Software Engineering Mini Project Report on Sentiment Analysis of Mobile Phone Reviews

National Institute of Technology Karnataka, Surathkal

Submitted To:

Dr. K. Chandrasekaran
Ambikesh G
Department of Computer Science and Engineering,
National Institute of Technology Karnataka

Submitted By:

Ankit Jain
Reg No.: 171CO208
anku.171co208@nitk.edu.in
Mobile: 9897320138

Chetan Agarwal
Reg No.: 171CO205
chetanag35@gmail.com
Mobile: 987005327

Abstract

In the recent years, world has seen an immense increase in the usage of social media and smartphones that has led users to comment on various online platforms regarding various different products. Everyone is actively participating in social media to present their opinions. Not only the users are obtaining information from social media but also generating information on it using platforms like blogs, forums, reviews on recent movies and many more. Good response from people awakes desire for the product and encourage a positive attitude of people towards the product. So getting sentiments of the reviews is one of the important aspects for judging the product. Since there can be thousands of reviews for the product, it can not be possible to extract sentiment manually. Sentiment Analysis is a method used to get the sentiments or opinion of users by analyzing the data collected from them through various online platforms. In our work, we aim to mine the users' opinion about different mobile phones. We collected data from various sources like Kaggle, Amazon, etc. about Mobile Phone reviews given by thousands of users. We classified the reviews as positive or negative and found out the words that that contribute to the positivity of data and words that contribute to the negativity of data. All the work of training models is done using two Machine Learning models and the resultant accuracy of both models is compared. Not only accuracy but the time taken by models to train is also compared.

Keywords: Natural language processing, data mining, data preprocessing, sentiment analysis.

Abbreviations:
SVM: Support Vector Machine
NBC: Naive Bayes Classifier
SW: Software
SE: Software Engineering
NLP: Natural Language Processing

2. Introduction

The Web has significantly changed the way people express their opinion and reviews. They can express their views on products at various sites by posting reviews or anything in internet forums, blogs or discussion groups. Now if one desires to shop for a product, he/she will see many product reviews on the net that offer opinions of existing users for that particular product. The corporate might no longer be necessary to conduct surveys by organizing teams or using external consultants in order to search out shopper opinions for its product and those of its competitors. As a result, the users' review content on the net will already offer them such information. However, it's tough for an individual to organize them into usable forms by finding relevant sources, extracting connected sentences with opinions, and summarizing. Thus, finding opinions using Machines is highly required to ease the work of humans.

Sentiment Analysis is used to mine reviews and opinions from the text, database sources and speeches by the help of Natural Language Processing. Sentiment analysis classifies emotions as positive, negative, neutral or any of the other relevant categories. Many people, before buying a particular product, want to see reviews of people who have used it before. These reviews are even very useful for the business organization to assess their products and make them even better to cater to users' needs. In this era of Artificial Intelligence, there are so many approaches available to analyse the Natural Language using various Machine Learning Methods or Models. So, choice of correct model largely affects the results we get for sentiment classification. Here is this project, we have used two Machine Learning Models namely **Naive Bayes' Classifier (NBC)** and **Support Vector Machine (SVM) classifier** to analyse the human written data in the form of various comments/reviews/blogs about different Smartphone models and extract sentiments, learn customer's emotional inclinations and predict the polarity of data. For simplicity, classification has been confined to just positive or negative. Each one of us worked on one model and thus compared the efficiency of both the models. We figured out which model works best and under what parameters. The above-mentioned comparative experiment is done on a large-scale, real-world dataset containing more than 400,000 mobile reviews for different mobile reviews across different mobile brands.

The rest of this document has been designed as follows. In section 3, we have discussed the related work and the experiments performed for the project. We have given the idea of experiment design, the process followed for experiment, as well as the background information of the relevant algorithms used. In section 4, we have given a concise account of the results obtained on performing the various experiments. Finally, in section 5, we have drawn some conclusions and propose future work i.e how the projected can be extended further.

