FLIP ROBO

COMMENT RATING CLASSIFIER

Submitted by:

KISHANKUMAR BAROCHIYA

ACKNOWLEDGMENT

Here, I am acknowledge that all data and process are done as best my knowledge. Used data are scraped from Flipkart.com. Used data are only for studying and analysis purpose. I am thankful KGPtalkie, google and GitHub for helping me for knowledge.

INTRODUCTION

- Business Problem Framing

  In the world of online shopping and social media, we all are aware about comment and rating, which are given by users for particular product, or content gives. Now sometimes users share only comment but not give rating. Here we will build a model, which can predict rating based on comment.

- Conceptual Background of the Domain Problem

  Suppose we buy a product from Flipkart and for this product, we shared rating and give our feedback on portal. This ratings and comments will now become useful for users who wants to buy same product. It will be very useful for users because they can get better review and understanding about product.

- Review of Literature

  We scraped data from Flipkart where we take Samsung mobile phones and scrape its review. We scraped 20000+ reviews for build model. In this section, we have comments where emoji used, short keywords used and other words also.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  Here we have data where total 23970 rows and 3 variables available. One variables is dummy so we will drop it and apart from it there are only 2 variables one is comment and one is rating.

- Data Sources and their formats

  Data are scrapped from FlipKart.com and it is objective and int64 format. There is no null values in dataset. In comment section, we have uncleaned data as many users use emoji, short words, long words, repetitive words etc. replace special character, remove space

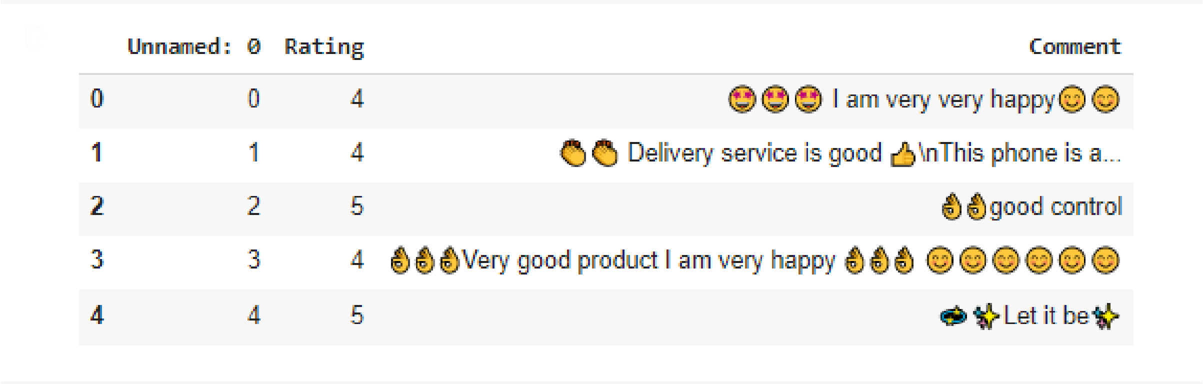
- Data Pre-processing Done

As we scrape data in raw form, we must need to clear and pre-process if for better accuracy before feeding in word cloud.

Here we used KGPtakie pre-processing library where we cleaned Upper case to lower case, short keywords to regular keywords, repeating keyword like superrrr to super, emoji?, replace special character, remove space

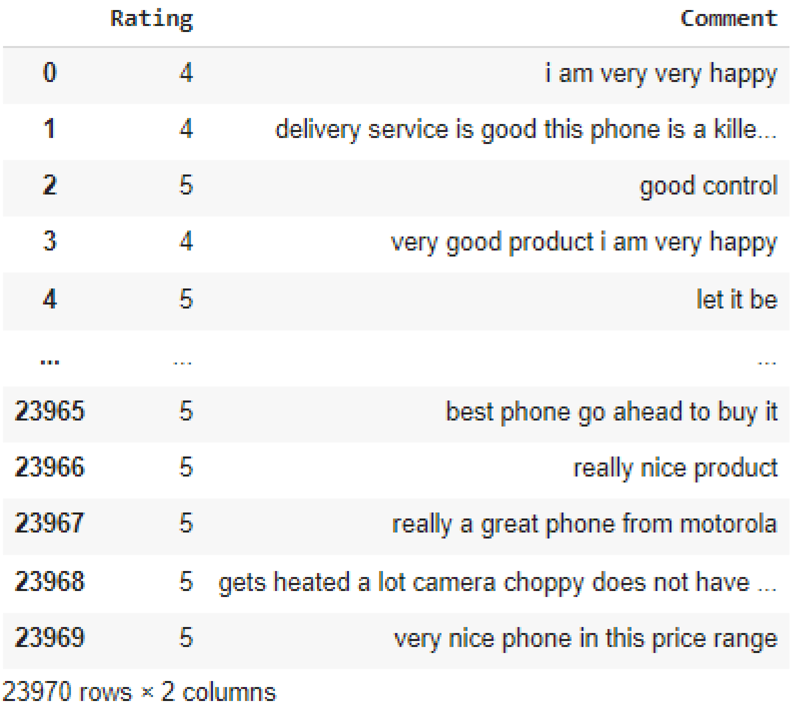
- Data Inputs- Logic- Output Relationships

