Mini Project Review

Sentiment Analysis of Amazon Customer Reviews.

Since 1962

Vivekanand Education Society's Institute of Technology (Mumbai University)

Domain :- Machine learning
Mentor :- Mrs. Vidya Zope
Group Members :- 1) Paras Patil (D7B-41)
2) Kaustubh Keny (D7B-26)
3) Kapish Madhwani (D7B-32)
4) Vishesh Mittal (D7B-35)

Index

Introduction For The Project

Our project is centered on Sentiment Analysis of Amazon Customer reviews. Sentimental analysis is the process of computationally identifying and categorizing opinions in a piece of text.

Furthermore, it focuses on classifying the polarities in user reviews using various Machine Learning Algorithms.

Motivation For The Project

- The domain of NLP and its application is huge and there is an enormous amount of data on web applications where people are provided a platform to express their views, thus it becomes necessary to take benefit of it.

- Machine learning is a booming industry and this motivated our group to explore something beyond the curriculum.

- Our Group shares a mutual interest in the Machine Learning and App Development domain, so we decided to combine these two.

Literature Survey

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
1	Textual Dissection Of Live Twitter Reviews Using Naive Bayes	1)Sourav.Kunal 2)Arijit Saha 3)Aman.Varma 4)Varma Vivek Tiwari	2018	This paper proposes the use of Tweepy and TextBlob as a python library to access and classify Tweets using Naïve Bayes. They developed an algorithm that takes the query as the person's name for whom the user wants to calculate the percentage of positive & negative tweets.	1)The live data capture is an excellent feature which takes care that all the data is up to date. 2) Using the pos_tagger makes it easier to use the lemmatizer. 3)TextBlob has a MIT License, is built on the top of NLTK and is very easily accessible. TextBlob is used for fast prototyping. On Comparing the Code Quality as calculated and provided by Lumnify, TextBlob is Level 3 and that of NLTK is Level 2.	1)If there is a foreign category in test dataset, it will be assigned 0 probability. This can be rectified using smoothing techniques. 2)Multinomial Naive Bayes can handle and analyse this data with higher accuracy and efficiency. 3)They also used uni-gram method.

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
2	Sentimental Analysis of Twitter Data with respect to General Elections in India	1)Ankita Sharma 2)Udayan Ghose	2020	In this paper, an efficient approach is proposed to analyze the sentiments of the twitter data.They used twitter for extraction of tweets 1.)Lexicon Based Approach for Sentiment Analysis it classifies words into 3 categories - positive,negative and neutral 2.)NRC Dictionary Based Approach it classify words Into 8 emotions along with negative and positive. They used Rapidminer and its Extension called AYL	1.)Lexicon based approach requires scoring function to score each sentence and it has good accuracy on single phase. 2.)In NRC approach Since data is sentence sized, the positive and negative sentiment scores aren't averaged to zero.	1.)Rapidminer accuracy is less compared to **VADER** (Valence Aware Dictionary for **Sentiment** Reasoning)**[8]** 2.)They used uni-grams which is less effective to catch sarcasm.

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
3	Sentiment analysis of social media Twitter with case of Anti-LGBT campaign in Indonesia using Naive Bayes, Decision Tree & Random Firest Algorithm.	1)Veny Amilia Fitri 2)Rachmadita Andreswari 3)Muhammad Azani Hasibuan	2019	This paper presents the usage of different classifier algorithms to analyse the sentiment associated with Anti-LGBT. It used twitter as a source of data to feed to the algorithms. Based on parameters like recall, precision, F1-measure, it was concluded that Naive Bayes had the highest accuracy (86.43%) compared to rest (~82%). After analysis of data, the resultant major tendency for comments was found to be neutral.	1)A detailed comparative study between Random Forest, Naive Bayes and Decision Tree was made. 2)Naive Bayes can easily provide high accuracy in case of sentence sized data like tweets.	1)The number of models compared is low. 2)Naive Bayes assumes all features to be independent. This will reduce accuracy when collinearity is to be considered.

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
4	Text Classification using Different Feature Extraction Approaches.	1)Robert Dzisevic 2)Dmitrij Sesok	2019	In this paper, three different text feature extraction dimensionality reduction approaches based on TF-IDF were applied for classification using keras. The TF-IDF LSA approach outperformed the plain TF-IDF by 1% for small datasets while the plain TF-IDF gave higher accuracy (91%) for larger datasets. The TF-IDF LDA approach failed to reduce the noise in data in both cases which ended up reducing the accuracy.	1)Overfitting of TF-IDF model is prevented with LSA and LDA enhancements which can become very apparent in very large data-sets. 2)The calculations involved in TF-IDF are very easy.	1) If a word occurs in every element of data-set, TF-IDF will assign it a 0 weightage thus, those words become irrelevant to other corpus terms. 2)The extraction time with TF-IDF increases exponentially with increase in document size.