3. Approaches and Methods

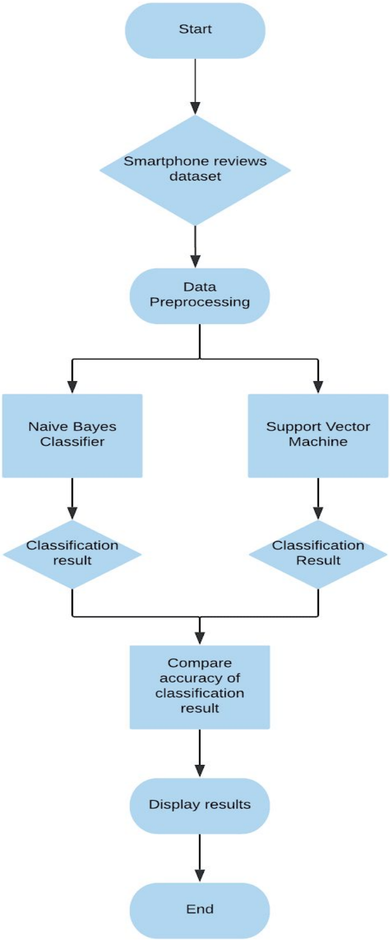
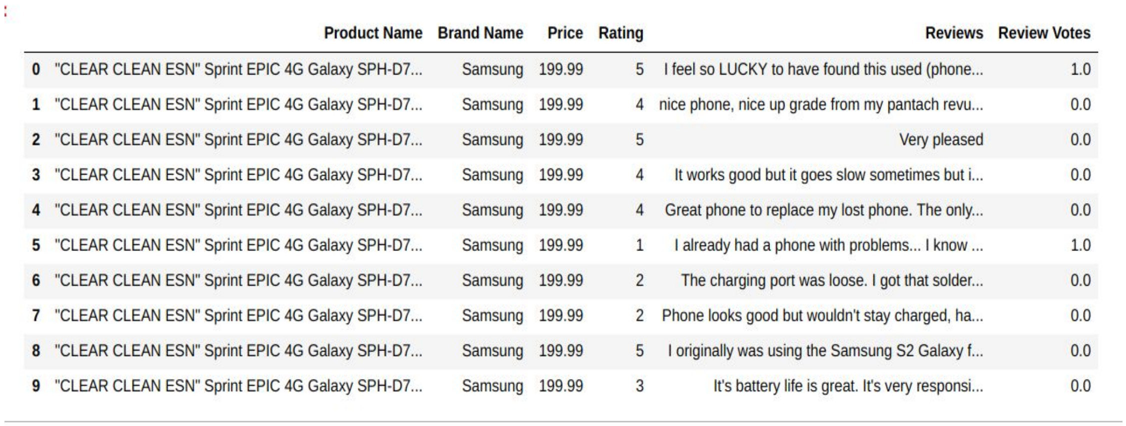


Fig.1 The Detail System Design For Sentimental Analysis

Our proposed work is shown in figure 1. We started by collecting dataset containing more than 400,000 smartphone reviews across various brands. After that cleaning and preprocessing of the dataset was done. Cleaning and Preprocessing is a crucial step for increasing accuracy of the model which includes dropping irrelevant columns and extracting out the information most suitable for a machine to understand and to make it suitable for fitting into our Machine Learning Models. The features extracted from data are trained on NBC and SVM classifier. On the resulting model, we fed test data which gave us sentiments of the test data. The classification result from both the models is thus compared to analyze the efficiency of the models.

3.1 Data Cleaning and Preprocessing

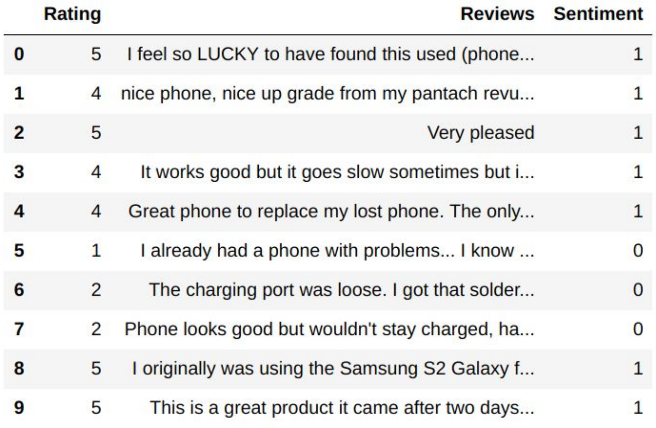
We started with collecting dataset and got dataset having more than 400,000 mobile reviews for different mobile models across different mobile brands. The dataset is a collection of real-world reviews in natural language of smartphones that are sold on Amazon.com to find out information with respect to the reviews, price, ratings.. The initial dataset is shown in figure 2.



	Product Name	Brand Name	Price	Rating	Reviews	Review Votes
0	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	5	I feel so LUCKY to have found this used (phone...	1.0
1	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	4	nice phone, nice up grade from my pantach revu...	0.0
2	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	5	Very pleased	0.0
3	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	4	It works good but it goes slow sometimes but i...	0.0
4	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	4	Great phone to replace my lost phone. The only...	0.0
5	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	1	I already had a phone with problems... I know ...	1.0
6	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	2	The charging port was loose. I got that solder...	0.0
7	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	2	Phone looks good but wouldn't stay charged, ha...	0.0
8	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	5	I originally was using the Samsung S2 Galaxy f...	0.0
9	"CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D7...	Samsung	199.99	3	It's battery life is great. It's very responsi...	0.0

Fig.2 Raw Dataset

The reviews in the dataset are scored from 1 to 5 points. In order to reduce the subjective difference of users on their degree of sentiments, reviews with ratings 4 and 5 points have been combined into the positive review “1”, reviews with rating 1 and 2 points as negative review “0” and reviews with rating 3 points have been dropped as they do not convey any relevant information about the sentiments of a user. We also dropped the insignificant columns like Product Name, Brand Name, Price and Votes. The data set after this cleaning operation is shown in figure 3.



