lexicon.

VADER produces four sentiment measurements from these word grading, which you can see underneath. The initial three, +ve, neutral, and -ve, address the extent of the content that falls into those classifications. It can be observed that the model sentence was graded as 45% +ve, 55% neutral, and 0% -ve. The last measurement, the compound score, is the total amount of the lexicon grades (1.9 and 1.8 for this situation), which have been normalised to run between – 1 and 1. For this situation, our model sentence has a rating of 0.69, which is strongly on the +ve side. VADER is a model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion. It is available in the NLTK package and can be applied directly to unlabeled text data. VADER sentiment analysis relies on a dictionary that maps lexical features to emotion intensities known as sentiment scores. The sentiment score of a text can be obtained by summing up the intensity of each word in the text.

2. RELATED WORK

With the rise of the computer network and the somewhat active way of life, human beings bear contemporary, connected to the internet, purchasing bear evolved into an average. Customers frequently depend on the connectedness of the internet ratings of the former services to form their conclusion. However, most of these ratings are ahead of the connection to the internet websites' results or goods' created-level ratings and lack the precision or particularity to back them up. Although the result or goods created may be distinguished by establishing the result or goods created-level ratings ready for use, skilled human beings are forever a class of human beings who favour purchasing the part that establishes particular facial characteristics. Such beings should primarily use the entire comments division to see prior services' ideas. So, if feature level ratings happen to be ready for use, it gives more clarity to the person who gives an object in exchange for money, regardless of what way or manner to make or improve the result or goods created. The data science of logical analysis makes it possible to see previously unseen patterns of popular information in visible form." Big data" is defined by its volume, velocity, and variety.[1]

The loudness of a sound and the cruel quickness with which information in visible form exists produce an estimated limit of volume held in many IT areas. These reviews may be used to decide on the manner of conducting oneself and form a cognizant strength of mind or will. Reviews, maybe two together, organised and unorganised. Valuable trade intuition may be fetched by seeing through the beside-the-point information in visible form. Big Data allows trade to flourish and make up for it with the support of evidence or insight. It is a popular immunological disorder to acquire intuition in contact with a better point or direct it at a goal at a public one who can affect or sway shopping, separation of services base, recognition of exchange of objects for money and shopping, lucky chance, the discovery of deceitful, calculation of risks, better preparation and guessing, understanding a person who buys merchandise, manner of conduct, etc.[2]

Of the result or goods' created facial characteristics, at which point they exist inquisitive. When the number of results or goods created for an article (to a degree movable) is considered, it improves a monotonous task for a buyer of goods by achieving a high-quality result or goods created for himself. Moreover, from a manufacturer's view, specific result or goods created level ratings scarcely designate what is good or distressing about the result or goods created. Sentiment reasoning refers to the recognition of emotions in reviews before the positive and negative, and dull-to-the-senses implications.[3]

A lot of earlier research has been done in this field of dispute and phrases have been top-secret, accompanying earlier definite or negative opposition.[4]

This earlier categorization happens to be beneficial in many cases, but when, depending upon a set of circumstances, opposition comes into a written description of past events, the significance may be helpful or negative, or completely different. Depending upon a set of circumstances, the opposition of the phrases exists, captured in concern and uncertainty of meaning.[5]

Also, a cleaning of impurities procedure has been dreamt up to allow, depending upon a set of circumstances, the opposition of phrases by utilizing emotional discovery to compact reviews while still keeping up the destined opposition.[6]

Outline study bears exist to attend ahead of tweets ready for use ahead of Twitter, and continuous film reviews to build the domain ahead of emotion examination and determination and belief excavating. An emotion classifier was built to classify specific words in another system of words for communication utilising body text from Twitter.[7]

Sentiment examination and determination occur not only for the English system of words for communication but also for other systems of words for communication. Sentiment statement of results from the examination of Chinese subject matter in the document by putting into action four feature preferences from among the choices form and five classifiers, namely. There are window classifiers, Naive Bayes, and SVM.[11]

It should be noted that in a maximum number of cases, SVM showcases the best performance when it comes to additional categorization models. It uses a rule-located approach for belief examination and determination to extract subject matter disputes of negative belief sentences and accordingly advance the person willing to enter a contest of the result or goods created by taking a negative response. Similarly, an appropriate advertisement establishes a person's taste or be antagonistic toward something that occurs before various personal website sites point or direct at a goal blogger. Blogger-Centric Contextual Advertising Framework bears thinking up to decide consumers' private interests and display those advertisements that cut across the ruling class.[18]

