

CS6109 - Compiler Design

Project Documentation

Predicting and Classifying Interests of Users Based on Direct Interactions

Team - 3

Vivek Ramkumar, 2018103082

Kariketi Tharun Reddy, 2018103034

1. R. Srikanth, 2018103603

# Introduction:

The basis of our project is to provide a textual analysis of the interactions between multiple users, and predict their interests, as well as categorize them.

This is an important function of recommendation systems, that is, to offer a personalized experience for each user, by understanding their preferences.

Direct interactions between users will be subject to lexical, syntax and semantic analysis, from which we will obtain their preferences to items, as well as actions and hobbies.

## List of Modules:

We will employ the standard principles of compiler design, up to intermediate code generation. Lexical, syntax and semantic analysis will be used in the following order to analyse the input text:

1. Identify tokens (cricket, leather bag, smart phone) and verbs in the continuous sense (e.g. eating, playing) and such, using lexical analysis.
2. Obtain the sentiment of the users to find out whether they like the item or not, and show the dependency trees for each dialogue.
3. Getting a long, unorganized list of possible recommendations based on the users’ interests, without sentiment considerations.
4. Recommending the products to the users with respect to sentiment, and recommending alternatives based on the interests of friends to users with negative sentiments.

The result will be an organized collection of information obtained from the interaction, showing the preferences and their respective categories.

## Literature Survey:

## literature_survey

## Problem Statement:

We can understand that we now have pre-trained deep learning models to work with, over machine learning models.​

We will try to separate the tokens and tag them, whilst filtering out useless words.​

The sentiment of the user's statement also needs to be considered while recommending.​

To obtain the raw recommendations, we need to check for correlations between the tokens and the product names.

Organizing the recommendations and allocating them to each user will give us our final set of recommendations.

## Architecture Diagram:

## 

## Pseudocode:

## Module – I:

For each dialogue in conversation:

Split dialogue in tokens and store them

doc = nlp(dialogue)

For each token in token\_list:

if tag == 'NOUN', add to noun\_list

else if tag == 'VERB' add to verb\_list

End for loop...

*#Using Spacy's functions, we can identify entities.*

For entity in doc.ents:

Append entity to users

Append entity\_lab to user\_types

End for loop...

End for loop....

*#Remove any leftover stop words in the nouns list*

nouns = [word for word in nouns if not word in stopwords.words()]

Store all tokens in a data frame, and return the data frame

## Module – II:

#Obtain sentiments using the NLTK Vader Sentiment Analyzer

Create a data frame to store sentiment values

sid = SentimentIntensityAnalyzer()

For each dialogue in conversation:

Calculate polarity scores for each dialogue

Append values to sentiments data frame corresponding

to each sentiment

Return the sentiment values data frame

End for loop....

*#Displaying the dependency trees....*

For each dialogue in conversation:

*#Use loaded dictionary of English words from Spacy library*

nlp = spacy.load('en\_core\_web\_md')

doc = nlp(dialogue)

*#Displacy is a visualization library for NLP*

displacy.render(doc, style = 'dep')

End for loop...

## Module – III:

For each noun in noun\_list:

For each item in complete\_list:

If lower(noun) is present in lower(item name):

Append item name to temp\_list

End for loop...

Append temp\_list to init\_rec

End for loop...

Return the list init\_rec

## Module – IV:

For index in the range of length of init\_rec:

If sentiment for user[index] is positive:

Append random items to final\_rec[index]

If sentiment for user[index] is negative:

Choose random items from other user interests

Append those to final\_rec[index]

*#We have chosen to recommend 5 items per user*

If sentiment for user[index] is neutral:

Append random items to final\_rec[index]

End for loop...

Return the list of lists final\_rec

## List of Modules (in detail):

## Module – 1:

The main focus of lexical analysis is to obtain tokens from the given text input.

We need to differentiate between entities (users, organizations, religion, etc.) and interests (garments, tech, etc.).

The technique we are going to employ is called POS tagging. (POS -> Part of Speech)

POS tagging algorithms allocate tags to each token, such as "NOUN", "VERB", "ADJ", etc., based on the similarity to their respective tagged categories.

We will be using the Spacy library, which contains pre-trained NLP models and a huge dictionary of words.

The input is parsed sentence by sentence, from which nouns, verbs, users, and user types are obtained. The respective information is summarized in a pandas data frame.

Keep in mind the input is made to understand short conversations - especially one-liners. Basically, we separate the text blob by the '\n', or newline symbol. This is done by using the splitlines() function.

If any respective dialogue consists of two lines, the parser will be confused and allocate an extra user to the extra line.

This is done with consideration to social messaging applications, as users frequently communicate with short dialogues.

## Module – II:

The main point of understanding the syntax of the sentence is to check the order by which tokens are parsed.

For example, seeing a "not" before a verb, say "enjoying", can imply a negative sentiment.

Sentiment analysis is an important tool in recommendation - it helps recommend products to those who want it.

People who express negative sentiment towards a product will not recieve recommendations on it.

Instead, they will get recommendations based on what their friends like.

The NTLK Vader Sentiment Analyzer is an excellent tool for quickly analyzing the sentiment of a sentence.

We will obtain the polarity scores in each sentiment category, i.e "Positive", "Negative", "Neutral", and "Compound", for each user.

The Spacy library consists of one more feature - the ability to show dependency trees. Dependency trees are useful in understanding the parsing process, in pursuit of finding the sentiments.

The arrows infer dependencies between words. If an arrow points from 'a' to 'b', then it can be understood that 'a' depends on 'b'.

This is useful in seeing how the tokens are tagged, and how the sentence is syntactically analyzed.

Finally, from experimentation, we have concluded that 0 is a suitable threshold to determine whether the said user should be recommended any products or not. The main reason is that, if the compound sentiment is zero or more, then the user is neutral or positive of the product, however if it is less than zero, the user does not like the product. Based on this logic, we recommend the products.

## Module – III:

We will check the list of nouns from our summary of tokens to see what the interests of the users are.

If the particular noun is present in any product name, we will store that product name in a new list.

This new list is considered as our initial set of recommendations.

Now, our initial recommendations are quite a large number.

We will pick five products at random, to recommend to each user.

Of course, we do not want to get the same products again in our random selection.

This is easily achieved by using random.sample(list, k), where k is the number of items selected. This function also ensures that there is no repetition in the selection of items.

## Module – IV:

This is the final recommendation.

You can see the user's names followed by a list of recommendations based on their interests.