	Rating	Reviews	Sentiment
0	5	I feel so LUCKY to have found this used (phone...	1
1	4	nice phone, nice up grade from my pantach revu...	1
2	5	Very pleased	1
3	4	It works good but it goes slow sometimes but i...	1
4	4	Great phone to replace my lost phone. The only...	1
5	1	I already had a phone with problems... I know ...	0
6	2	The charging port was loose. I got that solder...	0
7	2	Phone looks good but wouldn't stay charged, ha...	0
8	5	I originally was using the Samsung S2 Galaxy f...	1
9	5	This is a great product it came after two days...	1

Figure 3. Cleaned Dataset

Next comes Data Preprocessing. Data preprocessing is a data mining technique that includes transforming of raw data i.e. data which does not convey any information to computer into a machine-understandable format. Real-world data is mostly inconsistent, incomplete, and lacks some particular behaviors or trends, and is likely to have many errors. Data preprocessing is an efficient method for resolving such issues. The steps involved in our Data Preprocessing are summarised as follows:

- Get the most suitable libraries: We used Natural Language Processing Toolkit (NLTK) and SKLearn Library for data preprocessing.
- See the missing values in Dataset: It is very important to check for null values in the dataset in order to manage data efficiently. If the null values are not removed then we can end up getting wrong inferences about the data.
- Removal of Punctuation and Special Characters: All the non-alphabet symbols render no information the data, instead, they interfere with the meaning of data as perceived by machine. So, it is necessary to remove punctuation marks and special characters.
- Lowercase: Convert all the letters to lowercase so that the machine does not consider words in a different case with different meanings.
- Tokenization: It is the breaking of text into meaningful words, symbols and/or elements called tokens.
- Stop Words Removal: Words like “to”, “a”, “the”, “is”, “that”, etc that do not provide any significant information to the machine are stopped words. It is better to remove them to increase the processing speed as these words take up a huge count in raw data.
- Lemmatizing: It is the process of analyzing the words morphologically,i.e., relating words to their different forms.

- Stemming: It involves removal of the verb forms from words and getting the root word.

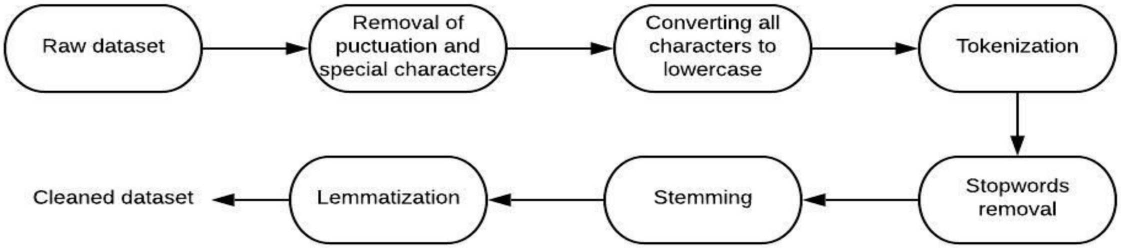
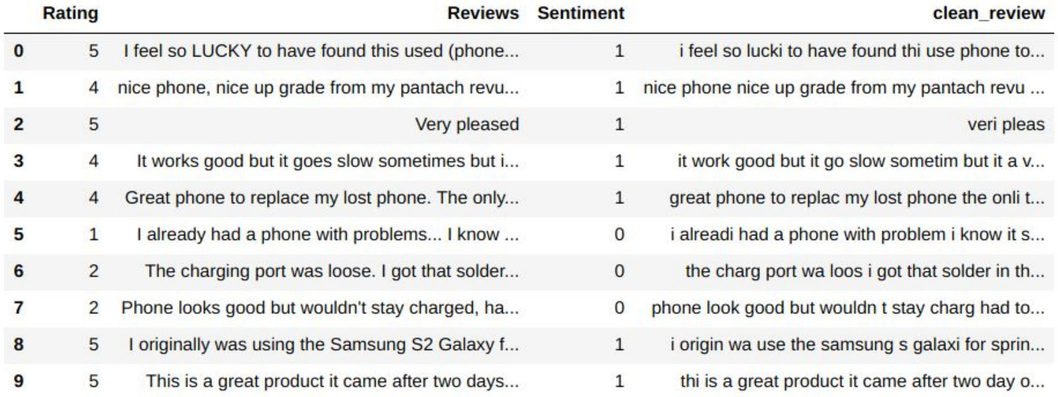

Raw dataset
Removal of puctuation and special characters
Converting all characters to lowercase
Tokenization
Stopwords removal
Stemming
Lemmatization
Cleaned dataset