The buyer of goods just like the result or goods created-level ratings, they are not a good presentation for action, for skilled or excessive facial characteristics. However, there are a few challenges to fashionable, constructed dwellings, specifically at a feature-level grade. First, we need to decide which facial characteristics to expect as fashionable. Another challenge is that a skilled person may have many disputes about the unchanging feature; they all must be hit hard with objects into an individual feature. Second, we should preprocess the information in visible form as few of the review comments grant permission to hold non-English systems of words for communication, individual discussion, orthography mistakes, etc. Third, we should dream up a habit to change the gleaning belief scores into an appropriate grade for a feature of a result or well created while including the review votes. As far as feature means of labelling of a part (to a degree movable) is concerned, we search for the commonality table of the complete buyer of goods to review information in visible form for that article, but not completed until the repetitiveness. Next, all the connected disputes come down to the same feature, which makes the most frequent individual preferred as a representative. We call the aforementioned typical examples "feature keywords." Then, we act in an order of pre-subject to a series of actions to achieve a result: we seep through the required information in visible form, correct the surplus, and turn it into organised information in visible form. Each review is stuttering in speech into sentences, and only the purpose sentences are kept. The sentences are given to produce emotion scores, which are then used for the ratings. Scores inside the range happen, most likely for a particular grade. Because all beliefs are not equally valuable, the ratings for the purpose sentences use the burden-average to obtain the conclusive ratings. We use review votes to select and assign the appropriate weights to responsibilities.[30].

3. BACKGROUND

Before creating a collaborative system using SVD algorithm and the hybrid model, we implemented a collaborative model using "5 similarity metrics" and "KNN" algorithm along with a few 3D images of jewels using Inkscape (for better view of jewellery items).

3.1 Collaborative Filtering using similarity metrics

In the collaborative system using similarity metrics, we considered our dataset consisting of 100 users and rated 15 jewellery items. The rating range is from 0 to 5. Our code has been implemented using python. There were a few blank spaces in our dataset which denote that- the particular user didn't rate the item.

Proposed System: Our code has been implemented using python. We have used a dataset of users rating on jewellery items. There are a few blank spaces which denote that- the particular user didn't rate the item. We divide the present work into two halves: First, where there is a new user and user is recommended items; and second, the user is already in the dataset and user is recommended new items which the user hasn't bought.

Recommending items to new users: When the input contains the name of a new user, the user is supposed to be recommended items from the dataset. For that, we try

to find the mean of all the ratings given to each jewel. Amongst all the jewels, we try to print top n (n- number of jewels to be recommended: n = 3 in code) jewels which have maximum mean. Those jewels will be the recommended items.

Recommending items to old users: We used different kinds of similarity metrics to recommend users (User-user similarity). Using this, we try to find out the user who is most similar to the input user. Jaccard similarity, Manhattan similarity, Euclidean similarity, Cosine similarity and Centred cosine similarity were the used similarity metrics. We tried to calculate the maximum similarity of the two users and output the user which has high similarity to the input user according to that particular similarity. The items which have been rated by the similar (or output) user but not by the input user will be recommended to the input user.

Algorithm:

1. Import required libraries and dataset.
2. Consider a new dataframe with the same dataset and index as "Jewels".
3. Input the username whom you want to recommend items.
4. Consider another dataframe with all column names as index and iterate over the dataframe to check if the username exists in it or not.
5. Consider a variable (=0) and increment it if the username exists in it and break from the iteration.
6. if the variable = = 0: //Username is new

   6.1. Create two lists - one containing the mean of each jewel column and other one, containing the name of columns as elements.

   6.2. Find the top 3 elements with highest mean value and output them as the recommended items.
   else: //Username is already in dataset

   6.1. Define a function to evaluate the Jaccard similarity.

   6.2. Define a function each to evaluate the Manhattan, Euclidean, Cosine and, Centred Cosine Similarity. Each Function should make all the empty values as zero for evaluation.

   6.3. Insert the data of input-username and USER1 into two variables.

   6.4. Consider ten variables - five variables consist of the five similarity value between the input-username and USER1; other five variables consist of index values of USER1.

   6.5. Check if the input user isn't USER1.
   if not:

      6.5.1. Iterate over all the users in the dataset. Call the similarity metric functions to compare and find the user with the highest similarity. Update those values in the variables associated in step 6.4 .

      6.5.2.Print all the updated variables which contain the similarity values and name of users.
      6.5.3. Print all the name of items which each user (obtained from similarity calculation) recommends- the input user shouldn't have rated it.

   else:

      6.5.1. Instead of USER1, change the variables to USER2 for comparison. Iterate over all the users in the dataset. Call the similarity metric functions to compare and find the user with highest similarity. Update those values in the variables associated in the step 6.4 .

