**Website Recommendation System**

**Group D1:**

1) Manan Satra

2) Mohit Bhagchandani

3) Kriti Ruhela

4) Rupal Desai

5) Namita Powar

* **Motivation:**

E-commerce is on the rise now a days. The number of people using the websites for buying and selling products has increased drastically thus increasing the competition between the websites for providing better service and facilities to the customers. Because of this the customers are often confused about where to buy the product from. Each time a person has to buy a product he has to go to each of the website and compare the factors of the product like cost, discount offered, delivery time etc. himself. This burdens the customer and thus many of the times customer ends up with just moderate deal even though there were other websites present, providing a better deal. To avoid this confusion and discomfort of customers, we are developing a system which will provide recommendation to the users by comparing two different websites. The main aim of the proposed system will be to reduce the burden of the customers and provide them with a recommended website which will help them make comparison between any two products. When a customer purchases a product it is a big investment for him and so during this purchase there are multiple factors to be considered and the proposed system does this task for him by comparing the features and providing him with the best possible option available.

* **Background:**

We have considered two websites that is amazon and bestbuy. We have studied the structure of both the websites separately. Scraping has been performed on both the sites for particular product (for now Samsung Galaxy S8 Midnight Black Unlocked version, Moto G 5 Plus). Features like final price, customer reviews and total number of reviews have been extracted from both the website. The reviews have been pre-processed by removing stop-words and performing lemmatization. This above process helped in cleaning the data. Further, sentiment analysis has been performed on customer reviews and intensity is found. After that, multi-label classification is performed on cleaned data using Linear SVC, which provided us with the predicted labels. According to labels, the reviews which are holding positive intensity value are added. This summation value is used to calculate the common factorization value. Then the final result is calculated by multiplying common factor which the sum of each label’s intensity. To give the recommendation to the user which gives a recommendation based on labels and also is provided an overall recommendation. Trivago is the top hotel price comparison site. It compares different hotels based on price and recommends the best hotel for the user. Our recommendation system, is being developed to do the similar task of recommending best deal for a particular product from different sites. Trivago recommends hotels based on only one factor of price, whereas our recommendation system compares a product based on different factors like customer reviews, price, and total number of reviews. In addition to that, our system is used to compare cell phones from different websites. As cell phone is one of the requirement of every individual for day to day life and developing a recommendation system is really going to be a great help and will make life easier for end users.

* **Approach:**

For providing the recommendation to the user we need to follow a certain procedure. The website recommendation system will work using the reviews given by the users of the product at the websites. Python 3 is used for designing purposes and the development is done in Anaconda IDE using Jupyter notebook. Currently there are two websites [www.amazon.com](http://www.amazon.com) and [www.bestbuy.com](http://www.bestbuy.com) acting as the data source. The system will derive at the recommendation using steps like:

1. **Data Scraping:**

Now the first step that is data extraction from the websites will be done using different packages available in python for web scraping. Data available on the websites is in the form of HTML webpages and hence we used packages like BeautifulSoup4 and requests to perform scraping of the attributes like price of the product, delivery time, ratings and reviews. The requests module is used for parsing XML and HTML documents very quickly, even handling messed up tags in the process and BeautifulSoup module is used for navigating, modifying and searching the data. The data present on both the websites was in the structured form but the format of the tags and attributes present was different for each of the webpage. The data was presented using different HTML tags and the hierarchy of the tags present for the data was also different for both the websites. After scraping the data from the webpages, text pre-processing was done on it that is the processes like lemmatization, removal of stop words were performed.

1. **Multi-label Classification:**

After the pre-processed data was obtained the reviews need to be analysed further in order to get the prominent features from the scraped data. Classification was used in order to get the features from each of the review present on both the websites. Classification is the process in which we train the model in order to get accurate results. While training the model we need to provide input as well as the output to the classifier so that the classifier knows what the desired results are for the input provided to it. We provide some reviews as an input to the classifier and as output classification labels are provided. For providing the output that is the classification label to the classifier we scanned few reviews manually. After manually scanning the reviews we got some of the factors occurring repeatedly and these factors were provided as output to the classifier. This data that we provide to the classifier is considered as the training data. Multi-label classification provided us with the predicted labels for the reviews. This process was done using Linear SVC model. Linear Support Vector Classification is used when we have huge datasets to be considered for the classification.