Figure 4. Data Preprocessing

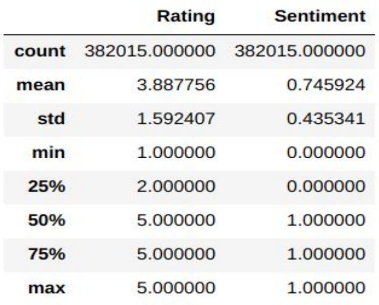


	Rating	Reviews	Sentiment	clean_review
0	5	I feel so LUCKY to have found this used (phone...	1	i feel so lucki to have found thi use phone to...
1	4	nice phone, nice up grade from my pantach revu...	1	nice phone nice up grade from my pantach revu ...
2	5	Very pleased	1	veri pleas
3	4	It works good but it goes slow sometimes but i...	1	it work good but it go slow sometim but it a v...
4	4	Great phone to replace my lost phone. The only...	1	great phone to replac my lost phone the onli t...
5	1	I already had a phone with problems... I know ...	0	i alreadi had a phone with problem i know it s...
6	2	The charging port was loose. I got that solder...	0	the charg port wa loos i got that solder in th...
7	2	Phone looks good but wouldn't stay charged, ha...	0	phone look good but wouldn t stay charg had to...
8	5	I originally was using the Samsung S2 Galaxy f...	1	i origin wa use the samsung s galaxi for sprin...
9	5	This is a great product it came after two days...	1	thi is a great product it came after two day o...

Fig.5 Processed and Cleaned data

Let's consider the first review. All special characters and punctuations are removed (here '(' is a special character). The resulting data is converted to lowercase(LUCKY is converted to lucky). Every word in the review is then tokenized. Stopped words are then removed from tokenized words. After all this, stemming and lemmatization is done on resulting data. This gives us cleaned and processed dataset as given in Fig.5

The statistical features of processed data are shown in figure 6. After dropping irrelevant and missing data from the dataset around 380,000 reviews are left out with us. Minimum Rating is 1 and min sentiment is 0.00. 25% means when total data is divided into 4 parts then the last rating of the first part will have rating 2 and sentiment 0. Max indicates that maximum rating is 5 and max sentiment is 1.00. Mean of all rating is 3.88 which implies that we have a dataset that is more inclined towards positivity. Mean of sentiment is 0.74 which justifies the same.



	Rating	Sentiment
count	382015.000000	382015.000000
mean	3.887756	0.745924
std	1.592407	0.435341
min	1.000000	0.000000
25%	2.000000	0.000000
50%	5.000000	1.000000
75%	5.000000	1.000000
max	5.000000	1.000000

Fig.6 Statistical Description of Processed data

We used the bag of words model for feature extraction from the cleaned and preprocessed dataset. The bag-of-words model is a simple representation used in Natural Language Processing and information retrieval. In this model, a text is represented as the bag of its words, disregarding the grammar and even the word order but keeping multiplicity of words. The 20,000 most frequently occurring words are used as the vocabulary of data. Firstly, the Unigram vector of reviews is made i.e. in our vocabulary all the words are single, there is no word pair. Similarly, Bigram vector of reviews is also made i.e. in the vocabulary all the words are either single or pair of two words. Among all possible words from the dataset, top 20,000 occurring words are taken into account for Bigram vocabulary. Only words that occur at least 5 times contribute to the vocabulary. Each review is turned to a vector with 20000 entries corresponding to the 20000 words of the vocabulary. If the word corresponding to position 'i' in vocabulary is present in the review, then the 'ith' entry in the vector is 1 otherwise it is 0. Then data is split into training set and test set. Now we will analyze both the Machine Learning Models used for this work namely NBC and SVM.

3.2 Naive Bayes' Classifier

Naive Bayes Classifier uses Bayes theorem in probability and predict unknown class. It is one of the most straightforward and simple supervised learning algorithms. Naive Bayes classifier is the accurate, fast and reliable algorithm. NVC has high accuracy and speed on large datasets. It uses the assumption that the effect of a particular feature is independent of other neighbouring features. It does not give importance to order of different words in the text. This assumption simplifies computation, and that's why it is considered as naive. So it fails to work when the order of data may be significant. For instance, it considers "not good, bad" and "not bad, good" as same.
After preprocessing dataset, each review is classified as followed:

$$p(positive|review) \;=\; (p(review|positive) * p(positive)) \div p(review)$$

And then, calculate $p(review|positive)$. Thus, compare the probabilities of each class and use the highest rank as the class for the review. The NBC is imported from SKLearn library where the predefined code for NBC is available. The model is trained for the Unigram train data and classification is done on test data. The words that contribute most to the positivity and negativity of reviews are analyzed. The confusion matrix is also obtained and is shown in the Results section. Then, the model is trained for Bigram data and classification as positive or negative is done on test data. Again, the bigram words that contribute most to the positivity or negativity of review are analyzed and the confusion matrix is also obtained.

3.3 Support Vector Machine

Support Vector Machine (SVM) is one of the supervised machine learning methods to train a classifier. It outputs optimal hyperplane which separates training data in different regions and when test data is feed to it, it finds out the region that best suits the data.SVM is mostly used in the classification problems. In this algorithm, we used to plot each data item as a point in n-dimensional space (where n is a number of features we have) with the value of each feature being the value of a particular coordinate. SVM can have various type of kernel. Kernel decides the shape of the divider, it can be Linear, radial, higher degree polynomial, and many others.