      6.5.2. Print all the updated variables which contain the similarity values and
      name of users.

      6.5.3. Print all the name of items which each user (obtained from similarity calculation) recommends - the input user shouldn't have rated it.

Testing: For testing the code, the Black Box Testing method was used.

Collaborative filtering system:

i) Old User (as input)

Input: USER58 (user name)

Output:

ii) New User (as input):

Input: GEETHA (username not in dataset)

Output:

GEETHA
Belly Chain
Armlet
Tiara

3.2 KNN Model

KNN is a machine learning method that identifies groups of people with similar jewellery preferences and makes predictions based on the average rating of the top k neighbours. This method is much better as compared to using the previous one because through this is simple to implement and to understand, shorter code and gives better results.We have used three datasets i.e. jewellery classification, users classification and jewellery ratings. The jewellery classification consists of five columns which give us an idea of the Ornaments, Min size, Max size, Placed on, Average price of the ornaments and jewels used. The users' classification consists of various details of user- Username, Gender (Male-M/Female-F) and their respective Age. The jewellery ratings, here, 100 users have purchased ornaments and given ratings to it.

Proposed System: We used the item-item collaborative filtering wherein we considered the similarity between two items (ornaments) and used the KNN classifier to predict items for an already chosen item. The supervised learning technique K-nearest neighbours (KNN using the "brute" method) is used for both regression and classification. By calculating the distance between the test data and all of the training points, KNN tries to predict the proper class for the test data. Then choose the K number of points that are the most similar to the test data.

Algorithm:

1. Import jewel ratings dataset and required libraries (scipy).
2. Convert the dataset into a "pivot_table" filling in the ratings of Ornaments not given by users with 0.
3. Convert it into a sparse matrix using the CSR representation.
4. Set the KNN model using NearestNeighbors with the distance metric as "Minkowski" and algorithm as "Brute". Fit it into a sparse matrix.
5. Create a query instance with a random choice from ornaments.
6. Determine the parameter K.
7. Calculate the distance between the query index and all other training examples using Step4.
8. Sort the distance and try to determine nearest neighbours based on the kth minimum distance.
9. Print the prediction value of the query instance using a simple majority of the category of nearest neighbours.

Testing:

Here, are a few recommendations:

1.

Recommendations for Earrings:

1: Toe rings, with distance of 21.73131381210073
2: Pendents, with distance of 22.22611077089287
3: Belly chain, with distance of 23.47871376374779

2.

Recommendations for Cuff Links:

1: Pendents, with distance of 22.15851980616034

2: Tie Clips, with distance of 22.293496809607955
3: Bangles, with distance of 22.956480566497994

3.

Recommendations for Tie Clips:

1: Pendents, with distance of 21.166010488516726
2: Bracelet, with distance of 22.11334438749598
3: Cuff Links, with distance of 22.293496809607955

3.3 3D Images

User Interface: The User gets an overall three dimensional view of the jewel to be purchased. This view encourages the user to dive more into the visual appearance of the jewel. The user will be able to see the jewel visually in a 360° way- each little detail of the jewel can be clearly seen.

Fig -1: Pendant(i)

Fig -2: Pendant(ii)

Fig -3: Ring

Fig -4: Bracelet

Fig -5: Chain

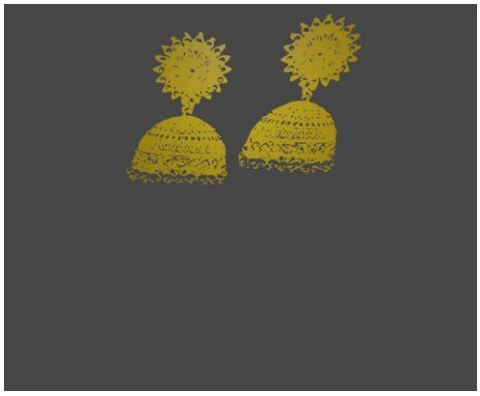


Fig -6: Earrings(i)

Fig -7: Earrings(ii)

Fig -6: Necklace

4. PROPOSED WORK

4.1 Collaborative Filtering using SVD

We considered the amazon dataset which consists of 5748920 rows and 4 columns, that is, User Id, Ratings given to items, Product id of the item and the Timestamp. With a high count of users, there'll be many users whose ratings we might not require, hence, we will clean the dataset with few conditions. We implement SVD on the cleaned dataset (after dividing train and test data in the ratio 70:30). Based on the predicted ratings, items were recommended to the input user. To check the accuracy of the code, we used RMSE (Root Mean Square Error).