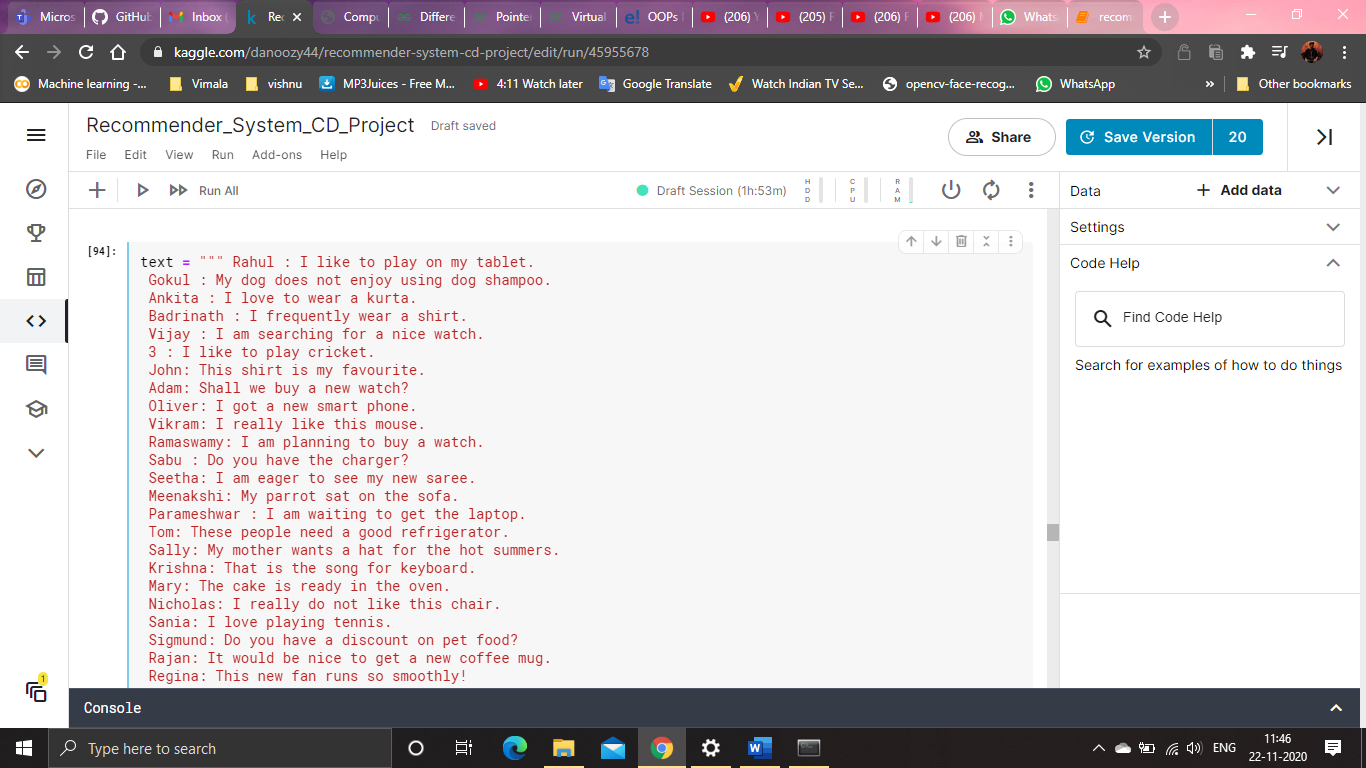
Keep in mind that for the most part the recommendations are accurate, but you may chance upon some outliers, due to the random nature of product selection.

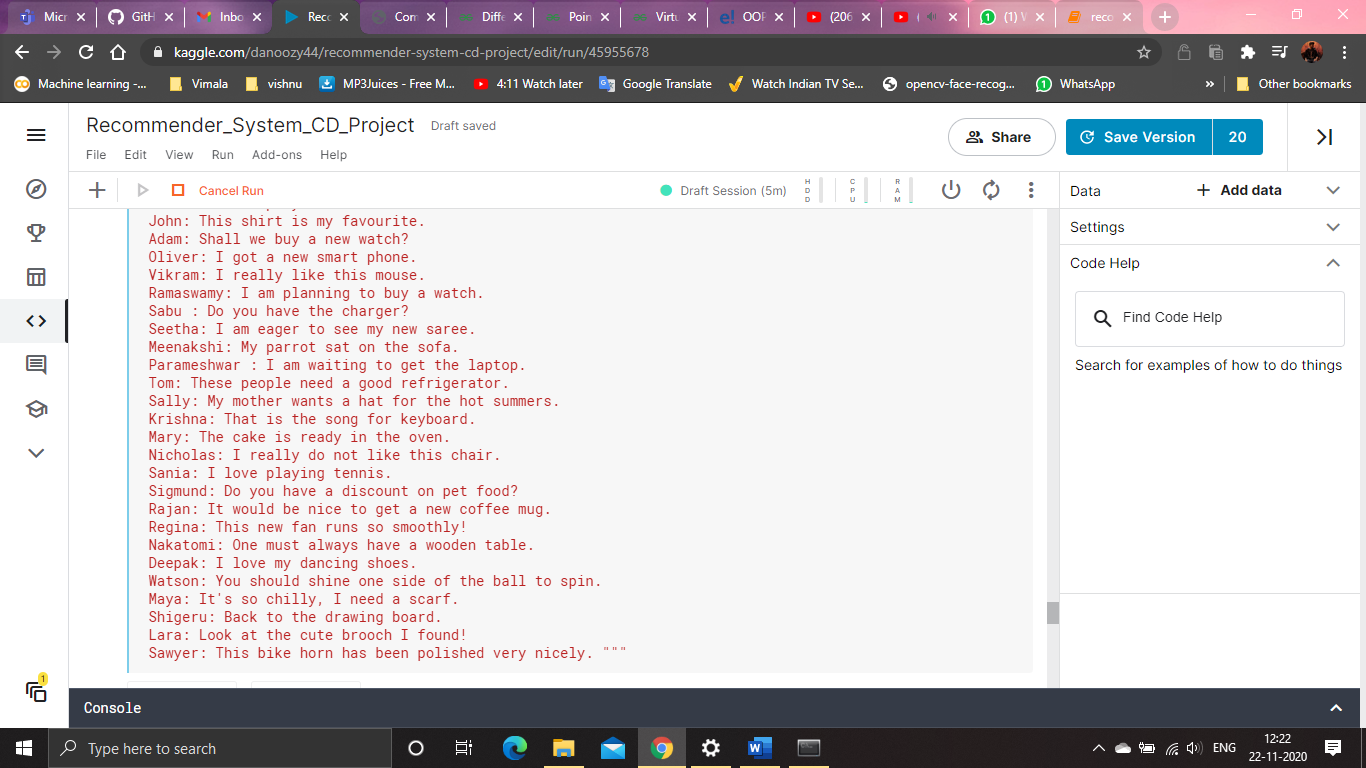
As stated before, the sentiment of the user is considered as a factor in the recommendation process. The compounded polarity score is considered here.

If a user's sentiment is negative, then we pick a random recommendation based on the interests of that user's friends.

Otherwise, if it is positive or neutral, we can directly recommend based on their individual interest.