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
5	Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec	1)Amit Kumar Sharma, 2)Sandeep Chaurasia 3)Devesh Kumar Srivastava	2019	This paper provides sentimental summarization of short sentences. Movie review corpus(imdb) was used. There search is giving a better accurate result for feature extraction through Word2Vec and CNN methods for small sentences of movie review corpus. The proposed model is providing 99.07% accuracy for training samples and 82.19% for testing samples.	1)The CNN model provides better and more accurate results as compared to NB and ELM models. 2) The proposed model is better for learning local features from phrases or words.	1)Occurence of multilingual content in the dataset leads to low sentimental results. 2)The model lacks in accuracy for sequential data (long sentences) 3) Social media databases are big and susceptible to noisy, incomplete, and inconsistent text due to their origin from different people and sources.

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
6	Constructing a heterogeneous training dataset for Emotion Classification	1)Anchal Gupta 2)Satish Mahadevan Srinivasan	2019	This paper presents a dataset for twitter sentiment analysis by using 6 models and compares their F1 Accuracy . In this paper they have used lexicon based NRC classifiers for different emotions such as joyful , sad , angry and surprise.	1) Deep Neural network showed a mediocre performance compared to other models , but these models did not overfit.	1) Models like RF performed good on testing data but performed poorly on training data set due to overfitting.

Sr	Paper	Authors	Year	Summary	Advantages	Disadvantages
7	String-based Multinomial Naïve Bayes for Emotion Detection among Facebook Diabetes Community.	1)Vimala Balakrishnan 2)Wandeep Kaur	2019	This paper determined the emotions among the online Diabetes community on Facebook . The emotions were classified according to Plutchik's wheel of emotions , comprising of 8 emotions fear, joy ,anger , sadness, surprise , trust , anticipation and disgust. It used 4 classifiers and their accuracy was compared .	1) String - based Multinomial Naive Bayes classifier outperformed every other classifier used for particular data set.	1) The biggest drawback of MNB is that if it comes across any category that didn't exist in training data set , it will not be able to predict the output, as the probability of that category was assigned to zero.

Problem statement

- We are performing the sentiment analysis of Amazon product reviews after text classification using various Machine Learning algorithms .

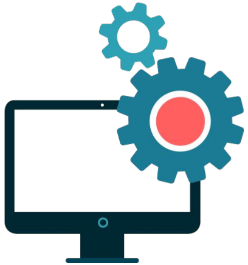
- We will test and compare a number of supervised models to determine which is the best suited for our purpose.

- As of now, we are going to use algorithms such as SVM, Decision Tree and different types of Naive Bayes in supervised learning domain.

Objectives

- To understand various aspects of different algorithms and thus, compare them qualitatively as well as quantitatively.
- To perform sentiment analysis of Amazon product reviews and determine the overall consumer polarity.
- To deploy an application to demonstrate sentiment analysis.