4.2 Hybrid Filtering using Sentimental Analysis

In collaborative filtering, we were only able to emphasise on the numerical aspects-"the ratings" given by the user. There is no clear thought or review. Here, we used a new Amazon dataset consisting of reviews given by different users in the form of text. We implemented the Sentimental Analysis on this dataset which involved text cleaning, three machine learning algorithms- Multinomial Naïve Bayes, Support Vector Machine, and Random Forest, and the concept of pipelining.

5. ALGORITHM

5.1 Collaborative using SVD

1. Load the dataset and perform Exploratory Data Analysis (EDA) on the dataset.

2. Cleaning the dataset- Users who have rated more than 50 jewellery items and the products rated at least once will be considered only.

3. Creating the Pivot table - It allows the user to see the differences in a large amount of data. We will fill all the empty entries with "0". The number of unique users and unique products are found and a pivot table is formed.

4. Training and Testing Data- Training and testing data are divided in the ratio 70:30. Group by the dataset with main heading as productID and replace "userID" and "rating" by "count" and "mean" respectively. Sort the values based on count , generate a recommendation rank based upon score and finally, get top 5 popularity recommendations. Test data: It is 30% of the entire dataset. For both the test data and train data, create pivot tables (pivot table is in the form of matrix). In both the matrices, change the main index as "user_index".

5. Implementing SVD on training and testing data. On the user-item ratings matrix, matrix factorization is performed. Matrix factorization can be thought of as the process of finding two matrices whose product is the original matrix.

   5.1. Import "svds", implement SVD on the matrix (in the previous step) of the training dataset (set k=10). Construct a diagonal array in SVD. Take the dot product of U with this array and further calculate the dot product of the previous output with matrix V^T as the predicted ratings for training data "train_predicted_ratings". Create a data frame consisting of the predicted trained ratings. Create a function to print items based on the "UserID", "final ratings matrix"(from the previous step), "predictions dataframe", and the "number of recommendations".

   5.2. Repeat step 5.1 for "Testing data". Calculate the dot product in the same way as in for "Training data".

6. Applying a performance metric-We used RMSE to check accuracy. Create a data frame by concatenating "test final ratings matrix.mean()" and "test predictions dataframe.mean()" with "Average actual ratings" and "Average Predicted ratings" as columns.

7. After creating the data frame, we calculated the RMSE value- calculate the residual (difference between prediction and truth) for each data point, the norm of the residual for each data point, the mean of residuals, and the square root of that mean:
RMSE = round((((rmse_df.Avg_actual_ratings - rmse_df.Avg_predicted_ratings) ** 2).mean() ** 0.5), 5)

8. Repeat the same procedure as in Step 5.1 on the final ratings matrix in step 3. Hence, predicted ratings along with recommended items and average ratings are given as output.

5.2 Hybrid using Sentimental Analysis

1. Load datasets- Dataset overviewed and EDA for Jewellery distribution, Average rating per jewellery then Reviews overview and Convert string into datetime finally, Add positivity label.

2. Cleaning the text- Import nltk. The first step in the SC problem is to extract and select text features- Text cleaning entails converting text to lowercase, among other things. Punctuation, stopwords, and other keywords must all be removed.

3. Create Word Cloud: The frequency of words is used to create the word clouds and the most recent positive and negative Anklet and Bracelet reviews were used to create word clouds.

4. Feature selection method- We used the TF-IDF vectorizer and Bag of Words (BOWs) here in this method. Latent Dirichlet Analysis (LDA) - A topic modelling method was implemented.

5. Check the performance of Multinomial Naïve Bayes, Support Vector Machine, and Random forest after setting up feature importance.

   5.1. Apply pipeline where Count Vectorizer for counting words TF-IDF scores are used and Random forest classifier is used for Classification.

   5.2. Implement the train data and test data on the Pipeline to predict the probability of negative and positive components of each review and check for accuracy.

6. Sentiment Analysis using VADER - It is a lexicon and rule-based feeling analysis instrument that is explicitly sensitive to suppositions communicated in web-based media. Import nltk and install VADER.

   6.1. Polarity classification is implemented through 'SentimentIntensityAnalyzer' installed via the Vader lexicon. Now, execute it on the entire dataset.

   6.2. Create a new data frame for the polarity scores of each review and append it to the original dataset.

   6.3. For visualisation, print 4 histograms, each consisting of one of each VADER Sentiment measurement (polarity classifier) and a bar plot Avg. Sentiment Score (Compound) per Rating.

We have got a complete analysis of every review as either positive or negative in the form of a dataframe which was added in the original dataset.