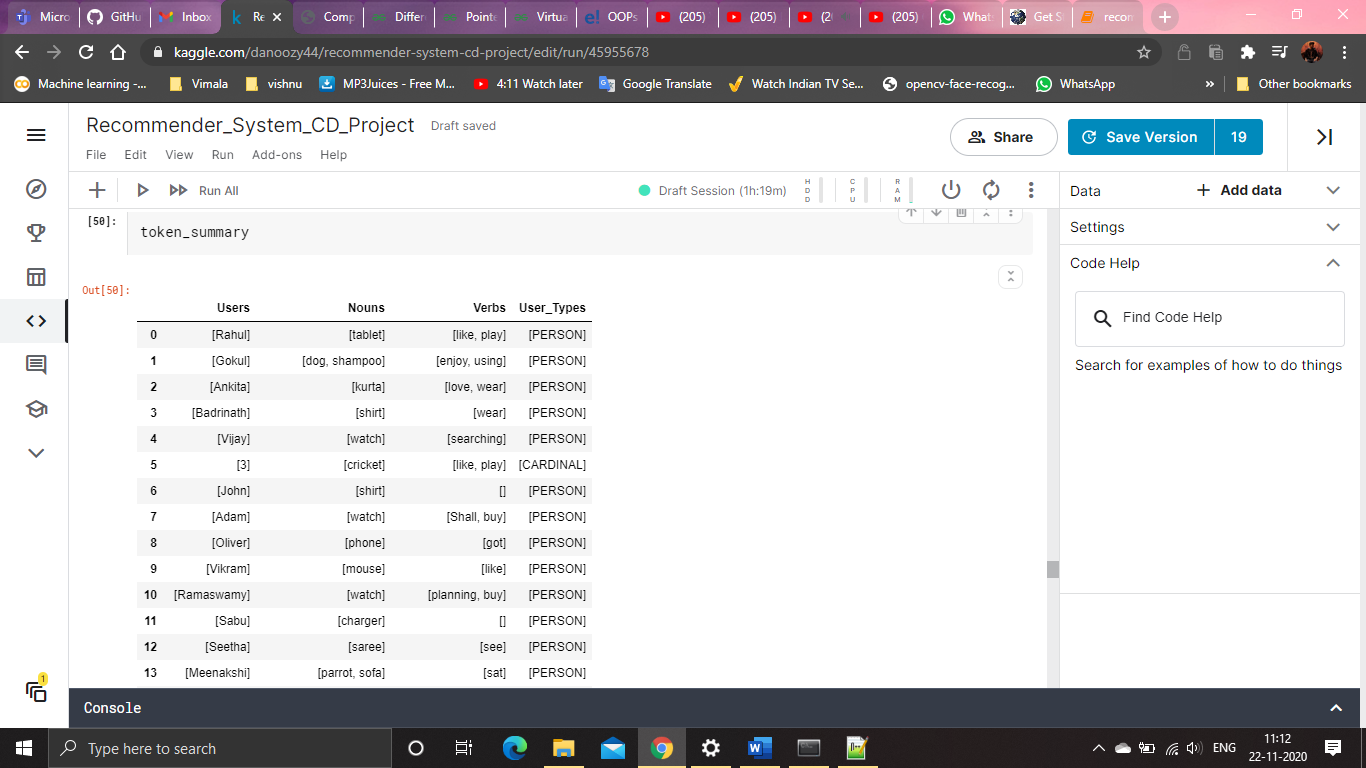
**Input:**

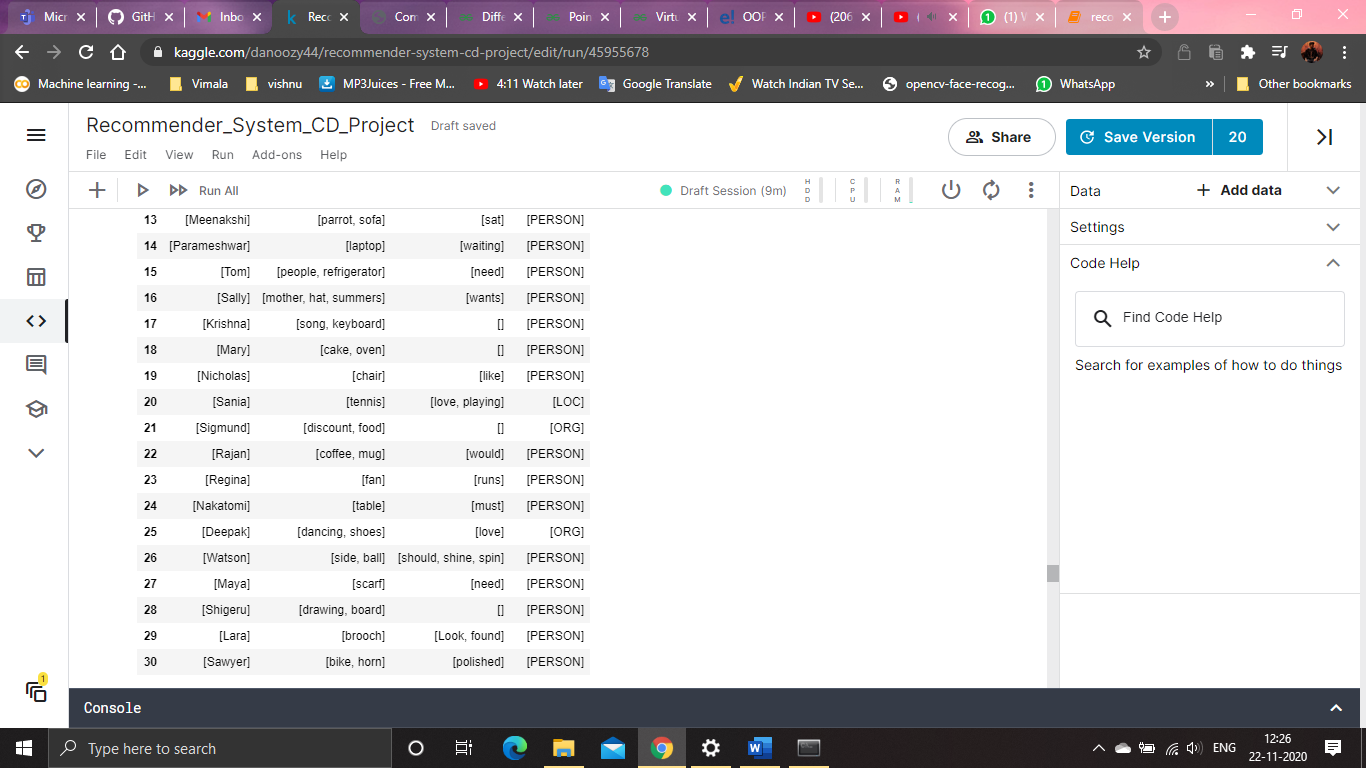