Requirements



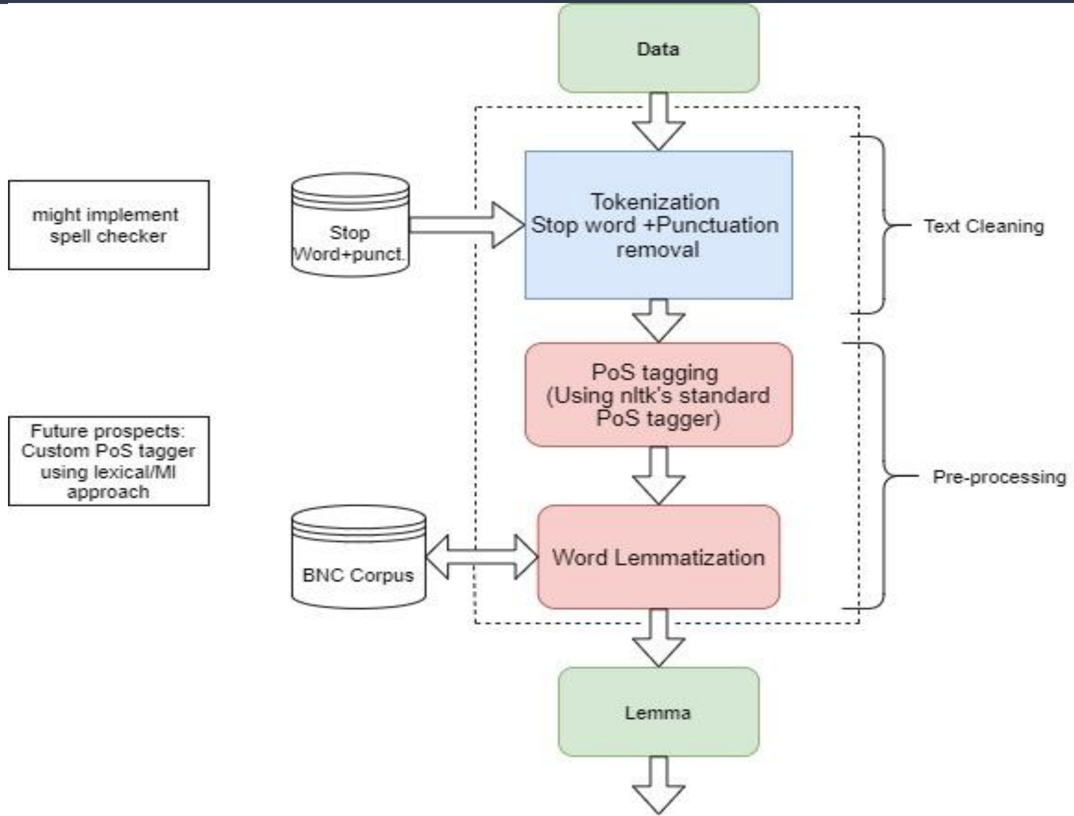
Hardware -

- HP Pavilion-15 laptop with an Intel core i5 9th gen processor upto 4.1 GHz and GTX 1050 ti mobile graphics card.
- HP Pavilion-15 laptop with a Ryzen 5 3rd gen processor upto 4.2 GHz and GTX 1650 mobile graphics card.