1. **Sentiment Analysis:**

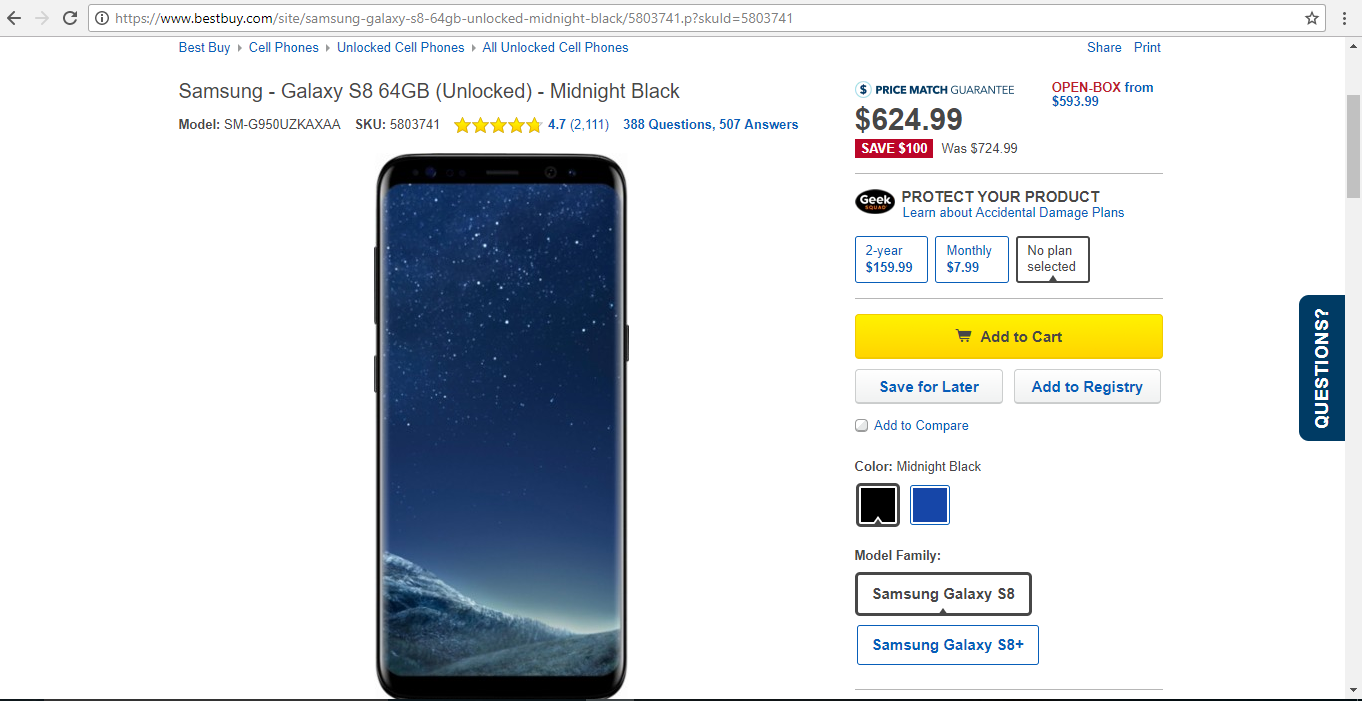
Along with multi-label classification being performed on the pre-processed data, sentiment analysis is also performed on it. The main objective behind sentiment mining is to get the overall sentiment of the review that is with the help of sentiment mining here we try to understand what the user or consumer was trying to convey about the product through the review. The overall sentiment of the review is calculated using VADER tool. VADER tool analyses a piece of text to check if any words in the text is present in the lexicon. The review as a whole is given as input to the tool and for that particular review we get the positive, neutral or negative lexicon ratings. Calculating the sum of these lexicon ratings we get the final metric that is the compound score. The compound score is standardised between -1 and 1 based on some heuristics. The main component used for getting the sentiment of the review is the compound score. Using the standard values of the lexicon ratings we analysed if the review is positive, negative or neutral. These computed values of the compound score and the sentiments are used for the further steps.

After performing multi-label classification we get the labels for the data and after performing sentiment mining on the data we get compound score. With the help of these labels and compound score we calculated a common factor useful for the calculation of the final recommendation of the website. The factors that multi-label classification gave us were price, delivery, customer-service, faulty product. By multiplying the compound score with the sum of the sentiments for each predicted label we get the value of the common factor. The value of common factor is between 0 and 1 for the normalisation purpose.