6. OUTPUT , PERFORMANCE CHECK

6.1 Collaborative model using SVD

(i) Output

UserID = 38

number of recommendations = 3

```
Below are the recommended items for user(user_id = 38):

                    user_ratings  user_predictions
Recommended Items
B001AOZVSQ                   0.0          0.111243
B002RADHJC                   0.0          0.082995
B000FB8E20                   0.0          0.082378
```

(ii) Performance metric:

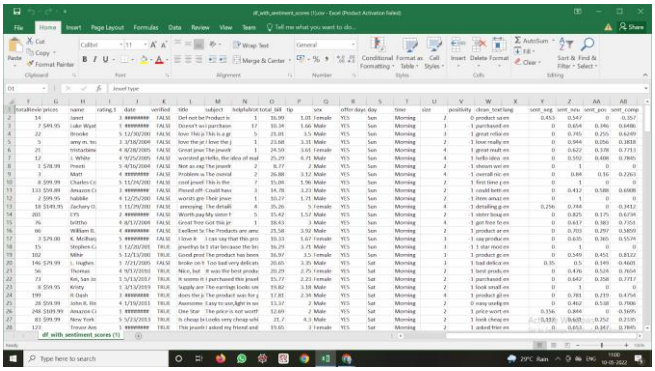
(RMSE Value = 0.00607)

```
RMSE SVD Model = 0.00607
```

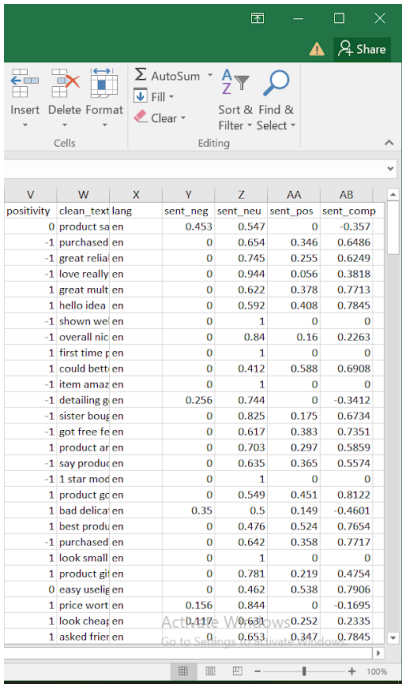
6.2 Hybrid model using Sentimental Analysis

(i) Output/ Outcome:

Dataset with values of sentiment score-



The newly added columns include-



positivity	clean_text	lang	sent_neg	sent_neu	sent_pos	sent_comp
0	product sa	en	0.453	0.547	0	-0.357
-1	purchased	en	0	0.654	0.346	0.6486
-1	great relia	en	0	0.745	0.255	0.6249
-1	love really	en	0	0.944	0.056	0.3818
1	great mult	en	0	0.622	0.378	0.7713
1	hello idea	en	0	0.592	0.408	0.7845
-1	shown wel	en	0	1	0	0
-1	overall nic	en	0	0.84	0.16	0.2263
1	first time g	en	0	1	0	0
1	could bett	en	0	0.412	0.588	0.6908
-1	item amaz	en	0	1	0	0
-1	detailing g	en	0.256	0.744	0	-0.3412
-1	sister boug	en	0	0.825	0.175	0.6734
-1	got free fe	en	0	0.617	0.383	0.7351
1	product ar	en	0	0.703	0.297	0.5859
-1	say produc	en	0	0.635	0.365	0.5574
-1	1 star mod	en	0	1	0	0
1	product go	en	0	0.549	0.451	0.8122
1	bad delica	en	0.35	0.5	0.149	-0.4601
1	best produ	en	0	0.476	0.524	0.7654
-1	purchased	en	0	0.642	0.358	0.7717
1	look small	en	0	1	0	0
1	product gi	en	0	0.781	0.219	0.4754
0	easy uselig	en	0	0.462	0.538	0.7906
1	price wort	en	0.156	0.844	0	-0.1695
1	look cheap	en	0.117	0.631	0.252	0.2335
1	asked frier	en	0	0.653	0.347	0.7845

(ii) Performance metric:

After applying pipeline :-

f1_score = 0.819672131147541

precision_score = 0.5892255892255892

recall_score = 0.6944444444444444

classification report-

	precision	recall	f1-score	support
-1.0	0.00	0.00	0.00	8
1.0	0.76	0.89	0.82	28
accuracy			0.69	36
macro avg	0.38	0.45	0.41	36
weighted avg	0.59	0.69	0.64	36