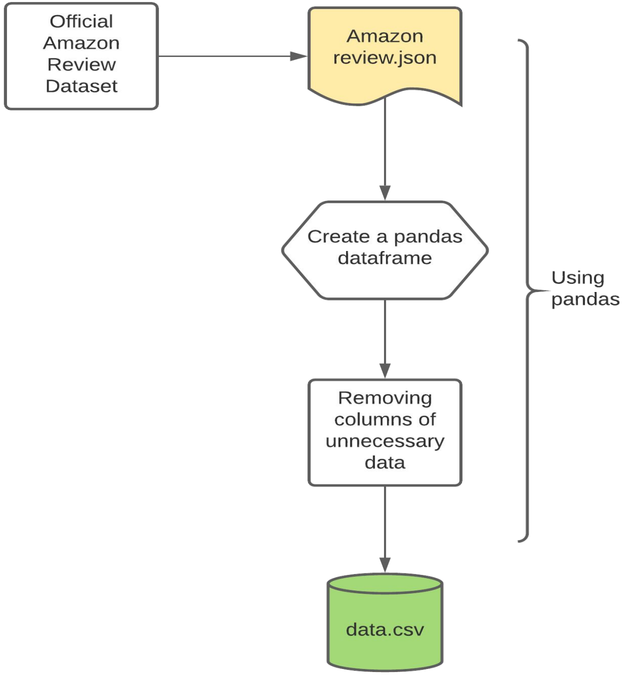
Software -

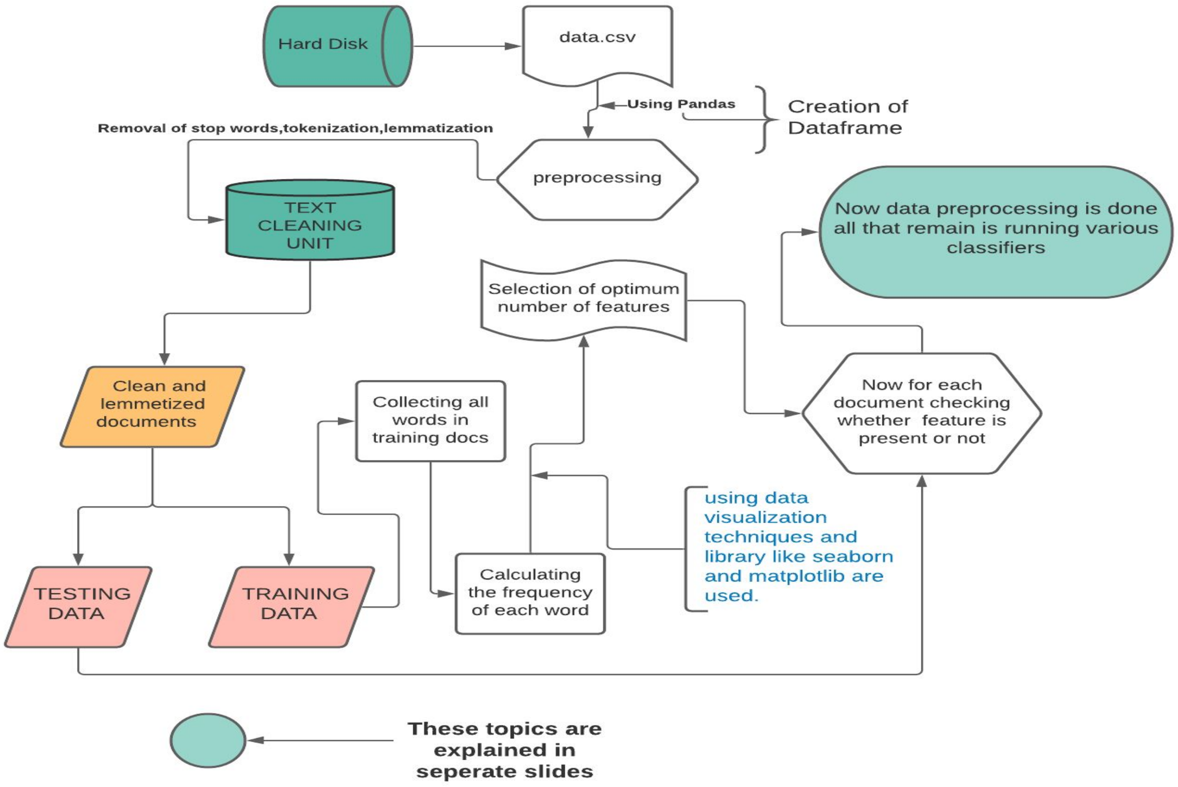
- Jupyter Notebook, Python, Anaconda & its libraries.
- GitHub.
- Android Studio, Flutter.