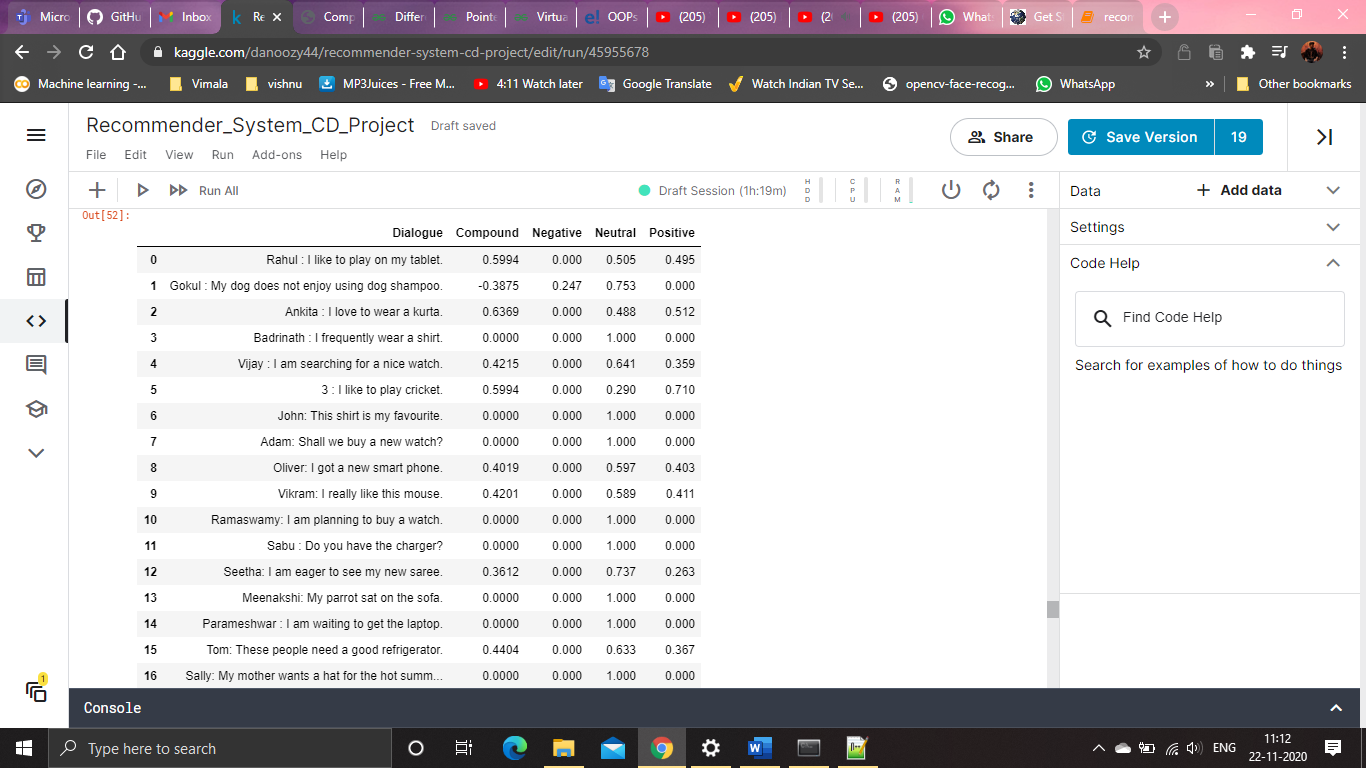
## Output:

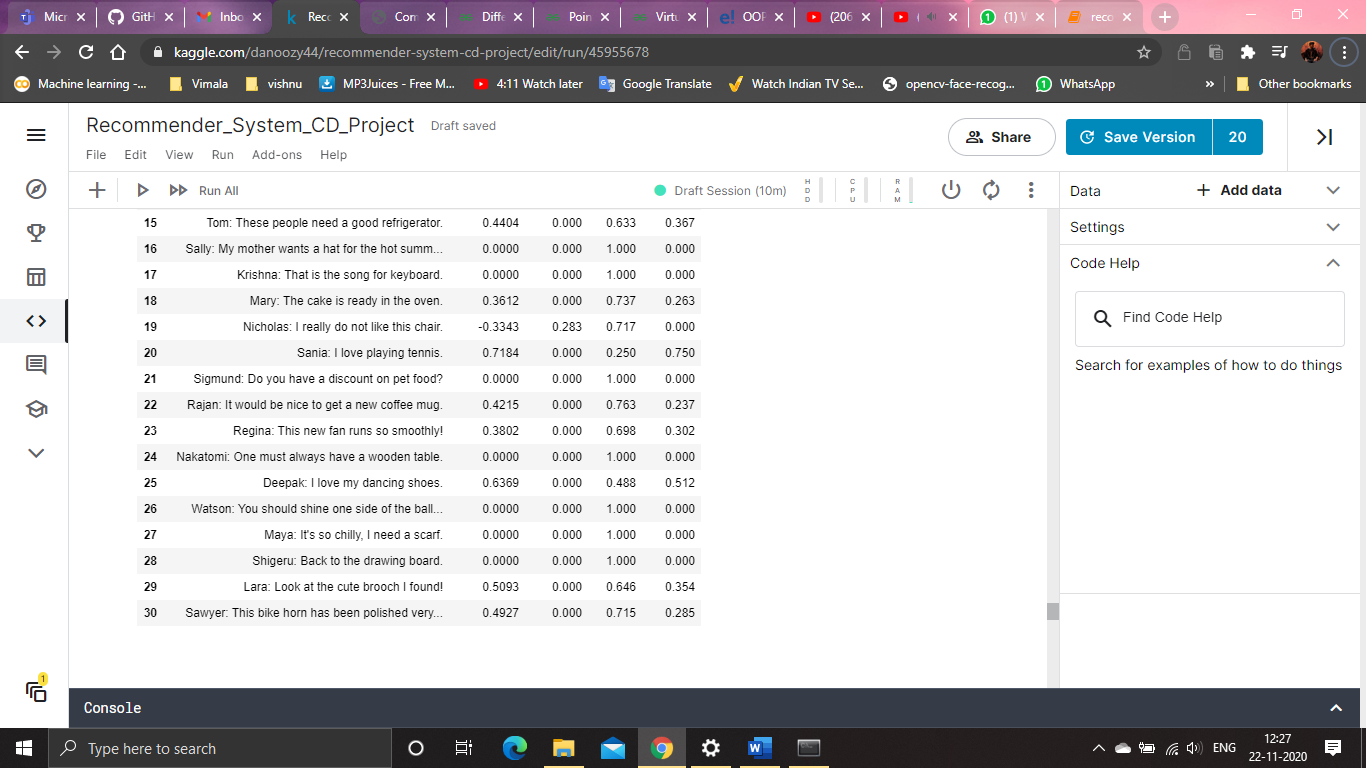
**Module 1(Tokenization):**



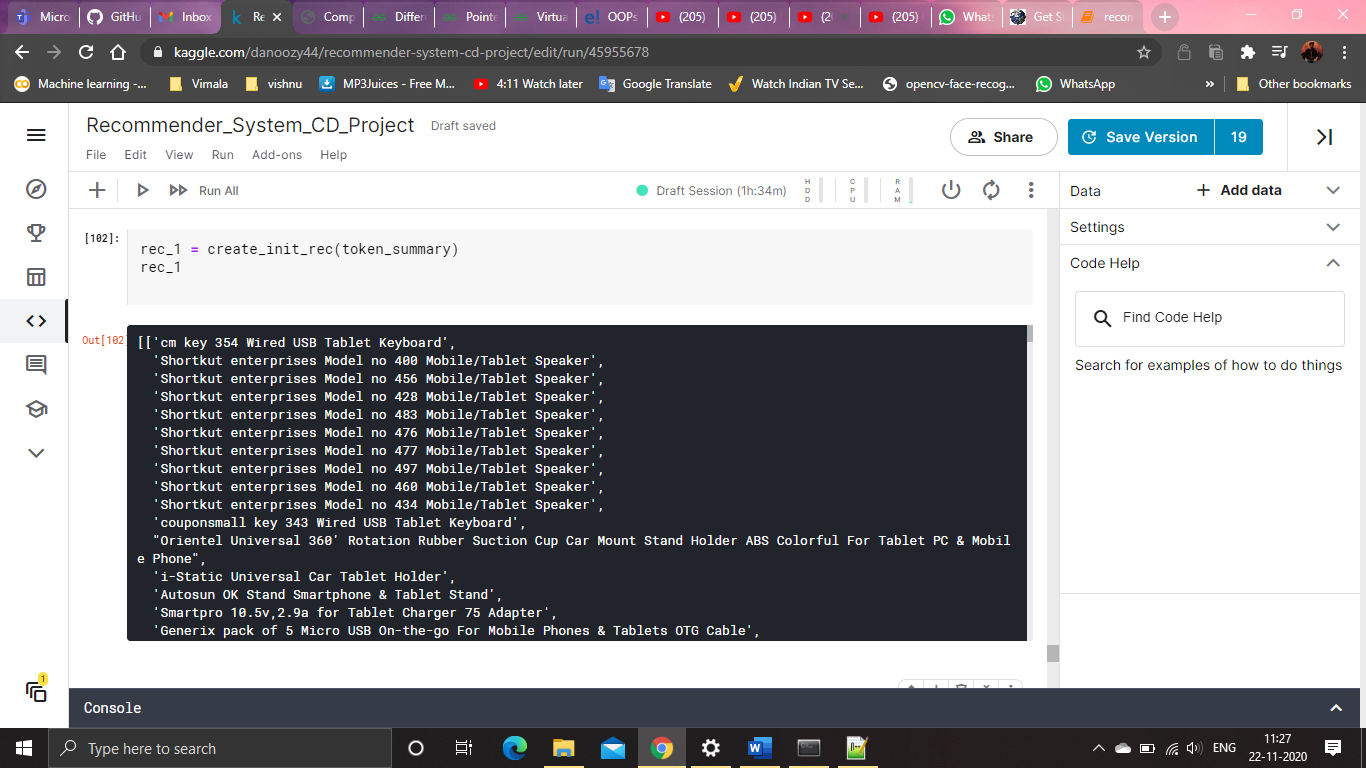


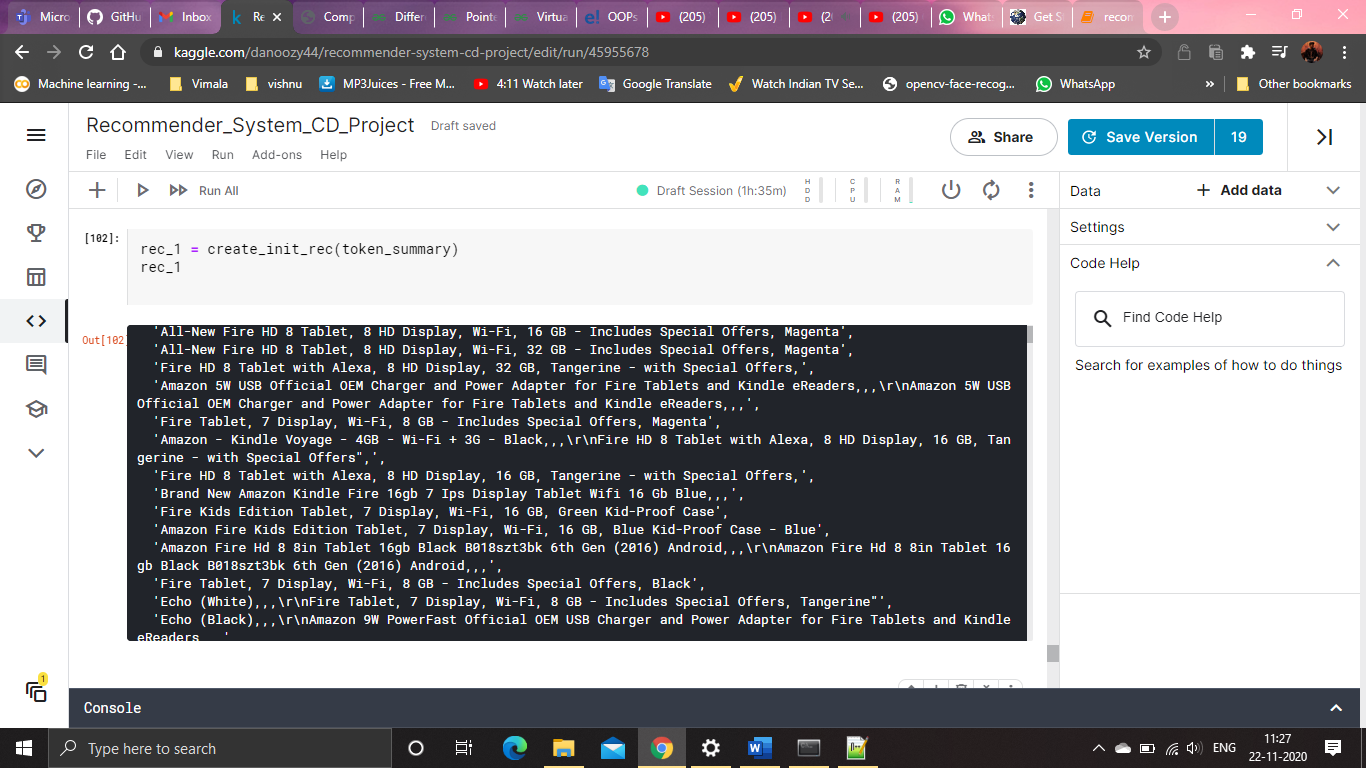
**Module 2(Syntax analysis):**

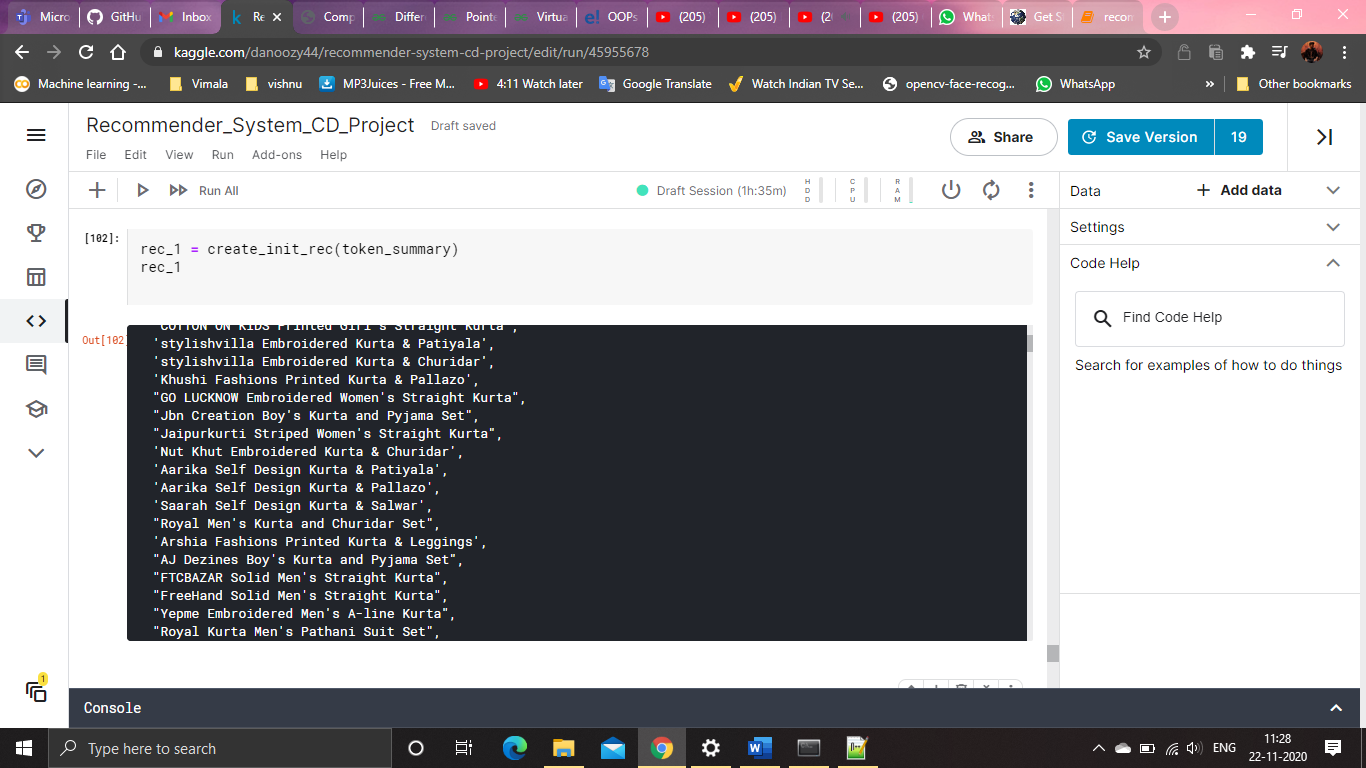


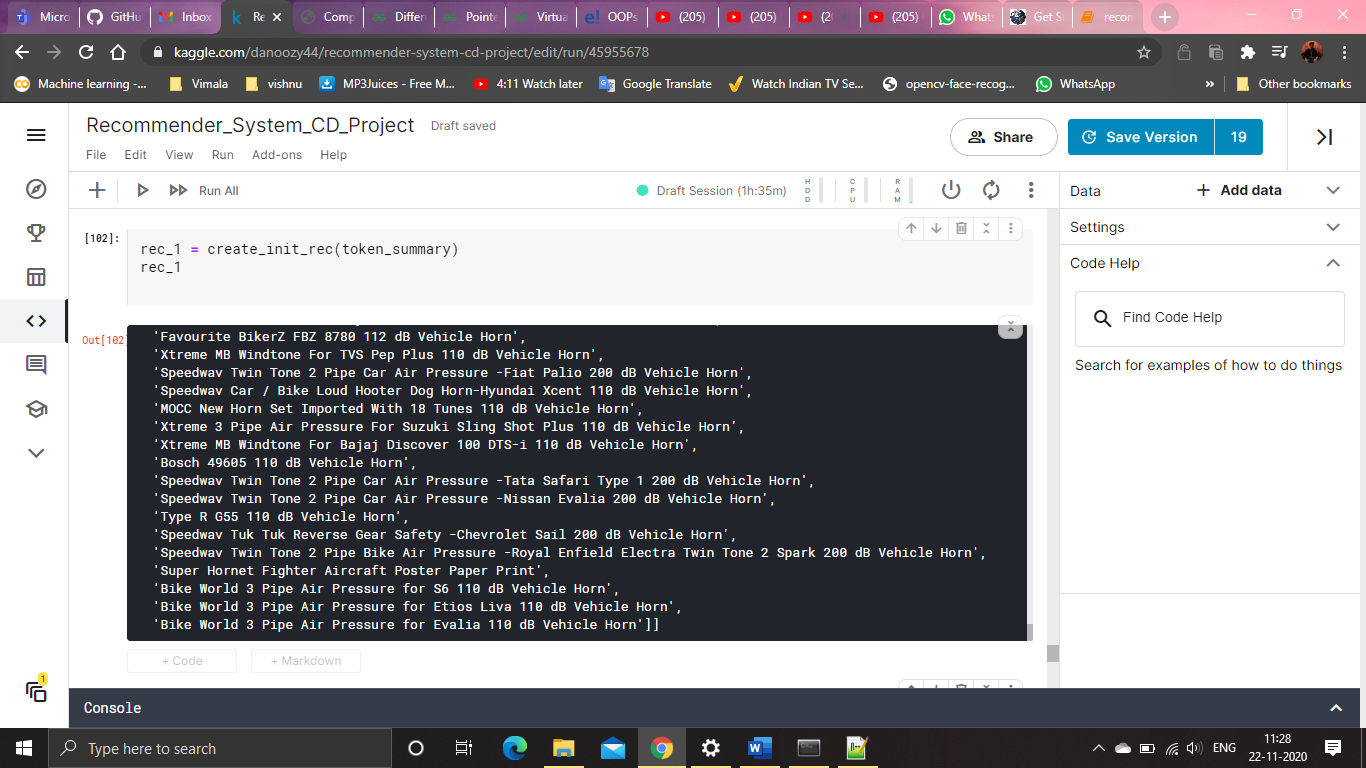


**Module 3(Top Recommendation):**



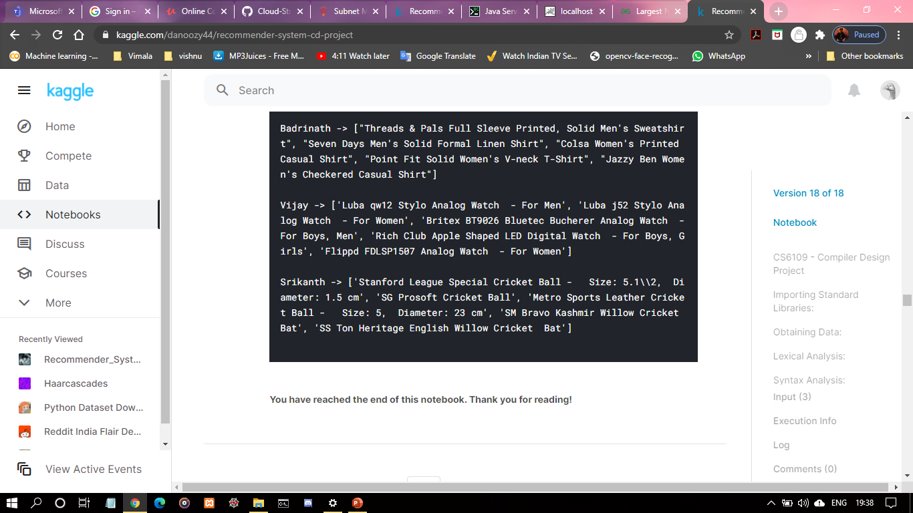