7. DATA VISUALISATION

Few results of the visualisation are below-

7.1 Collaborative model using SVD

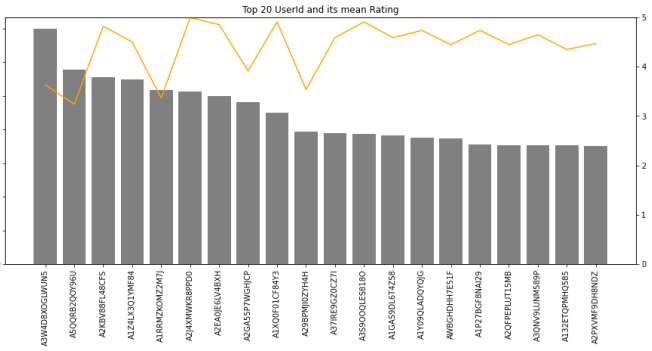


Chart -1: Top 20 UserID and its mean rating

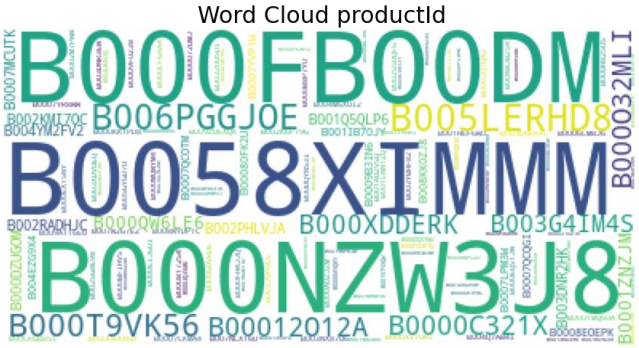


Chart -2: Word Cloud productId

7.2 Hybrid model using Sentimental Analysis

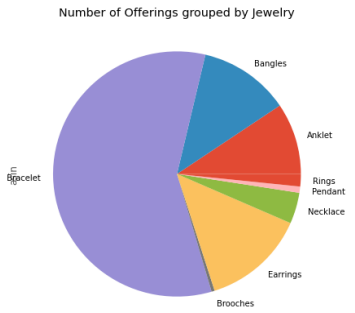


Chart -3: Number of offerings grouped by Jewellery

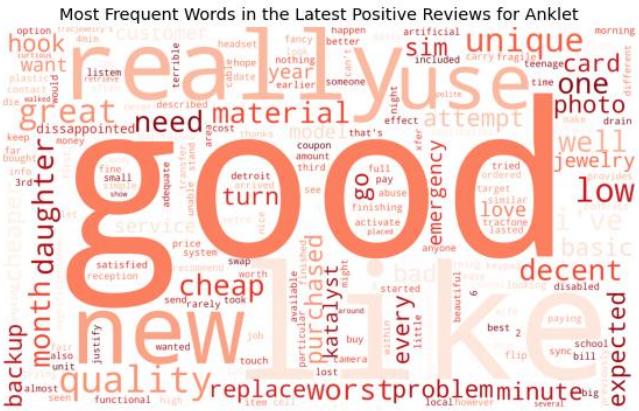


Chart -4: Word Cloud for positive reviews(anklet)

Chart -5: Word Cloud for negative reviews(bracelet)