- More requirements will be added as the project progresses.

Text-Cleaning

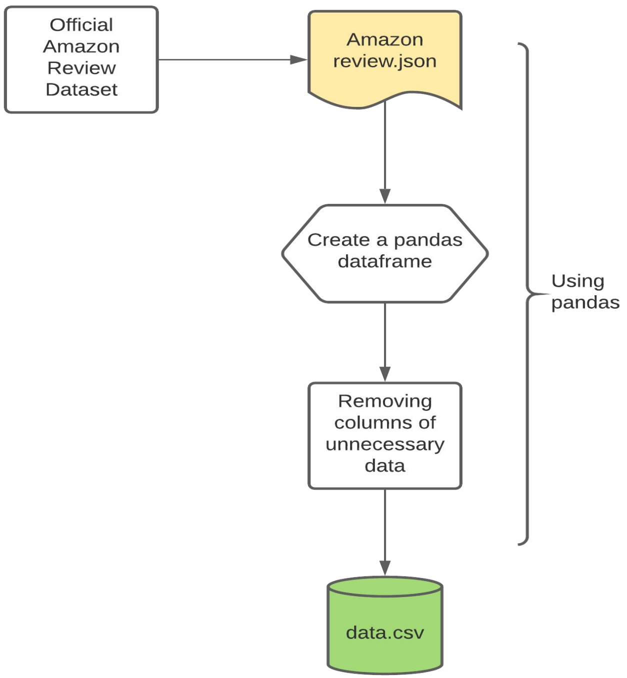


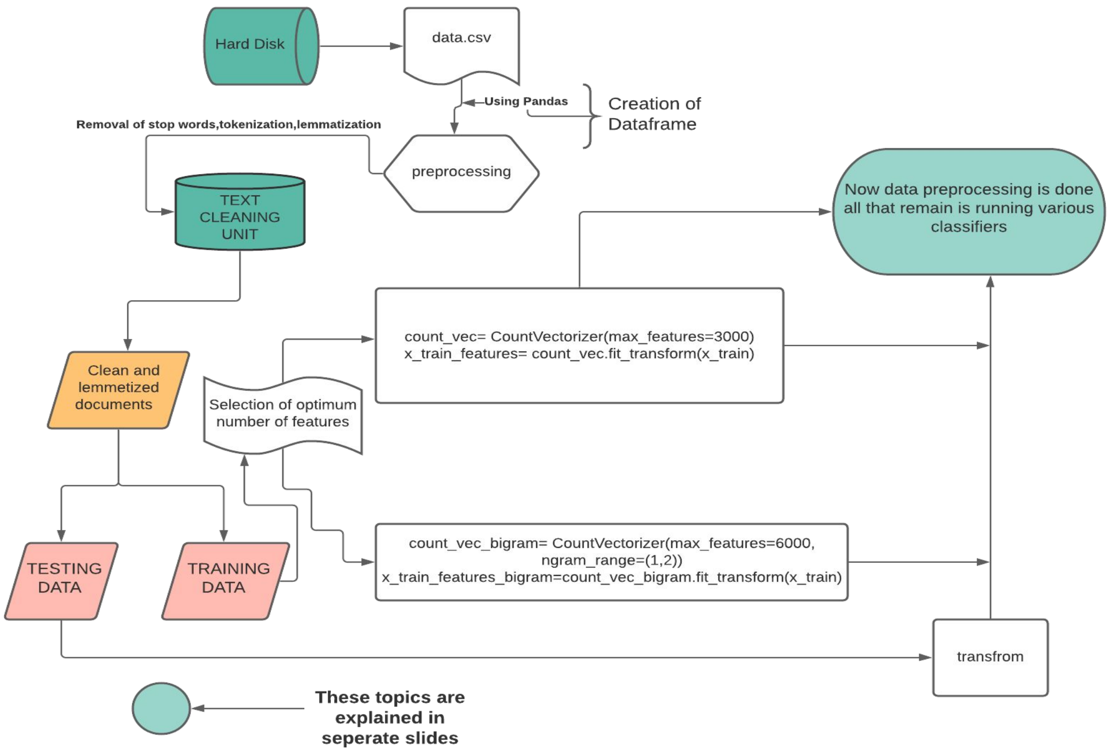
Analysis Unit
(True/False Based)