SVM has various hyperparameters of which kernel is the that we should choose with great accuracy as it changes the accuracy of the model to a great extent. Another hyperparameter is C that stands for regularization. It tells the SVM optimization how much you want to avoid misclassifying each training example. For larger values of C, the optimization will choose a smaller margin hyperplane if that hyperplane does the desired job of getting all the training points classified correctly,i.e, good training accuracy. Whereas, a very smaller value of C will cause the optimizer to look for a larger margin separating hyperplane, even if that hyperplane misclassified more points. The job of regularization is to eradicate overfitting or underfitting of data. Another hyperparameter affecting our model is gamma, Gamma values decide how far off points from the hyperplane should be considered. Low gamma value means range where points affect hyperplane is large whereas high gamma value indicates that the range within which points affects hyperplane is low. Margin is another characteristic of SVM. The margin is a separation of the line to the closest class points. A good margin is the one where this separation is larger for both the classes. A good margin allows points to be in their respective classes without crossing to other class.

In this project, we had imported SVM from the sklearn library. Here SVC(Support Vector Classifier) for SVM is used to classify dataset and generate a classifier. SVC is one type of SVM classifier. The major hyperparameters needed to train is gamma, regularization, and kernel. Since we can't say that hyperplane is linear. So, the kernel needs to be 'rbf' (radial bias function) and this is also the default kernel in SVM. The second one is C which is the major characteristic to decide the accuracy of the model. Various values of C is tried and curves are plotted to find the best value of C for the best accuracy of the model. Best accuracy of the model doesn't mean that test accuracy or train accuracy should be very good. Instead, train and test accuracy both should be very good. If train accuracy is good and test accuracy is not so good then this implies that our model is overfitted and we need to regularize it. So, to reduce overfitting, we will then decrease the value of C and hence making our model less overfitted. Conversely, if we get model were both training data accuracy and test data accuracy are lower, then our model is under fitted. So, here we need to increase the value of C to make it more regularized and hence bring our model from under fitted zone to best-fitted zone. After trying a lot of values, it comes out that model is giving best accuracy for C=500 for bigram model whereas, for unigram model, it comes out to be around 450 for our dataset of training and test data.

4. Results:

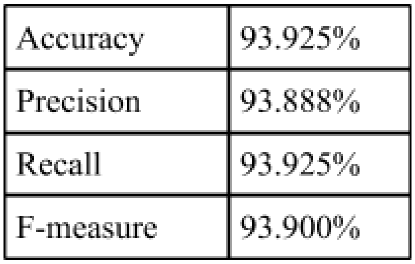
Performance of each classification model is evaluated by using four indexes: Accuracy, Precision, Recall, and F-measure. The followings are equations for indexes:

$$accuracy = (TP + TN) \div (TP + TN + FP + FN)$$
$$recall = TP \div (TP + FN)$$

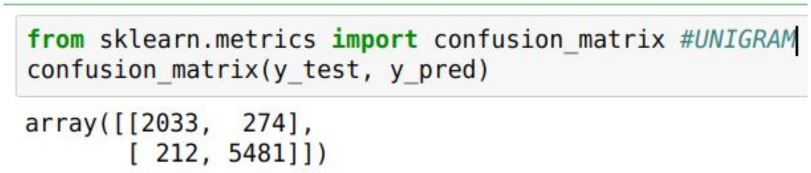
$$precision\ = TP \div (TP + FP)$$
$$f - measure = (2 * precision * recall) \div (precision + recall)$$
$$where\ TP = True\ Positives,\ FP = False\ Positives,\ TN = True\ Negatives,\ FN = False\ Negatives$$

4.1 Naive Bayes Classifier

The dataset is split into two sets, namely, training set and test set. The training set is 80% of the total dataset whereas test set is 20% of the dataset. On training the NBC against Unigram vector of reviews, the accuracy of prediction on the test data came out to be 93.925% which is quite a good score. The recall, precision and f-measure are 93.925%, 93.888%, 93.900% respectively. The confusion matrix for Unigram classification is given in the is given in fig. 8. Confusion matrix C is such that C(i,j) is equal to the number of observations known to be in group i but predicted to be in group j. Thus in binary classification, count of the true negatives is C(0,0), false negatives are C(1,0), true positives is C(1,1) and false positives is C(0,1). This shows that 2033 reviews are correctly classified as negative, 5481 are correctly classified as positive, 274 are wrongly classified as positive and 212 are wrongly classified as negatives.



Accuracy	93.925%
Precision	93.888%
Recall	93.925%
F-measure	93.900%

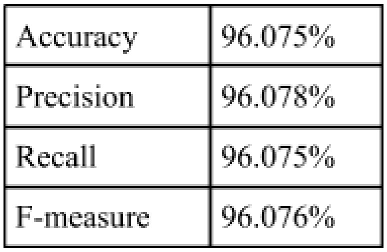


```
from sklearn.metrics import confusion_matrix #UNIGRAM
confusion_matrix(y_test, y_pred)

array([[2033,  274],
       [ 212, 5481]])
```

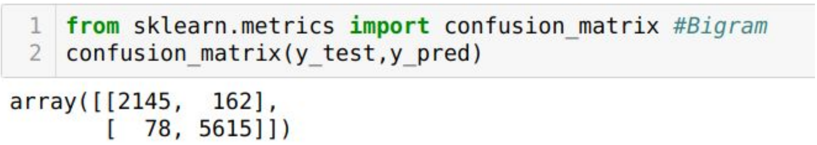
Figure 8. Confusion Matrix for Unigram NBC