**Data sources:**

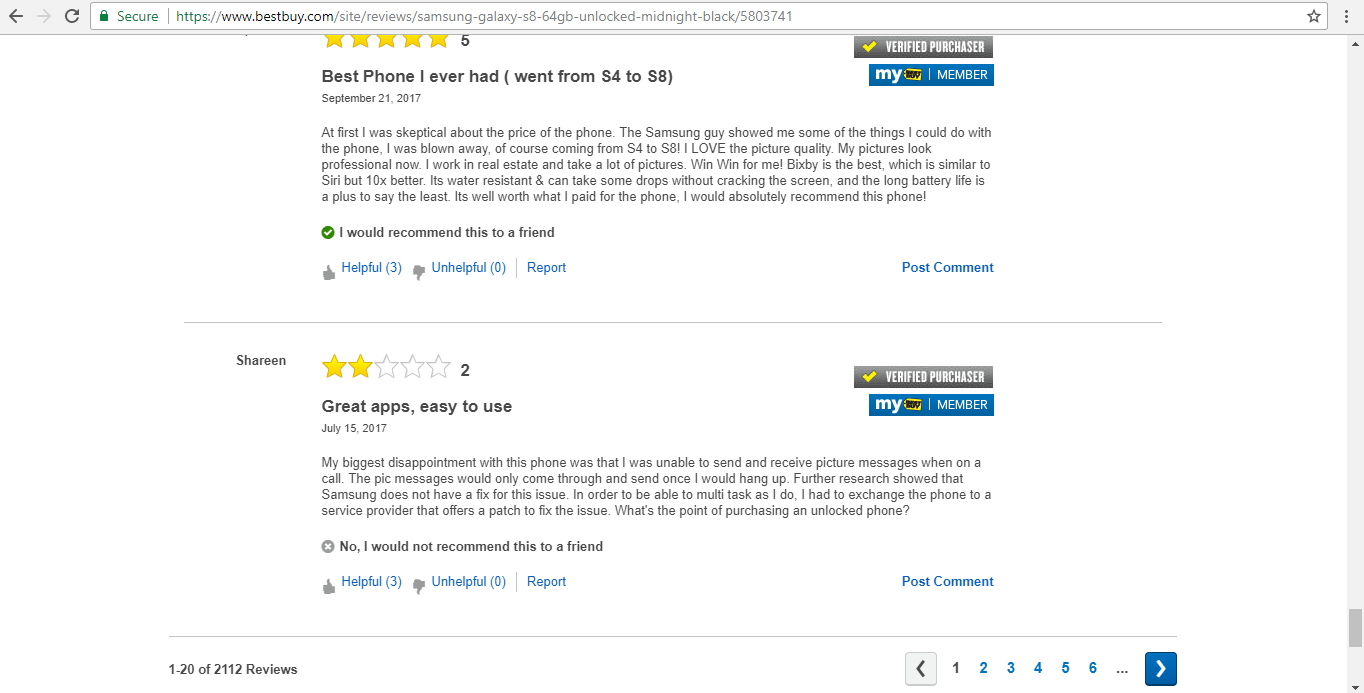
BESTBUY HOME PAGE

<https://www.bestbuy.com/site/samsung-galaxy-s8-64gb-unlocked-midnight-black/5803741.p?skuId=5803741>



BESTBUY REVIEWS

<https://www.bestbuy.com/site/reviews/samsung-galaxy-s8-64gb-unlocked-midnight-black/5803741>