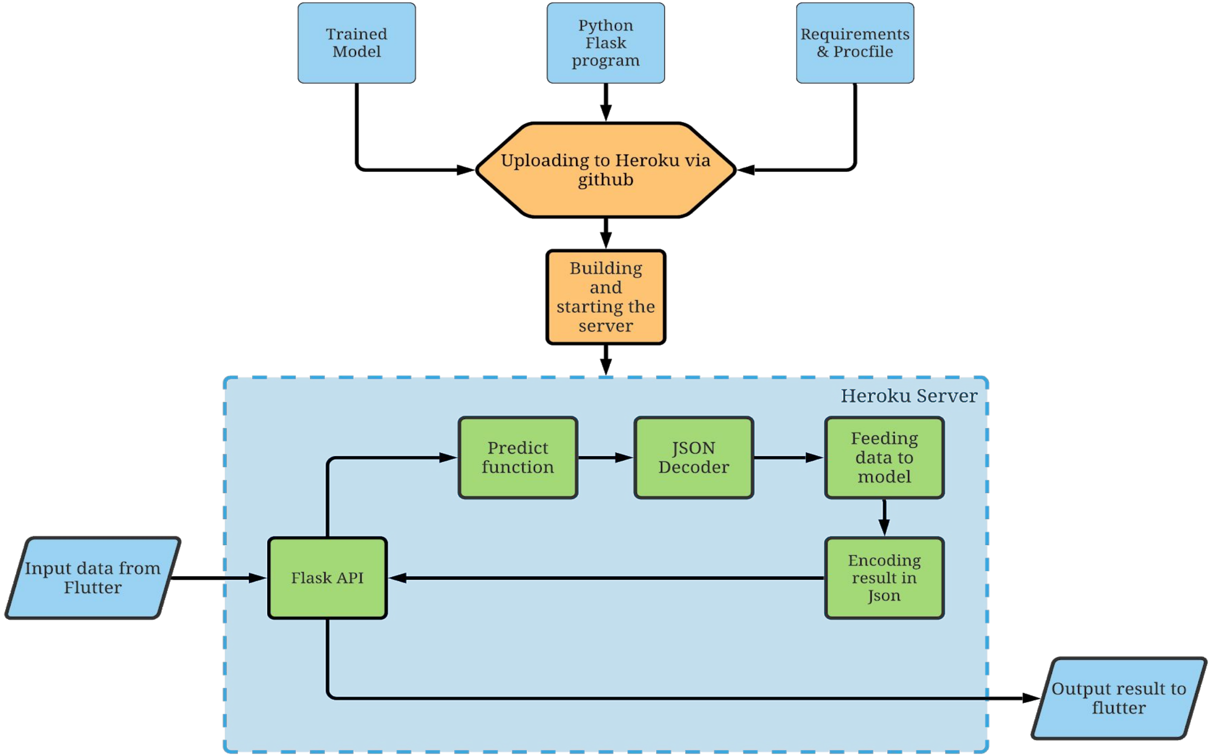


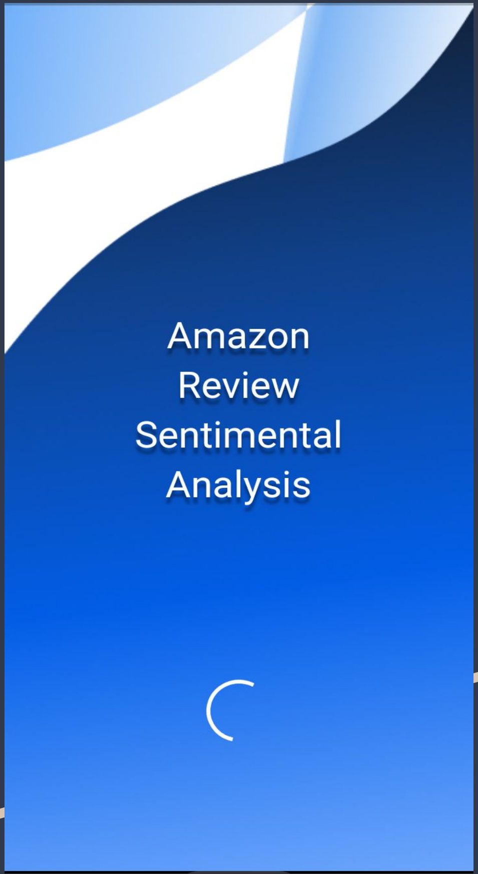
Analysis Unit
(Count Vectorized Based)