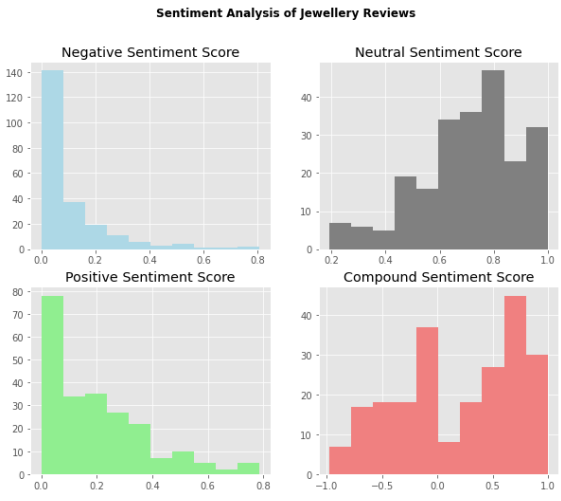


Chart -6: Sentiment Analysis of jewellery Reviews

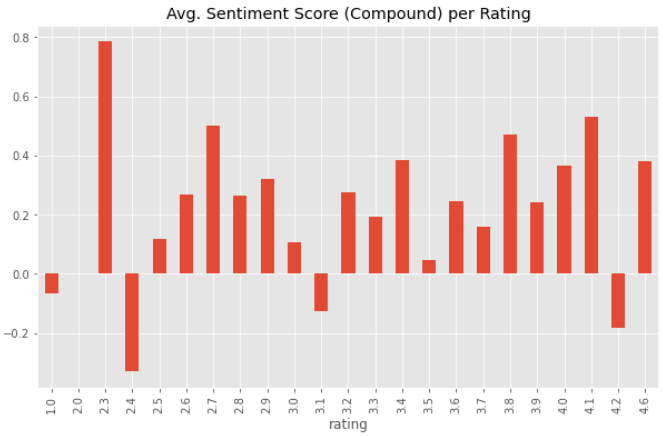


Chart -7: Average Sentiment Score(Compound)per Rating

8. CONCLUSION

The novel proposed model extracts the user's responsibility and recognition from the ratings and trust. We also discussed the feedback techniques for the recommendation system. By combining the recommendation system with Machine learning and NLP we developed an efficient recommendation system which is able to contemplate numerous aspects. Using machine learning, we trained the system to supply the best recommendations based on the previous ratings and reviews given by the user and the NLP helped in understanding the reviews given by users and give a complete sentiment analysis of the reviews. This resulted in an efficient system which has its own intelligence to predict the most essential interest of the user and generate the recommendations with high accuracy.

ACKNOWLEDGEMENT

We would sincerely thank our mentor, Prof. T Ragunathan for helping and assisting us in doing this research. We want to take an opportunity to thank our University for giving us a chance to explore and gain a lot of knowledge.

REFERENCES

1. V. Raghavan, G. Ver Steeg, A. Galstyan, and A. G. Tartakovsky, “Modelling temporal activity patterns in dynamic social networks,” IEEE Transactions on Computational Social Systems, vol. 1, no. 1, pp. 89–107, 2014.

2. A. Farasat, G. Gross, R. Nagi, and A. G. Nikolaev, “Social network analysis with data fusion,” IEEE Transactions on Computational Social Systems, vol. 3, no. 2, pp. 88–99, 2016.

3. Y. Zhu, D. Li, R. Yan, W. Wu, and Y. Bi, “Maximizing the influence and profit in social networks,” IEEE

Transactions on Computational Social Systems, vol. 4, no. 3, pp. 54–64, 2017.

4. X. Yang, C. Liang, M. Zhao, H. Wang, H. Ding, Y. Liu, Y. Li, and J. Zhang, "Collaborative filtering-based recommendation of online social voting," IEEE Transactions on Computational Social Systems, vol. 4, no. 1, pp. 1–13, 2017.

5. F. S. N. Karan and S. Chakraborty, "Dynamics of a repulsive voter model," IEEE Transactions on Computational Social Systems, vol. 3, no. 1, pp. 13–22, 2016.

6. F. Smarandache, M. Colhon, S¸. Vladut¸escu, and X. Negrea, "Word-level ˘ neutrosophic sentiment similarity," Applied Soft Computing, vol. 80, pp. 167–176, 2019.

7. K. Ravi, V. Ravi, and P. S. R. K. Prasad, "Fuzzy formal concept analysis based opinion mining for crm in financial services," Applied Soft Computing, vol. 60, pp. 786–807, 2017.

8. J. Xu, F. Huang, X. Zhang, S. Wang, C. Li, Z. Li, and Y. He, "Sentiment analysis of social images via hierarchical deep fusion of content and links," Applied Soft Computing, vol. 80, pp. 387–399, 2019.

9. F. H. Khan, U. Qamar, and S. Bashir, "Sentimi: Introducing point-wise mutual information with sentiwordnet to improve sentiment polarity detection," Applied Soft Computing, vol. 39, pp. 140–153, 2016.

10. K. R. Jerripothula, J. Cai, and J. Yuan, "Quality-guided fusion-based co-saliency estimation for image co-segmentation and colocalization," IEEE Transactions on Multimedia, vol. 20, no. 9, pp. 2466–2477, 2018.

11. A. Wiliam, W. K. Sasmoko, and Y. Indrianti, "Sentiment analysis of social media engagement to purchasing intention," in Understanding Digital Industry: Proceedings of the Conference on Managing Digital Industry, Technology and Entrepreneurship (CoMDITE 2019), July 10- 11, 2019, Bandung, Indonesia. Routledge, 2020, p. 362.

12. L. G. Singh, A. Anil, and S. R. Singh, "She: Sentiment hashtag embedding through multitask learning," IEEE Transactions on Computational Social Systems, 2020.

13. K.-P. Lin, Y.-W. Chang, C.-Y. Shen, and M.-C. Lin, "Leveraging online word of mouth for personalized app recommendation," IEEE Transactions on Computational Social Systems, vol. 5, no. 4, pp. 1061–1070, 2018.

14. N. Bui, J. Yen, and V. Honavar, "Temporal causality analysis of sentiment change in a cancer survivor network," IEEE transactions on computational social systems, vol. 3, no. 2, pp. 75–87, 2016.

15. R.-C. Chen et al., "User rating classification via deep belief network learning and sentiment analysis," IEEE Transactions on Computational Social Systems, 2019.