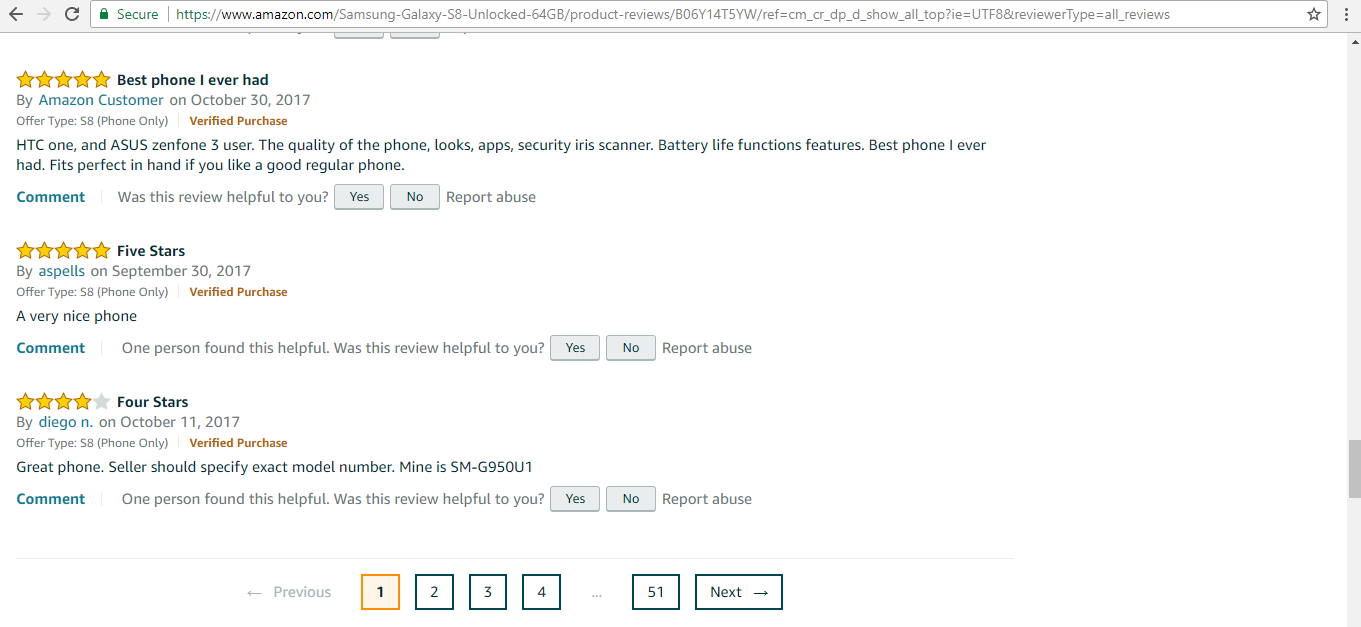
AMAZON HOME PAGE

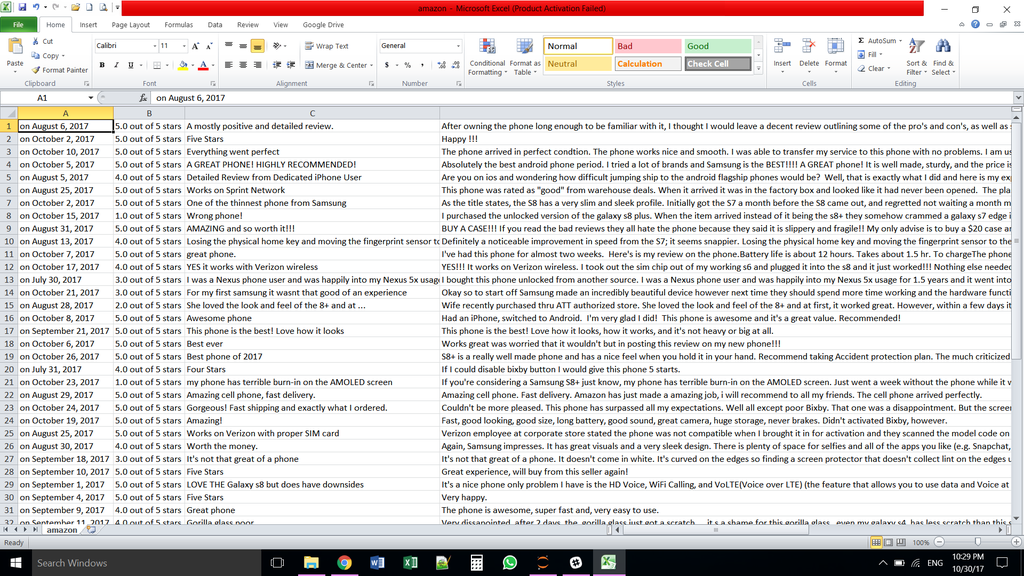
<https://www.amazon.com/Samsung-Galaxy-S8-Unlocked-64GB/dp/B06Y14T5YW>

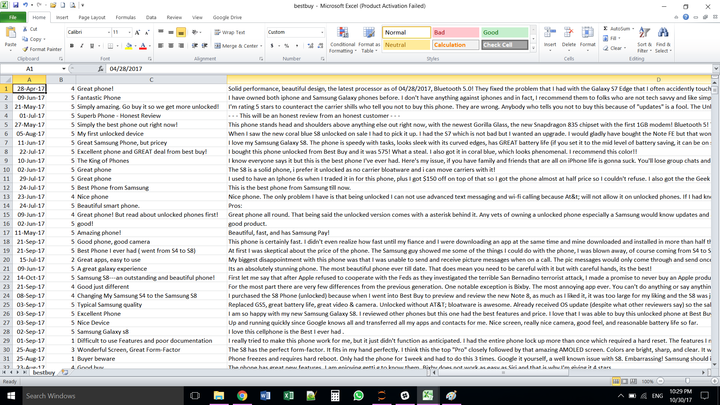


AMAZON REVIEW PAGE

<https://www.amazon.com/Samsung-Galaxy-S8-Unlocked-64GB/product-reviews/B06Y14T5YW/ref=cm_cr_getr_d_show_all?ie=UTF8&reviewerType=all_reviews&pageNumber=1>