**Module 4(Final recommendation):**

## 



## Results Discussion:

Since we’ve checked for subsequences of the occuring string, the nouns, inside product names, of course we cannot guarantee an accuracy of 100%.

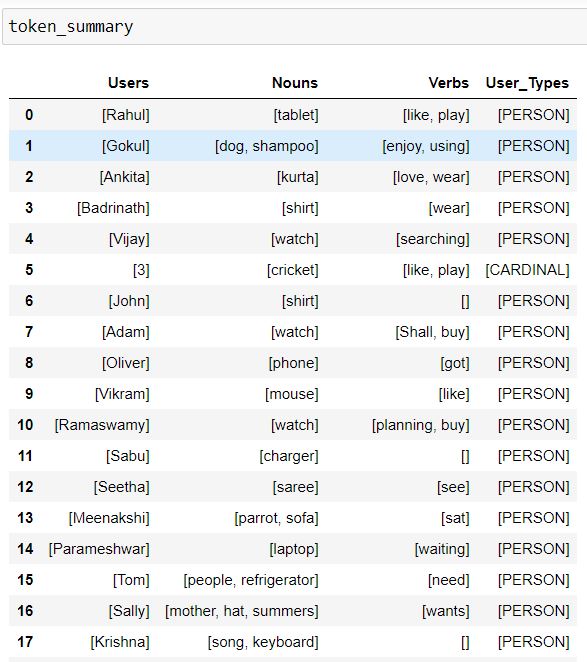
Regardless, the recommendation system seems to works fairly well - it recommends what the users want based on their interests.

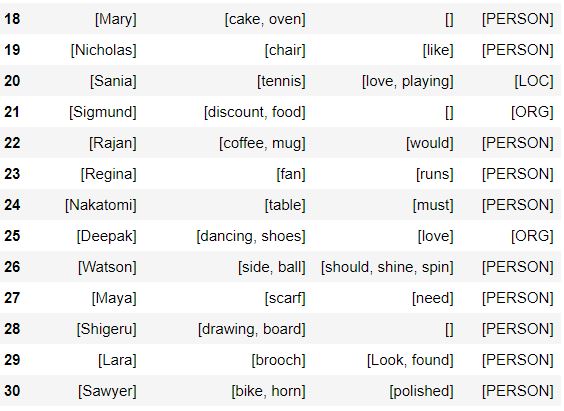