Here, we can see both dataset one is before pre-processing and other one is after pre-processing.



	Unnamed: 0	Rating	Comment
0	0	4	😍😍😍 I am very very happy😊😊
1	1	4	👏👏 Delivery service is good 👍\nThis phone is a...
2	2	5	👌👌good control
3	3	4	👌👌👌Very good product I am very happy 👌👌👌 😊😊😊😊😊😊
4	4	5	🐟✨Let it be✨

. After pre-processing:



	Rating	Comment
0	4	i am very very happy
1	4	delivery service is good this phone is a kille...
2	5	good control
3	4	very good product i am very happy
4	5	let it be
...	...	...
23965	5	best phone go ahead to buy it
23966	5	really nice product
23967	5	really a great phone from motorola
23968	5	gets heated a lot camera choppy does not have ...
23969	5	very nice phone in this price range

23970 rows × 2 columns

Now we have clear comments so now by using other features, we will feed comments as input and our model will process it and find out frequent words used in particular class.

- Hardware and Software Requirements and Tools Used

  Here we used Selenium for Datascraping, KGPtalkie for pre-processing and Python for coding

Model/s Development and Evaluation

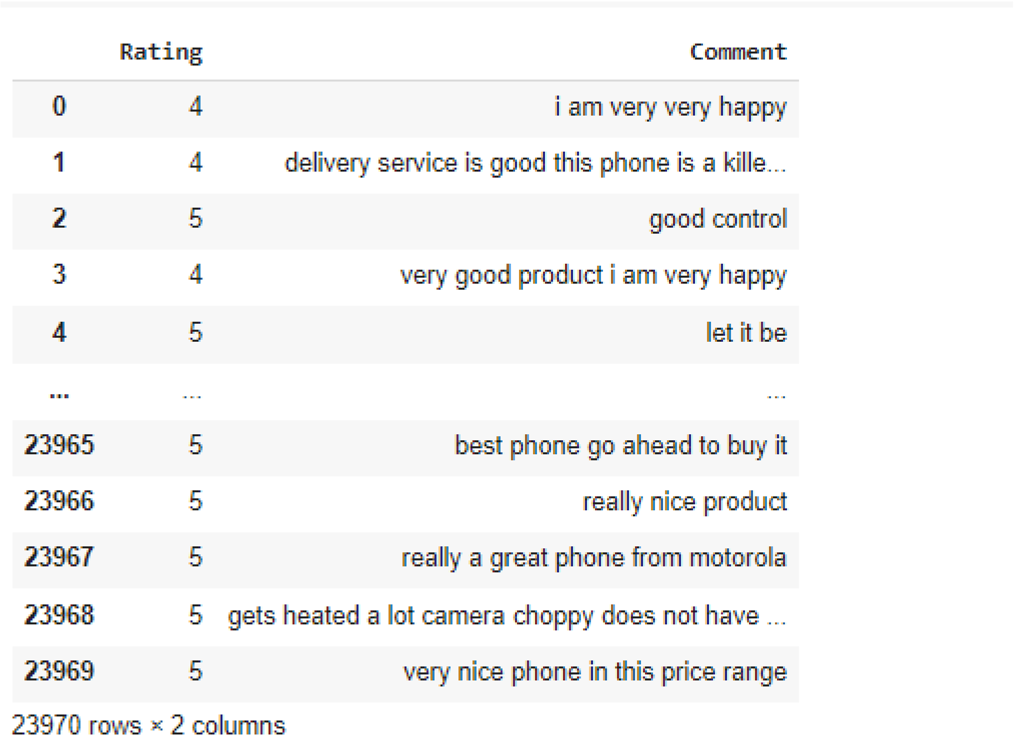
- Identification of possible problem-solving approaches (methods)

  First approach is to scrape 20000 comments and rating. Now understand the dataset. Removing null values and un-related columns. Now main attention is required that how we clean comments like to remove space, capital words, short form, emoji's etc. than covert text into vector form and create word cloud where frequent words used in particular rating category is divided and based on it we will predict the other comment