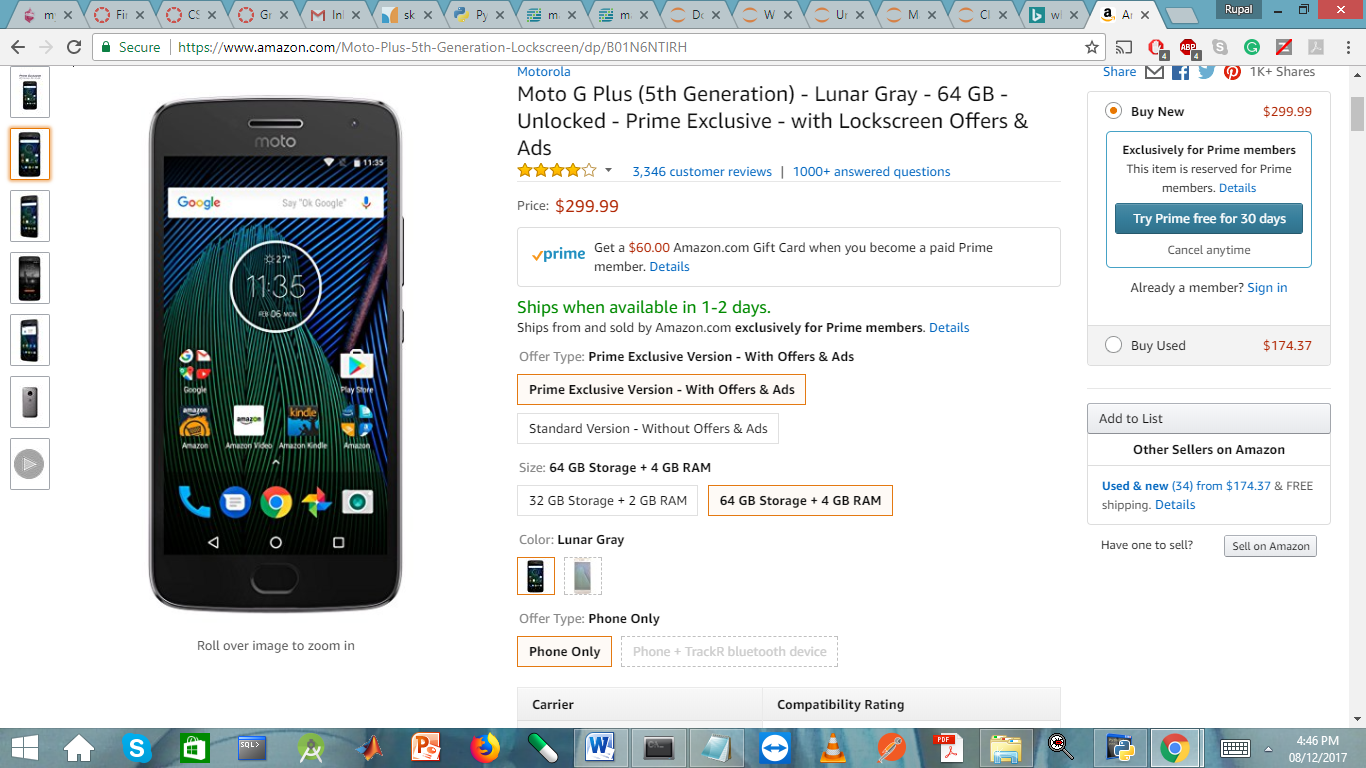






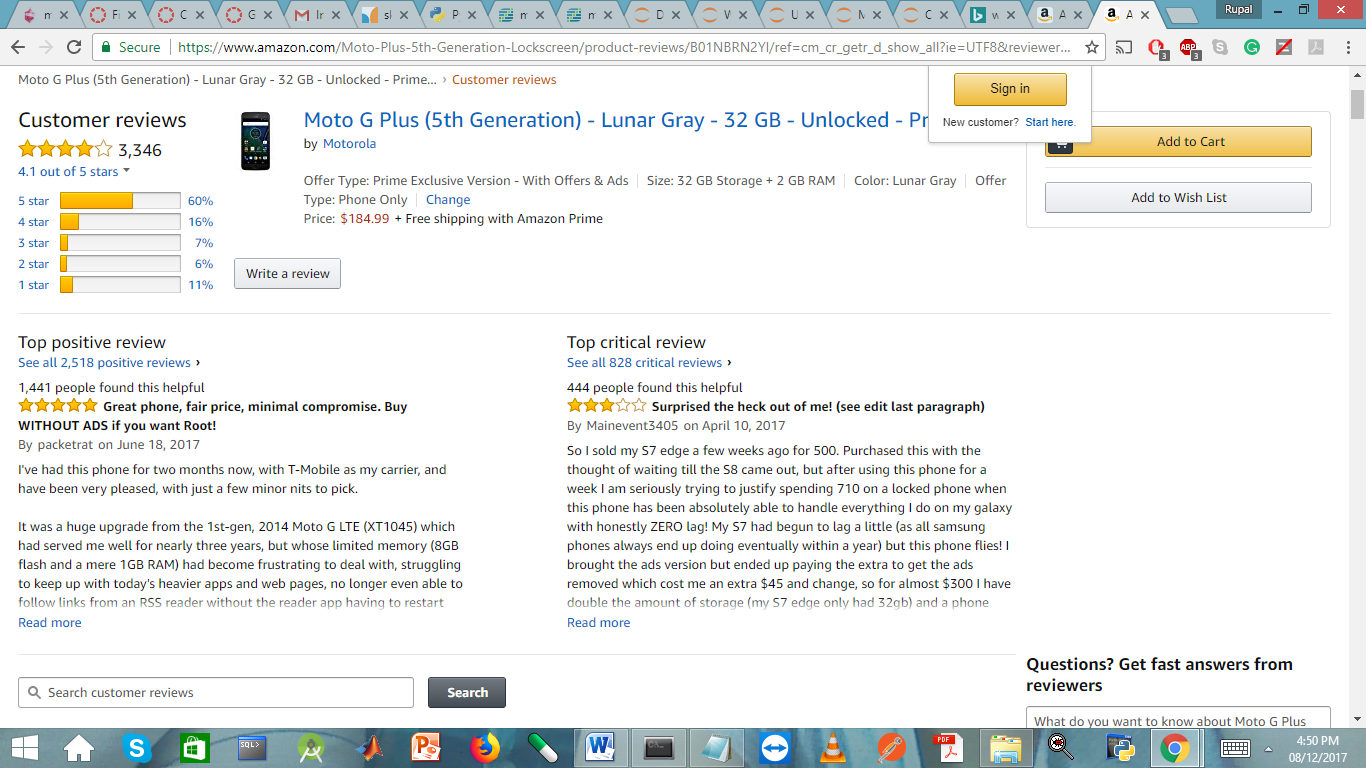
Amazon Moto G Plus

<https://www.amazon.com/Moto-Plus-5th-Generation-Lockscreen/dp/B01N6NTIRH>



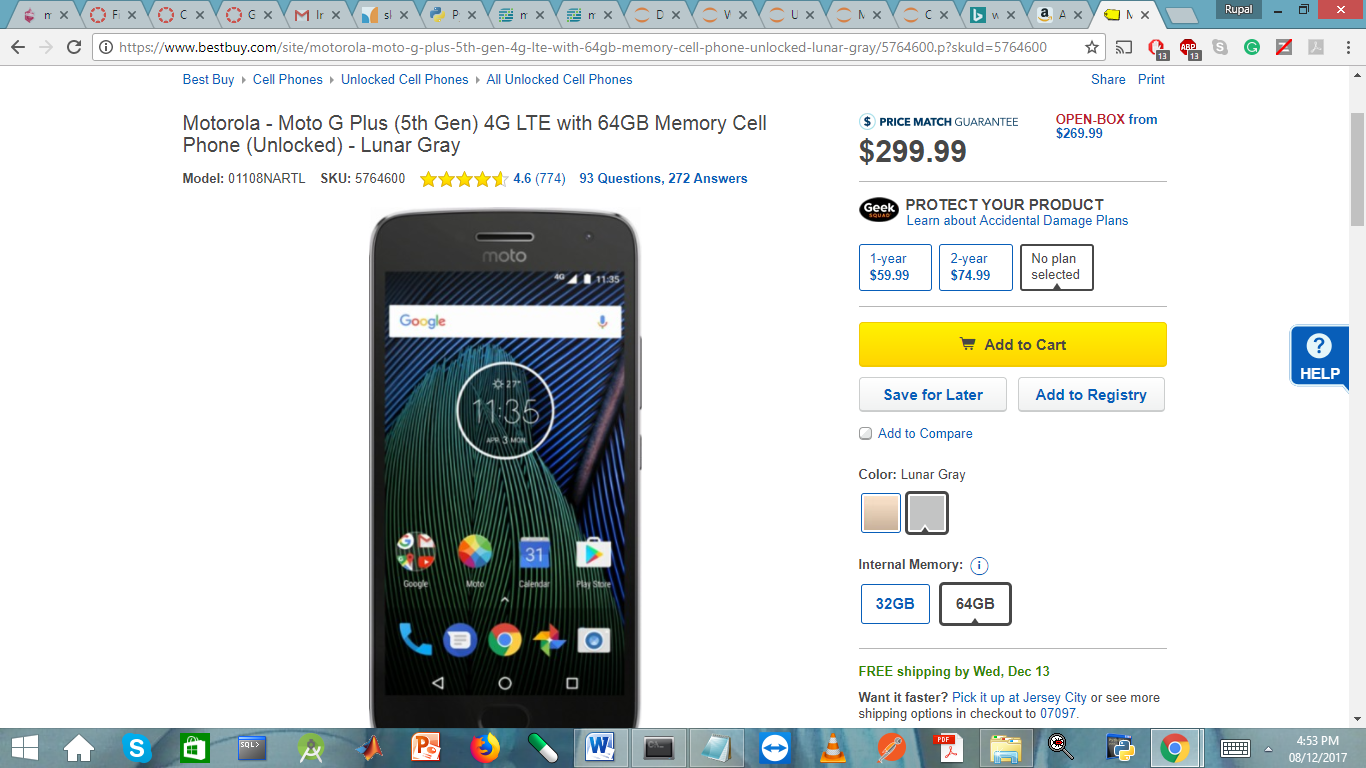
Amazon Reviews of MotoG Plus

<https://www.amazon.com/Moto-Plus-5th-Generation-Lockscreen/product-reviews/B01NBRN2YI/ref=cm_cr_getr_d_show_all?ie=UTF8&reviewerType=all_reviews&pageNumber=1>