Figure 7. Performance of Accuracy, Precision, Recall and F-measure for Unigram NBC

Coming to the Bigram model, on training the NBC against Bigram vector of reviews, the accuracy of prediction on the test data came out to be 96.075% which is even better than Unigram NBC. The recall, precision and f-measure are 96.075%, 96.078%, 96.076% respectively. The confusion matrix for Bigram classification is given in the is given in fig. 9. This shows that 2145 reviews are correctly classified as negative, 5615 are correctly classified as positive, 162 are wrongly classified as positive and 78 are wrongly classified as negatives. This clearly shows that **Bigram worked better than Unigram model in case of NBC**.



Accuracy	96.075%
Precision	96.078%
Recall	96.075%
F-measure	96.076%



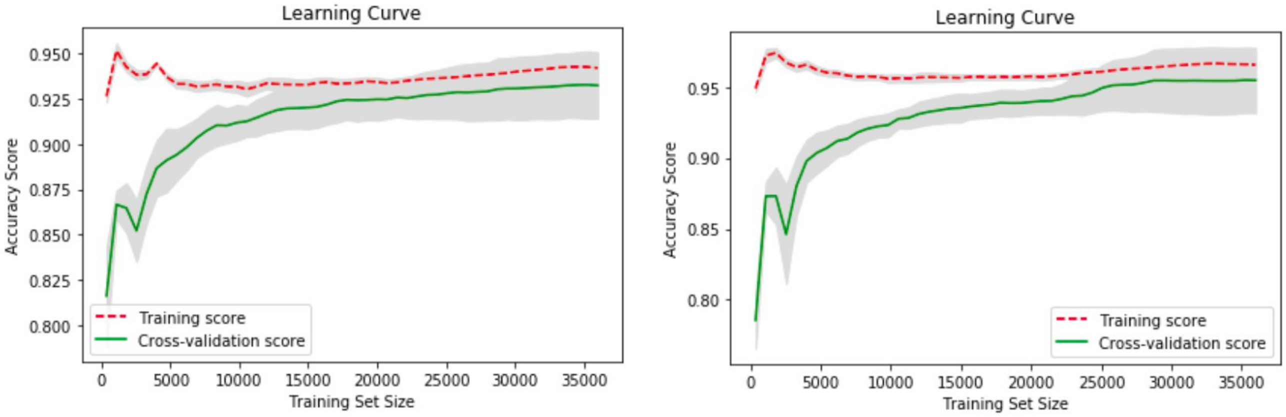
```
from sklearn.metrics import confusion_matrix #Bigram
confusion_matrix(y_test,y_pred)

array([[2145,  162],
       [  78, 5615]])
```

Figure 10. Confusion Matrix for Bigram NBC

Figure 9. Performance of Accuracy, Precision, Recall and F-measure for

Bigram NBC

A plot of Accuracy score vs Training set size for both Unigram and Bigram is shown in figure 8. It tells how better the model gets at predicting with increasing training set size.

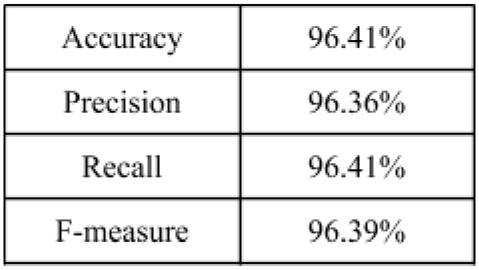


(a) **(b)**

Figure 11. Learning Curves for (a) Unigram NBC and (b) Bigram NBC

4.2 Support Vector Machine

In SVM model, the dataset is split into two sets, namely, training set and test set. Training set is 80% of the total dataset whereas test set is 20% of the dataset. We are trying to find out accuracy of our model on mobile reviews dataset using unigram and bigram model.We are also trying to find out the best hyperparameters for the model to best fit the training data. The key difference between parameters and hyperparameters is that parameters are trained by model to best fit the dataset whereas hyperparameters is decided by the model developers to best train the parameters to give the maximum accuracy of the model on dataset. Here we have hyperparameters like kernel, C, gamma and many more. But here, we are focussing on just kernel and C. Kernel stands for type of curve that SVM is going to plot finally. Here we are using default value of kernel,i.e., rbf. RBF stands for radial bias function. We are trying out different values of C any finally choosing the value that best suits to give maximum accuracy to our dataset. Rest all hyperparameters are set as default values.