For users with negative sentiment - it doesn’t recommend the products associated with the user. Rather, it recommends products associated with the user’s friends.

We also ensure any product recommendation is not repeated for the same user.

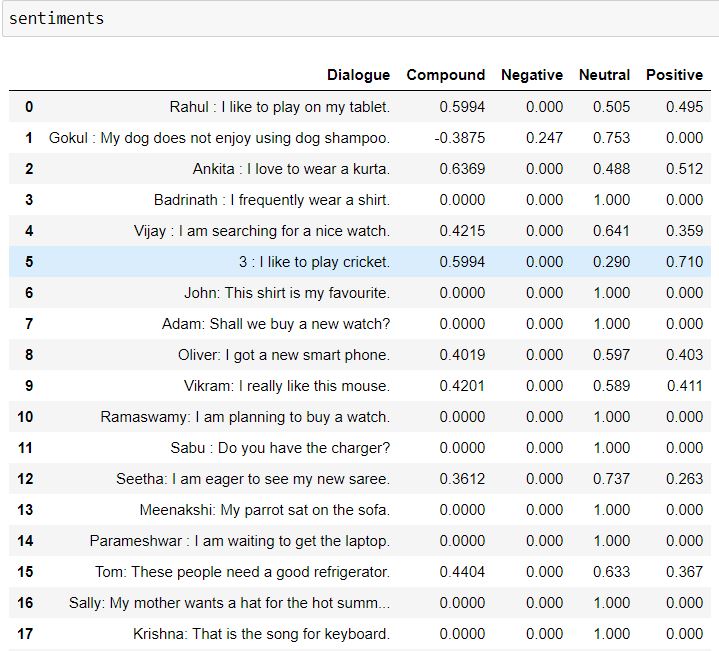
The end result is fairly straight-forward. It works for most cases, with some exceptions, a rare case.