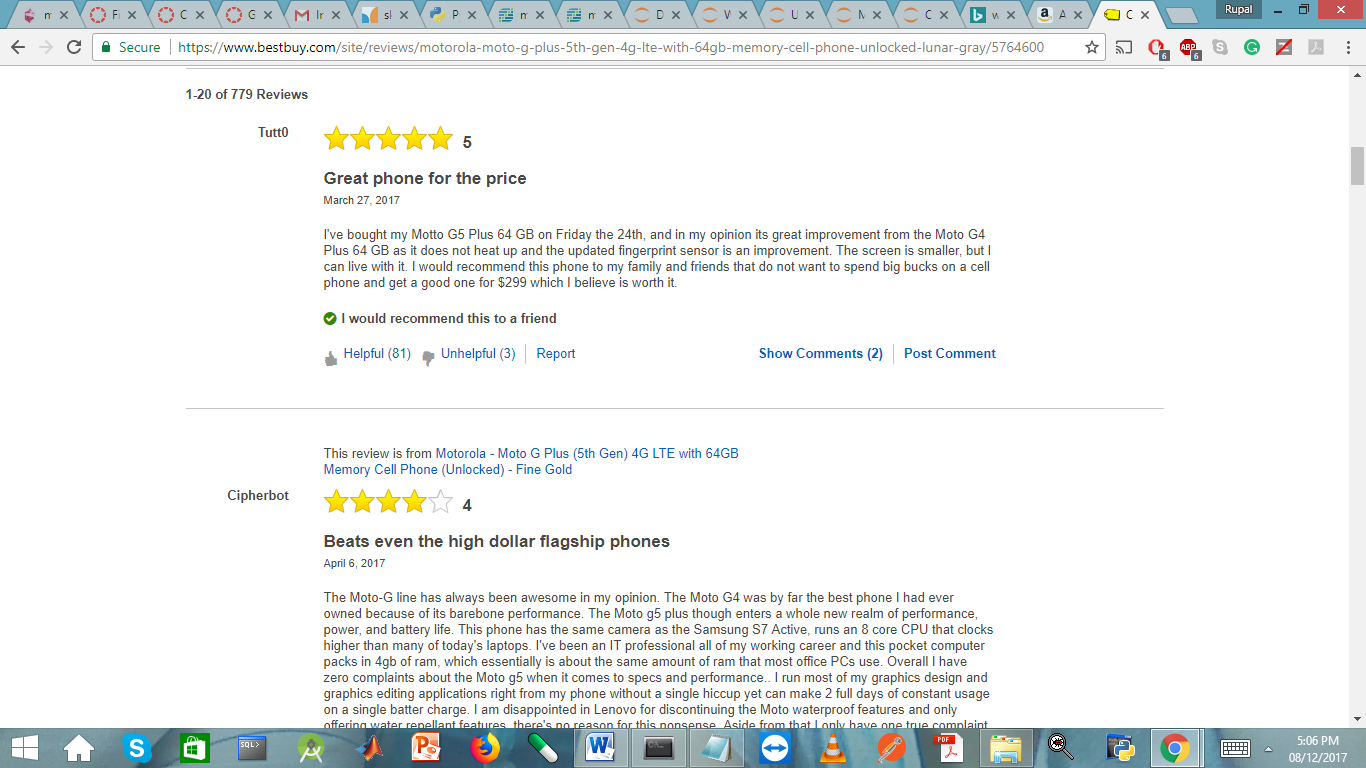
BestBuy Homepage

<https://www.bestbuy.com/site/motorola-moto-g-plus-5th-gen-4g-lte-with-64gb-memory-cell-phone-unlocked-lunar-gray/5764600.p?skuId=5764600>'