Accuracy	96.41%
Precision	96.36%
Recall	96.41%
F-measure	96.39%

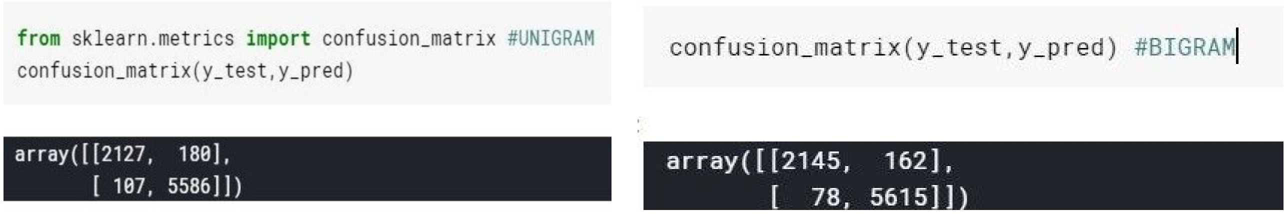
Fig.12 Accuracy, Precision, Recall and F-measure of unigram SVM model on mobile reviews

We find out that accuracy of SVM model under unigram vectorization is 96.41% on test data and about 98% on training accuracy. This clearly implies that our model is neither overfitting the dataset and neither underfitting the dataset. Then we tried to find out precision, recall and F-measure of our model. We find out that the precision, recall and F-measure of our unigram model are 96.36%, 96.41% and 96.39% respectively. Here we find out that maximum accuracy of the model is attained at C=500.

Accuracy	97%
Precision	96.99%
Recall	97%
F-measure	96.98%

Fig.13 Accuracy,Precision,Recall and F-measure of Bigram SVM model

When we trained our model on bigram vectorization, we get test accuracy of about 97% which is quite high. Whereas the training accuracy of the model, in this case, is 98.5%. When tried to calculate precision, recall and F-measure we got 96.99%,97%,96.98% respectively which is evident that our model is giving accuracy to about 97-98 and is one of the best model to be applied for training in this dataset. Here we find out that the maximum value of accuracy is attained at C=400. The confusion matrix for Unigram SVC classification is given in the is given in fig. 14 (a). This shows that 2127 reviews are correctly classified as negative, 5586 are correctly classified as positive, 180 are wrongly classified as positive and 107 are wrongly classified as negatives. The confusion matrix for Bigram SVC classification is given in the is given in fig. 14 (b). This shows that 2127 reviews are correctly classified as negative, 5586 are correctly classified as positive, 180 are wrongly classified as positive and 107 are wrongly classified as negatives.



```
from sklearn.metrics import confusion_matrix #UNIGRAM
confusion_matrix(y_test,y_pred)
```

```
array([[2127,  180],
       [ 107, 5586]])
```

```
confusion_matrix(y_test,y_pred) #BIGRAM
```

```
array([[2145,  162],
       [  78, 5615]])
```

(a) **(b)**

Figure 14. Confusion Matrix for (a) SVC Unigram and (b) SVC Bigram

Based on the works done in this project it can be inferred that SVM works better than NBC in this case but alongside it is also clear that SVM is way more slower than that of NBC.

5. Discussion

In this paper, we demonstrated how we can use Machine Learning to get sentiments of users over web data. For this, we used two Machine Learning algorithms namely Naive Bayes algorithm and Support Vector Machine algorithm. The dataset was obtained from famous websites which contains 400,000+ smartphone reviews across different brands. All positive and negative reviews are collected and compared. The proposed algorithms have successfully filtered the positive and negative reviews counting. Then this counting is used to get the confusion matrix results, which is showing the overall result performance of the models. A comparative analysis for both NBC and SVM is shown in figure 15. It is clear that in this case, SVM worked out better than NBC.

But accuracy comes with the cost of time. SVM takes way more time to train and test than that of NBC model.

Model	Accuracy(%)	Precision(%)	Recall(%)	F-measure(%)
NBC Unigram	93.25	93.89	93.92	93.90
NBC Bigram	96.07	96.08	96.07	96.07
SVM Unigram	96.41	96.36	96.41	96.39
SVM Bigram	97.00	96.99	97.00	96.98

Figure 15. Comparing Performance of Accuracy, Precision, Recall and F-measure for each model

The experimental results show that:

- On the classification algorithm link, the SVM method is proved to be better than Naive Bayes method in this case
- On feature representation, N-Gram is applied (N=2, Bigram) for the best result.

As per the increase in technology, online shopping websites have become a huge market for consumers, and these websites also help in giving opinions regarding the articles bought. Our proposed algorithm is very useful in analysing the positivity and negativity of the consumers and improving the quality factors of the articles.

We can further extend our work to the application of neural network in collaboration with NLP. The neural network is basically a network of neurons and each neuron reviews inputs from neurons in previous layers. Neuron then tries to best fit the input to get the desired output. Each neuron can be assumed as neuron of the brain and as each neuron gets signals from other, it process signals in its own way to give out another signal. In the same way, we can feed raw data as input to the neural network and then neurons will process input in its own way to get maximum accuracy and pass out the result to next neurons. This is the basic idea behind neural networks. We can apply the neural network in on this dataset by feeding the initial layer with cleaned data as input and then neurons will set parameters accordingly to get maximum accuracy of the model. Here we can assume that number of layers and number of neurons in each layer is hyperparameters as this has to be set by us and we need to check that which hyperparameters values gives out the best accuracy of the model.