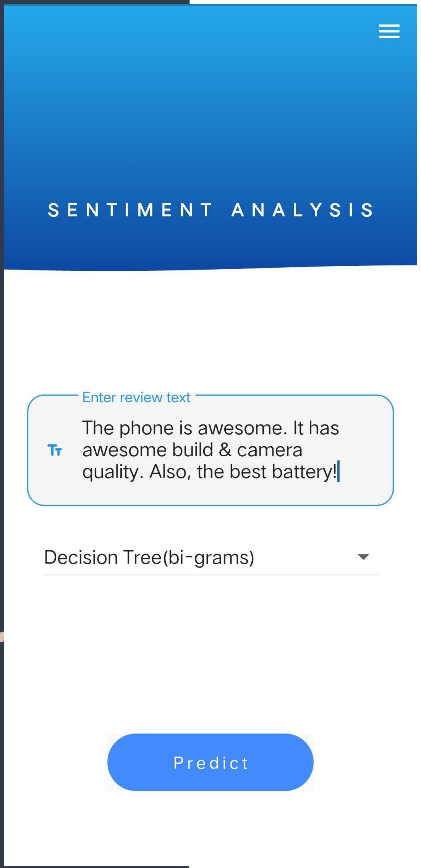


Flutter Integration via API

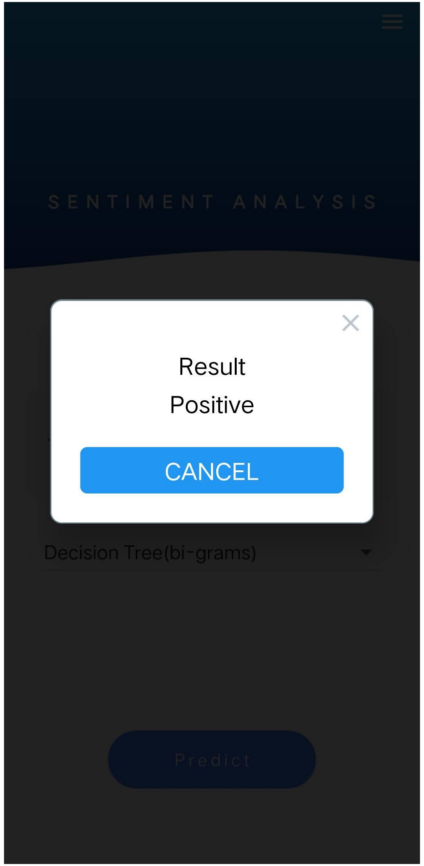




Loading Page



Home Page



Result

Data Introduction

	A	B	C	D	E	F	G	H	I
1		rating	verified	reviewTim	reviewerID	productID	reviewTex	summary	unixReview
2	0	5	TRUE	08 4, 2014	A24E3SXT(	7.51E+09	Looks eve	Can't stop	1.41E+09
3	1	5	TRUE	02 12, 201	A269FLZCE	7.51E+09	When you	1	1.39E+09
4	2	3	TRUE	02 8, 2014	AB6CHQW	7.51E+09	so the case	Its okay	1.39E+09
5	3	2	TRUE	02 4, 2014	A1M117A5	7.51E+09	DON'T CAI	CASE	1.39E+09
6	4	4	TRUE	02 3, 2014	A272DUT8	7.51E+09	I liked it be	Cute!	1.39E+09
7	5	2	TRUE	01 27, 201	A1DW2L6X	7.51E+09	The produ	Not so hap	1.39E+09
8	6	3	TRUE	01 23, 201	AQC61R4U	7.51E+09	I FINALLY g	It's cute!	1.39E+09
9	7	5	TRUE	01 17, 201	A31OVFL9	7.51E+09	It is a very	Cute case	1.39E+09
10	8	1	TRUE	12 27, 201	A1K0VLK6(	7.51E+09	DO NOT B	WORST ITI	1.39E+09
11	9	4	TRUE	12 16, 201	A1K3BWU	7.51E+09	I really lov	Pretty Cute	1.39E+09
12	10	4	TRUE	10 27, 201	A1LBVKZ3	7.51E+09	its super c	case	1.38E+09
13	11	5	TRUE	10 23, 201	A2ZB7KGU	7.51E+09	Another gr	Bling bling	1.38E+09
14	12	1	TRUE	10 8, 2013	A1ROZSS0I	7.51E+09	Very cheap	cheap plas	1.38E+09
15	13	4	FALSE	09 19, 201	A1EHAOKC	7.51E+09	I purchase	Great	1.38E+09
16	14	5	TRUE	09 10, 201	A18U23JW	7.51E+09	Beautiful c	I can't stop	1.38E+09
17	15	5	TRUE	08 28, 201	A1JQUCTF	7.51E+09	It is such a	I love it	1.38E+09
18	16	3	TRUE	07 14, 201	A3NS9QFI	7.51E+09	Can not ar	Great for p	1.37E+09
19	17	2	TRUE	06 27, 201	A1BZ055L\	7.51E+09	I used this	....	1.37E+09
20	18	5	TRUE	06 4, 2013	A29KSIE8B	7.51E+09	Super dura	Good case	1.37E+09

- RevieweD - ID of the reviewer, e.g. A2SUAM1J3GNN3B
- asin - ID of the product, e.g. 0000013714
- review Text - text of the review
- summary - summary of the review
- unix Review Time - time of the review (unix time)
- review Time - time of the review (raw)
- image - images that users post after they have received the product

The dataset is from Amazon Review Data (2018):https://nijianmo.github.io/amazon/index.html

Methodology

Methodology for Text Cleansing

A) Tokenization and Stop word removal: These tasks are doing using **standard** python libraries. However ,a separate dataset of stop words was created for stop word removal. The final list of clean words will be fed into the POS tagger.

B) POS tagger: During the initial time of our research, we sought to implement **python nltk's pre-trained PoS taggers**. However for optimisation of PoS tagging with respect to our Customer Review Dataset, we plan to implement a separate PoS tagger. There are two approaches for this:

1) Treating the dataset as a classification problem where given a word and features like previous word,context and so on to classify the given word into a PoS category.

2) While other approach try to model it as a generative model using similar features.

C) Lemmatizer: Lemmatizing is the process of obtaining base word(lemma) from a word based on its part_of_speech. Our team has utilized an open source corpus , the British National Corpus(BNC) for creating the Lemmatizer. We further created an annotated dataset (tagged corpus) to provide the lemma for each word. Basically it is a huge list of words and their related lemma for each PoS.

Methodology for Model Selection and Future Scope

MODELS / GOALS	SUPERVISED MODELS:	UNSUPERVISED MODELS:
THIS ROW IS SOMETHING FOR SURE WE ARE GOING TO IMPLEMENT.	We are implementing **inbuilt models** as of now,mainly various types of Models in sklearn and nltk.	We are using LSTM model which is inside keras which is a high level API and is integrated as a part of Tensorflow 2.0
THIS ROW IS US BEING NAIVE AND HOPING TO IMPLEMENT.	We are optimistic to build a multinomial naive bayes(MNB) and KNN model from the very scratch and compare its accuracy with inbuilt models.	We are hoping to increase its accuracy by a significant amount by using some manual methods and better optimized text cleaning based on different ml models in accordance with the used dataset.

Methodology For Integrating App with ML Model

- The trained model, API (python flask) & other requirements are uploaded to Heroku server. The user supplies the input data from flutter interface which sends it as a query to API on server using http dependency after Analyze button is pressed.

- This also invokes the '/predict' app route where data is fed to the model imported into flask API via 'pickle library'. The model predicts a result which is converted into json before sending to the server. The 'http' dependency has a Response class that stores the responses received from http 'GET' function.

- The json format received from the API is decoded in flutter using 'convert' dependency. The result of this decoding is shown as predicted output to the user.

Expected Output

Based on reference of research papers -

- In Supervised Machine learning algorithms, Multinomial Naive Bayes will outperform its counterparts.

- DT, SVM and Multinomial Naive Bayes will provide similar accuracy.

- Deep learning models like LSTM however, should outperform these Machine Learning algorithms by a considerable margin.