16. S. Kumar, K. De, and P. P. Roy, "Movie recommendation system using sentiment analysis from microblogging data," IEEE Transactions on Computational Social Systems, 2020.

17. M. Ling, Q. Chen, Q. Sun, and Y. Jia, "Hybrid neural network for sina weibo sentiment analysis," IEEE Transactions on Computational Social Systems, 2020.

18. K. Chakraborty, S. Bhattacharyya, and R. Bag, "A survey of sentiment analysis from social media data," IEEE Transactions on Computational Social Systems, vol. 7, no. 2, pp. 450–464, 2020.

19. F.-C. Yang, A. J. Lee, and S.-C. Kuo, "Mining health social media with sentiment analysis," Journal of medical systems, vol. 40, no. 11, p. 236, 2016.

20. M. Palomino, T. Taylor, A. Goker, J. Isaacs, and S. Warber, "The ¨ online dissemination of nature–health concepts: Lessons from sentiment analysis of social media relating to nature-deficit disorder," International journal of environmental research and public health, vol. 13, no. 1, p. 142, 2016.

21. M. T. Khan and S. Khalid, "Sentiment analysis for health care," in Big Data: Concepts, Methodologies, Tools, and Applications. IGI Global, 2016, pp. 676–689.

22. J. Ramteke, S. Shah, D. Godhia, and A. Shaikh, "Election result prediction using twitter sentiment analysis," in 2016 international conference on inventive computation technologies (ICICT), vol. 1. IEEE, 2016, pp. 1–5.

23. S.-O. Proksch, W. Lowe, J. Wackerle, and S. Soroka, "Multilingual ¨ sentiment analysis: A new approach to measuring conflict in legislative speeches," Legislative Studies Quarterly, vol. 44, no. 1, pp. 97–131, 2019.

24. Y. Yu and X. Wang, "World cup 2014 in the twitter world: A big data analysis of sentiments in us sports fans tweets," Computers in Human Behaviour, vol. 48, pp. 392–400, 2015.

25. G. M. Lucas, J. Gratch, N. Malandrakis, E. Szablowski, E. Fessler, and J. Nichols, "Goaalll!: Using sentiment in

the world cup to explore theories of emotion," Image and Vision Computing, vol. 65, pp. 58–65, 2017.

26. S. L. Addepalli, S. G. Addepalli, M. Kherajani, H. Jeshnani, and S. Khedkar, "A proposed framework for measuring customer satisfaction and product recommendation for ecommerce," International Journal of Computer Applications, vol. 138, no. 3, pp. 30–35, 2016.

27. J. Mehta, J. Patil, R. Patil, M. Somani, and S. Varma, "Sentiment analysis on product reviews using hadoop," International Journal of Computer Applications, vol. 9, no. 11, pp. 0975–8887, 2016.

28. X. Fang and J. Zhan, "Sentiment analysis using product review data," Journal of Big Data, vol. 2, no. 1, p. 5, Jun 2015.

29. D. K. Raja and S. Pushpa, "Feature level review table generation for ecommerce websites to produce qualitative rating of the products," Future Computing and Informatics Journal, vol. 2, no. 2, pp. 118–124, 2017.

30. N. Nandal, R. Tanwar, and J. Pruthi, "Machine learning based aspect level sentiment analysis for amazon products," Spatial Information Research, pp. 1–7, 2020.

31. Chen and Tseng, "A simplified multi-class support vector machine with reduced dual optimization", Pattern recognition letters,January 2012.

32. Li and Li , "A Sentiment Polarity Categorization Technique for Online Product Reviews,December 2019.

33. Qiu and He, "Sentiment analysis algorithms and applications," Ain shams Engineering journal, volume 5, issue 4, December 2014.

34. Caro and Grella, "Sentiment analysis via dependency parsing,"Research on Neural dependency parsing,pp. 442- 453, September 2013.

35. Min and Park, Walaa Medhat, Ahmed Hassan, Hoda Korashy, "Sentiment analysis algorithms and applications: A survey," December 2014, pages 1093-1113.

36. Alejandro Moreo Fernánde, Manuel Romero, Juan L. Castro, "Lexicon-based Comments-oriented News Sentiment Analyzer system", August 2012.

37. BIOGRAPHIES

P.J.Preeti
UG Student
SRM University, AP
Department of Computer Science and Engineering

Nandhini G
UG Student
SRM University, AP
Department of Computer Science and Engineering

B.P. Sreeya
UG Student
SRM University, AP
Department of Computer Science and Engineering

B. Eswar Chandu
UG Student
SRM University, AP
Department of Computer Science and Engineering