**Module 1(Tokenization):**



****

**Module 2(Syntax analysis):**





**Module 3(Top Recommendation):**

|  |
| --- |
| [ 'cm key 354 Wired USB Tablet Keyboard',  'Shortkut enterprises Model no 400 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 456 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 428 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 483 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 476 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 477 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 497 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 460 Mobile/Tablet Speaker',  'Shortkut enterprises Model no 434 Mobile/Tablet Speaker',  'couponsmall key 343 Wired USB Tablet Keyboard',  "Orientel Universal 360' Rotation Rubber Suction Cup Car Mount Stand Holder ABS Colorful For Tablet PC & Mobile Phone",  'i-Static Universal Car Tablet Holder',  "RETAIL SHOP INDIA Printed Girl's Straight Kurta",  "Love More Printed Women's Flared Kurta",  'Saarah Self Design Kurta & Churidar',  'Salwar Studio Geometric Print, Paisley Kurta & Salwar',  'Salwar Studio Self Design Kurta & Salwar',  'F-Loop Self Design Kurta & Churidar',  "Aarika Girl's Kurta and Pallazo Set",  "Swag India Girl's Kurta and Pallazo Set",  "Retaaz Girl's Kurta and Leggings Set",  "NIPPY Boy's Kurta, Waistcoat and Breeches Set",  'JDK NOVELTY Brocade, Art Silk, Organza Floral Print Multi-purpose Fabric,  Kurti Fabric, Blouse Material,  Kurta Fabric, Dress/Top Material, Jacket Fabric, Kurta & Pallazo Material, Sherwani Fabric, Kurta & Churidar Material, Lace, Salwar Suit Material, Kurta & Patiyala Material, Lehenga Choli Material',  'JDK NOVELTY Brocade, Art Silk, Organza Self Design Multi-purpose Fabric, Kurti Fabric, Blouse Material, Kurta Fabric, Dress/Top Material, Jacket Fabric, Kurta & Pallazo Material, Sherwani Fabric, Kurta & Churidar Material, Lace, Salwar Suit Material, Kurta & Patiyala Material, Lehenga Choli Material',  'JDK NOVELTY Brocade, Art Silk, Organza Floral Print Multi-purpose Fabric, Kurti Fabric, Blouse Material, Kurta Fabric, Dress/Top Material, Jacket Fabric, Lehenga Choli Material, Kurta & Patiyala Material, Salwar Suit Material, Lace, Kurta & Churidar Material, Sherwani Fabric, Kurta & Pallazo Material',  "TSG Breeze Printed Women's Round Neck Multicolor T-Shirt",  "Starsy Printed Women's Round Neck Green T-Shirt",  "Clovia Women's T-Shirt Bra",  "run of luck Solid Women's Round Neck Dark Blue T-Shirt",  "Starsy Solid Women's Round Neck Green T-Shirt",  "Attitude Printed Women's Round Neck Black T-Shirt",  "The Cotton Company Solid Women's Polo Neck Pink T-Shirt",  'Adimani Mustuw Scarf Fabric Necklace',  'Adimani Felimq Scarf Fabric Necklace',  ] |

**Module 4(Final Recommendation):**

|  |  |
| --- | --- |
| ***USERS*** | ***RECOMMENDED PRODUCTS*** |
| **Rahul** | 'Brand New Amazon Kindle Fire 16gb 7 Ips Display Tablet Wifi 16 Gb Blue,,,', 'Amazon 9W PowerFast Official OEM USB Charger and Power Adapter for Fire Tablets and Kindle eReaders,,,\r\nAmazon 9W PowerFast Official OEM USB Charger and Power Adapter for Fire Tablets and Kindle eReaders,,,', 'All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi, 32 GB - Includes Special Offers, Black', 'Echo (White),,,\r\nFire Tablet, 7 Display, Wi-Fi, 8 GB - Includes Special Offers, Tangerine"', 'All-New Fire HD 8 Tablet, 8 HD Display, Wi-Fi, 16 GB - Includes Special Offers, Blue' |
| **Gokul** | 'Brand New Amazon Kindle Fire 16gb 7 Ips Display Tablet Wifi 16 Gb Blue,,,', "Gulmohar Jaipur Embroidered Women's Straight Kurta", 'Aarika Self Design Kurta & Pallazo', "ShowTime Women's Full Coverage, T-Shirt Bra", 'HMT S3648 Analog Watch - For Men', 'Spick Cricket Shoes', "Mast & Harbour Women's Printed Casual Shirt", 'Gift Island SW13390E Fashionin Digital Watch - For Men', 'AdroitZ Premium Phone Socket Holder For Nokia N9', 'Ekku Mickey Mouse - 15 inch', 'Disney DW100405 Digital Watch - For Boys', 'Hako 85W Magsafe Laptop Charger For APPLE MacBook Pro 17 MA897LLA 85 Adapter', 'AMY Karena Kapoor in Green Saree A5 Notebook Spiral Bound', 'Adaa Terracotta - Reclining Ganesha on Sofa - Pink Showpiece - 11.9 cm', 'PRINT SHAPES Best green ball Laptop Skin with Mouse pad Combo Set', "People Women's Checkered Formal Shirt", 'AKUP happy-mothers-day Ceramic Mug', 'Saco Karbonn Smart Ta Fone A37 Hd Tablet Wired USB Tablet Keyboard', 'MasterweaverIndia Cotton Woven Blouse Material', 'RoyalOak Metal Outdoor Chair', 'Dunlop Fort Tennis Ball - Size: 0, Diameter: 2.5 cm', 'Royal Canin Maxi Starter 1kg Vegetable Dog Food', 'Prithish Wild And Free Ceramic Mug', 'JRB 1019 Smallest Mobile Powered By OTG Enabled Android Smart Phone Portable 1019 USB Fan', 'Craft And Curtain Multicolor 178 cm Table Runner', 'N Five Running Shoes', 'Uniball Kurutoga Round Shaped Pencils', 'Dimpy Stuff Bear W/Scarf in Sitting 50Cm - 50 cm', 'Satzuma UCH100 USB Clapper Board', 'B-Fashionable Combo of Drop & Round Bead Kilangi Brooch', 'Cm Treder cm h4 light 145 Motorbike LED Bulb' |
| **Ankita** | "Gulmohar Jaipur Embroidered Women's Straight Kurta", "Jeet Boy's Kurta and Pyjama Set", "Gulmohar Jaipur Printed Women's Straight Kurta", 'Aarika Self Design Kurta & Pallazo', "Swag India Girl's Kurta and Pallazo Set" |
| **Badrinath** | "J Marks Men's Striped Formal Shirt", "ShowTime Women's Full Coverage, T-Shirt Bra", "Penny Pro Women's T-Shirt Bra", "Concepts Women's Printed Casual Shirt", "Omtex Solid Men's Polo Neck T-Shirt" |
| **Vijay** | 'Now B170-SLL09 Now Analog Watch - For Girls', 'Cartier W6701005 Analog Watch - For Boys, Men', 'Zoop C4040PP03 Digital Watch - For Boys, Girls', 'Alfajr WQ18 Qibla Compass Digital Watch - For Men, Boys, Girls', 'HMT S3648 Analog Watch - For Men' |

## Conclusion:

The information obtained from direct interactions can be a valuable tool, as we can create recommendation systems on the basis of user-user interactions, such as comments on a forum, rather than user-site interactions.

Previous systems understand and provide recommendations based on similarities between users, which might lead to a bias evaluation for user preferences. This provides a solution for this issue by integrating direct interactions into recommendation systems.

However, there is still much left to improve, and a full accuracy of such systems is never guaranteed, but this approach will definitely up the bar in extending further recommendation systems.

## References:

1. C.Li, F.Xiong, “Social Recommendations with Multiple Influence from Direct User Interactions”, Aug. 2017.

1. C. Pan, W. Li, Research paper recommendation with topic analysis in Computer Design and Applications IEEE, 4 (2010), pp. V4-264

1. J.A. Konstan, J. Riedl, Recommender Systems: From Algorithms to User Experience, User Model User-Adapt Interact, 22 (2012), pp. 101-123