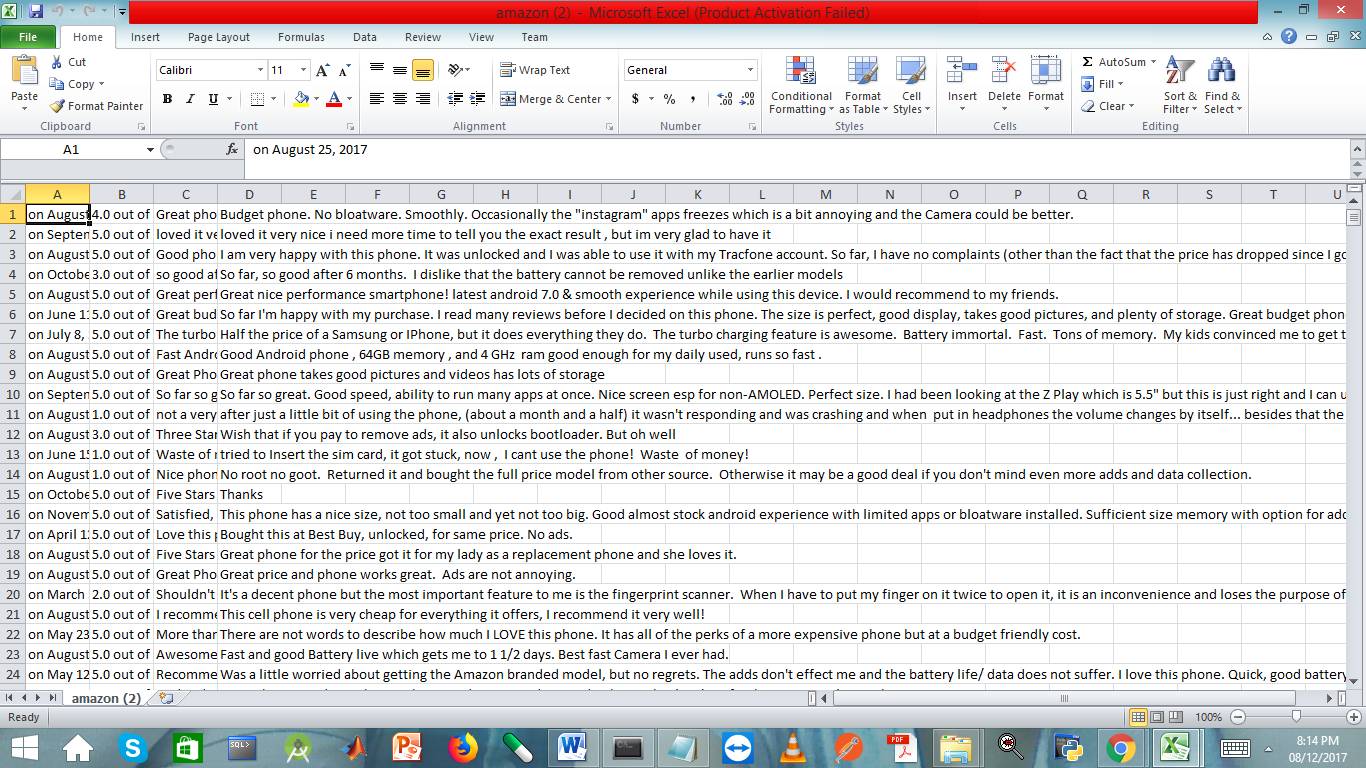


BestBuy Review

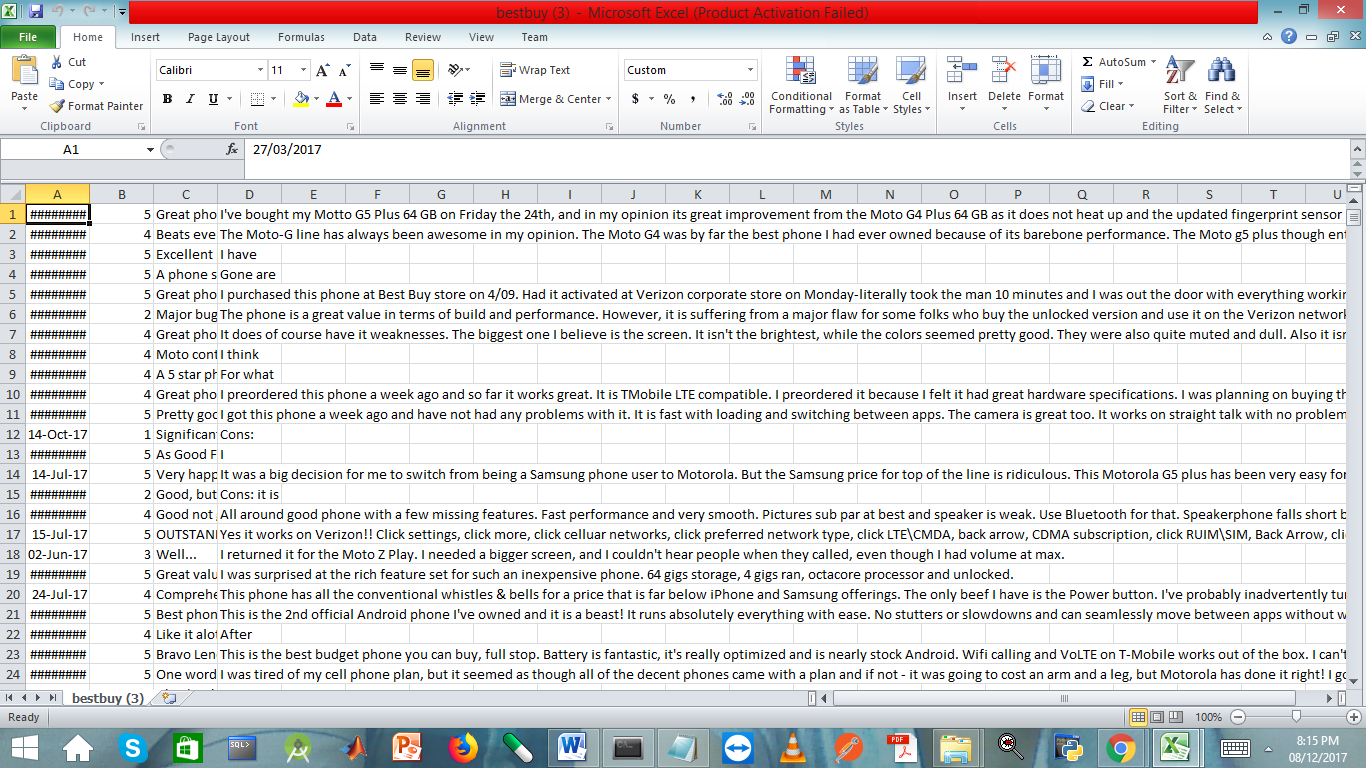
<https://www.bestbuy.com/site/reviews/motorola-moto-g-plus-5th-gen-4g-lte-with-64gb-memory-cell-phone-unlocked-lunar-gray/5764600>



AMAZON



BESTBUY

Our system’s descriptive statistics of data sources done till now are reviews and price. From amazon total reviews scraped are 613 and from bestbuy it is 2111. It will dynamically upgraded is any reviews are added in further. Price on amazon website for MOTO G Plus is 239.99$ and bestbuy price is 299.99. The price will change according to the website.

* **Methodology:**

The main objective of providing the recommendation is done by carrying out steps like web scraping, classification and sentiment analysis. The initial step is scraping the websites for getting the required data relevant to the product. The details regarding the product like its price, delivery, faulty product and customer service was manually labelled. These are the factors regarding the product that will be used for comparison and for providing a recommendation.

During the development of the system different methodologies and algorithms are used like LDA was used for classification, but after its implementation it was realized that it did not helped in producing relevant labels from reviews. Hence we switched to Linear SVC One-Vs-the-Rest strategy was used for multi-label classification. For Linear SVC, initially 70 percentage of dataset was considered under training but the accuracy was around 85 and so we took 80 percentage of dataset as training. On the remaining 20 percentage of testing data set, comparison is done between the predicated labels produced and the manual labels when the match is found, the counter is incremented by one and then divided by total number of reviews which gave accuracy to 91 percentage. Further, mathematical formula was applied to calculate the common factor then followed by calculating the result. Sentiment analysis is performed for each review to calculate the positivity of the review based on the labels mentioned above. Multiplying this positive intensity with the value of the common factor, we get the result of individual label. Then taking individual results we calculate the overall result according to the majority.

**Challenges:**

There were some challenges that we encountered during the project:

* Website structure of the two websites into consideration, that is amazon and best buy was not only a bit complex but also different from each other. So, the review extraction process needed a greater understanding of the structure of both the websites.
* The most challenging part was to traverse the next section of reviews in both the websites. This was because of the use of Ajax in reviews of both the websites. The classes of Ajax were dynamically updating. We overcame this by understanding the patterns in classes and how they were dynamically changing.
* Also, we didn’t use selenium. We had to code everything ourselves, which was an interesting challenge in itself.
* Classification using LDA was a challenge, as it was not able to pick the relevant data according to the review.
* To classify predicted labels.
* Pydict and thesaurus was used previous for extracting label but that was giving some issues. So we had to do manual labelling afterwords.
* Furthermore, we had to modify our code because websites restrict access after a certain number of attempts. So, to avoid that, we had to modify the code.