This project can further extended to figure out the reason for giving positive or negative reviews. In a way that if a person had given negative review then software should figure out why he don’t like phone whether mobile battery drains fast or mobile is defective or any other reason. Similarly, if one had given positive review then why he had given positive reviews whether price is low as compared to other companies or he liked the quality of camera or any other reason. This extension of software will help companies to figure it out that which side of the company is lagging behind the expectation of the customer and helps them to improve them upto expectation of the customers.

Also, there can be one other view to extention of this project. One can extend this project to develop software that first checks that given review of the mobile whether rating given by the user is justified by the reason given by the user in his/her review. This helps to remove or modifies reviews that are either exaggreated or fake. This helps to given a really well trained best suited model for sentimental analysis.

Also, there is one another scope for extention of this software. One can extend it to not just positive or negative reviews but can check percentage of positivity and negativity in the reviews given by the user. Lets assume someone had written "Camera quality is not upto mark but graphics and sound is great. Overall satisfied with the product". Then this project can tell that this is a positive review but extended software will tell that the review is 30% negative and 70% positive. This gives accurate jugdement of the review.

One other extension of the project can be estimation of rating out of 5 based on reviews. This will help to even find out that whether a given review is a neutral review.

6. References:

[1] Varghese,R., Jayasree, M., "A Survey on Sentiment Analysis and Opinion Mining," IJRET: International Journal of Research in Engineering and Technology, Volume: 02 Issue: 11, October 2013.

[2] Periakaruppan Sudhakaran, Joan Lu.,"Classifying product reviews from balanced datasets of Sentiment Analysis & Opinion Mining" Published in 2014 6th International Conference on Multimedia, Computer Graphics and Broadcasting IEEE.

[3] Vohra, S. M. and Teraiya, J. B., "A Comparative Study of Sentiment Analysis Techniques," Journal of Information, Knowledge and research in Computer Engineering Volume – 02,Issue –02, October 2013.

[4] Wei Yen Chong, Bhawani Selvaretnam, and Lay-Ki Soon, "Natural Language Processing for Sentiment Analysis" Published in 2014 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology. IEEE

[5] Anchal Kathuria, Dr. Saurav Upadhyay, "A Novel Review of Various Sentimental Analysis Techniques" International Journal of Computer Science and Mobile Computing IJCSMC, Vol. 6, Issue. 4, April 2017, pg.17 – 22.

[6] Tanvi Hardeniya, Dilipkumar A. Borikar "Dictionary Based Approach to Sentiment Analysis- A Review" IJAEMS: International Journal of Advanced Engineering, Management and Science, Volume-02 Issue-5, May 2016.

[7] R.Nithish, S.Sabarish, M.Navaneeth Kishen. "Ontology based Sentiment Analysis for mobile products using tweets" Published in 2013 IEEE Fifth International Conference on Advanced Computing (ICoAC).

[8] Lu Lin, Jianxin Li, Richong Zhang,Weiren Yu and Chenggen Sun,"Opinion Mining and Sentiment Analysis in Social Networks: A Retweeting Structure-aware Approach" Published in 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing.

[9] Deepak Singh Tomor, Pankaj Sharma “A Text Polarity Analysis Using Sentiwordnet Based an Algorithm” IJCSIT: International Journal of Computer Science and Information Technologies, Volume-7(1), 2016.

[10] Gaurav Dubey, Ajay Rana, Naveen Kumar Shukla, "User reviews data analysis using opinion mining on web" Published in 2015 1st International Conference on Futuristic trend in Computational Analysis and Knowledge Management, IEEE.

[11] Haruna Isah, Paul Trundle, Daniel Neagu, “Social Media Analysis for Product Safety using Text Mining and Sentiment Analysis ”.

[12] Taysir Hassan A. Soliman, Mostafa A. Elmasry,"Utilizing support vector Machines in mining online customer" Published in Reviews 2012 IEEE.

[13] Liliana Ferreira,"A Comparative Study of Feature Extraction Algorithms in Customer Reviews" Published in The IEEE International Conference on Semantic Computing.

[14] Nur Azizah Vidya, Mohamad Ivan Fanany, Indra Budi, “Twitter Sentiment to Analyze Net Band Reputation of Mobile Phone Providers” 3rd Information Systems International Conference, Procedia Computer Science 72, 2015, 519-526.

[15] Mukwazvure, A., K.P. Supreethi, “A Hybrid Approach to Sentiment Analysis of News Comments. Reliability,” In 4th International Conference on Infocom Technologies and Optimization (ICRITO)- Trends and Future Directions, IEEE, 2015.

[16] Efthymios kouloumpis, Theresa Wilson, Johanna Moore, “Twitter Sentiment Analysis: The Good the Bad and the OMG” Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media.

[17] Bing Liu, “Sentiment Analysis and Subjectivity”, Handbook of Natural Language Processing, Second Edition, 2010.

[18] Rekha, Dr. Williamjeet Singh, “Sentiment Analysis of Online Mobile Reviews”, In International Conference on Inventive Communication and Computational Technologies (ICICCT), 2017.

[19] May Thanda Theint Aung, Aye Aye Kyaw, “Sentiment Analysis of The Smartphone Product Reviews”, Proceedings of Academicsera 24 th International Conference, 2018.