- Testing of Identified Approaches (Algorithms)
  - Selenium
  - Python
  - Seaborn
  - Matplotlib
  - Kgptalkie
  - TfidfVectorizer
  - train_test_split
  - classification_report

- Run and Evaluate selected models

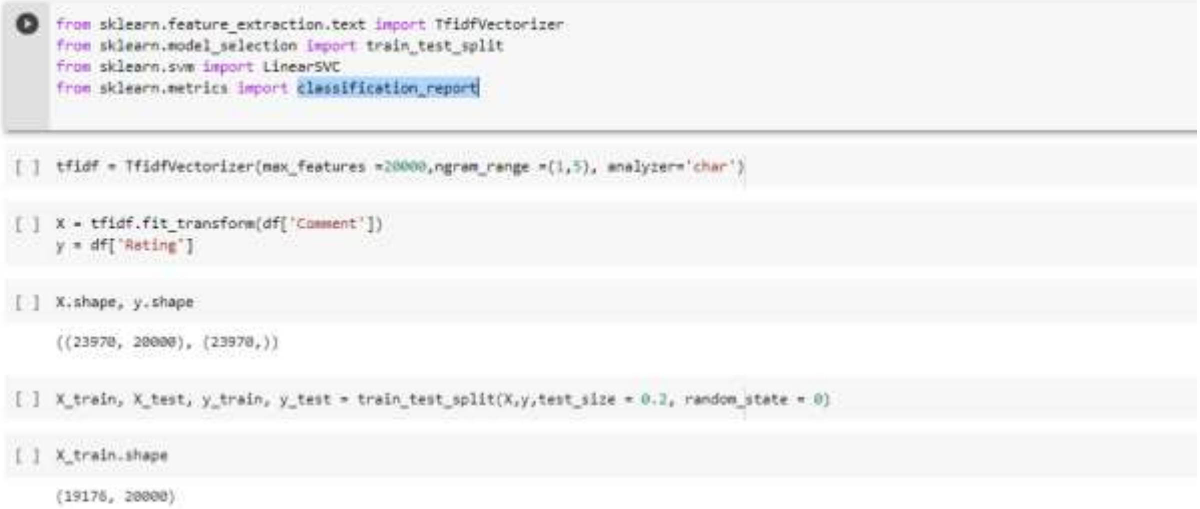
  After cleaning the data we have dataset as below.



	Rating	Comment
0	4	i am very very happy
1	4	delivery service is good this phone is a kille...
2	5	good control
3	4	very good product i am very happy
4	5	let it be
...	...	...
23965	5	best phone go ahead to buy it
23966	5	really nice product
23967	5	really a great phone from motorola
23968	5	gets heated a lot camera choppy does not have ...
23969	5	very nice phone in this price range

23970 rows × 2 columns

Now we will convert it into vector and split data into training and testing.



```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.svm import LinearSVC
from sklearn.metrics import classification_report
```

```
[ ] tfidf = TfidfVectorizer(max_features =20000,ngram_range =(1,5), analyzer='char')
```

```
[ ] X = tfidf.fit_transform(df['Comment'])
    y = df['Rating']
```

```
[ ] X.shape, y.shape

    ((23970, 20000), (23970,))
```

```
[ ] X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.2, random_state = 0)
```

```
[ ] X_train.shape

    (19176, 20000)
```

After it we will go with linear regression and check with classification report that how much accuracy we got.

- Key Metrics for success in solving problem under consideration

  Key metrics is Data cleaning process that how pretty I handle my row comment and clean it because model will understand and find out frequent words from it and proceed so model is purely based in it.

CONCLUSION

- Key Findings and Conclusions of the Study

  Here outcome is that rating is predicted based on words used in comment. Sometimes it change like if we comment good but I am average satisfied then I will give 4 rating and if am fully satisfied than will give 5 rating but in input data good words in found so it is not based on that particular words comes only in one category we can get nearby result.

- Learning Outcomes of the Study in respect of Data Science

  Here outcome is that rating is predicted based on words used in comment. Sometimes it change like if we comment good but I am average satisfied then I will give 4 rating and if am fully satisfied than will give 5 rating but in input data good words in found so it is not based on that particular words comes only in one category we can get nearby result.

- Limitations of this work and Scope for Future Work

  We need to modify comments cleaning as it have different type of format. It is totally depend on quality of comment and how we cleaned it. If there is not much effort put on cleaning then we will not getting proper accuracy.