Proposed Evaluation Measures

- Depth of comparison between different algorithms while selection phase.
- Accuracy of trained model.
- Response time of the program.
- Deployment of model via mobile application.
- System Requirements for execution of application.

Confusion Matrices:

```
In [68]:
y_pred_bi=MultinomialNB_bigram_clf.predict(x_test_features_bigram)
cm=confusion_matrix(y_test,y_pred_bi)
cm

Out[68]: array([[1714,  638],
                [ 824, 6824]], dtype=int64)
```

MNB(bi-grams)

SVC(bi-grams)

```
In [53]:
y_pred_bi=svc_bigram.predict(x_test_features_bigram)
cm_bi=confusion_matrix(y_test,y_pred_bi)
cm_bi

Out[53]: array([[1296, 1150],
                [ 257, 7297]], dtype=int64)
```

```
In [109]:
y_pred_uni=Dt_clf_uni.predict(x_test_features)
cm=confusion_matrix(y_test,y_pred_uni)
cm

Out[109]: array([[1322, 1124],
                 [1052, 6502]], dtype=int64)
```

DT(uni-grams)

Confusion Matrices:

SVC(uni-grams)

```
In [48]:  y_pred_uni=svc.predict(x_test_features)
          cm=confusion_matrix(y_test,y_pred_uni)
          cm

Out[48]: array([[1258, 1188],
                [ 263, 7291]], dtype=int64)
```

MNB(uni-grams)

```
In [63]:  y_pred_uni=MultinomialNB_uni_clf.predict(x_test_features)
          cm=confusion_matrix(y_test,y_pred_uni)
          cm

Out[63]: array([[1551,  801],
                [ 671, 6977]], dtype=int64)
```

DT(bi-grams)

```
In [115]:  y_pred_bi=Dt_clf_bi.predict(x_test_features_bigram)
           cm_bi=confusion_matrix(y_test,y_pred_bi)
           cm_bi

Out[115]: array([[1303, 1143],
                 [1006, 6548]], dtype=int64)
```

Experiment & Result:

Accuracy: the percentage of texts that were categorized with the correct tag.

Precision: the percentage of examples the classifier got right out of the total number of examples that it predicted for a given tag.

Recall: the percentage of examples the classifier predicted for a given tag out of the total number of examples it should have predicted for that given tag.

F1 Score: the harmonic mean of precision and recall.

SVM(uni-grams)

Model		Precision	Recall	F1-score	Support
SVM (uni-grams)	**neg**	0.83	0.51	0.63	2446
	pos	0.86	0.97	0.91	7554
	accuracy		*0.85*		10000
	macro avg	0.84	0.74	0.77	10000
	weighted avg	0.85	0.85	0.84	10000

SVM(bi-grams)

Model		Precision	Recall	F1-score	Support
SVM (bi-grams)	**neg**	0.83	0.53	0.65	2446
	pos	0.86	0.97	0.91	7554
	accuracy		*0.86*		10000
	macro avg	0.85	0.75	0.78	10000
	weighted avg	0.86	0.86	0.85	10000

DT(uni-grams)

Model		Precision	Recall	F1-score	Support
DT (uni-grams)	**neg**	0.56	0.54	0.55	2446
	pos	0.85	0.86	0.86	7554
	accuracy		*0.78*		10000
	macro avg	0.70	0.70	0.70	10000
	weighted avg	0.78	0.78	0.78	10000

DT(bi-grams)

Model		Precision	Recall	F1-score	Support
DT (bi-grams)	**neg**	0.56	0.54	0.55	2446
	pos	0.85	0.86	0.86	7554
	accuracy		*0.79*		10000
	macro avg	0.71	0.70	0.70	10000
	weighted avg	0.78	0.79	0.78	10000

MNB(uni-grams)

Model		Precision	Recall	F1-score	Support
MNB (uni-grams)	**neg**	0.70	0.66	0.68	2352
	pos	0.90	0.91	0.90	7648
	accuracy		*0.85*		10000
	macro avg	0.80	0.79	0.79	10000
	weighted avg	0.85	0.85	0.85	10000

MNB(bi-grams)

Model		Precision	Recall	F1-score	Support
MNB (bi-grams)	**neg**	0.68	0.73	0.70	2352
	pos	0.90	0.91	0.90	7648
	accuracy		*0.85*		10000
	macro avg	0.79	0.81	0.80	10000
	weighted avg	0.86	0.85	0.85	10000

Conclusion

After research, we can conclude that -

- It was observed that DT has the lowest scores **(0.78)** while the rest of the models had a similar accuracy **(0.86 - 0.87)**.
- Bigram models outperformed Unigram ones by **1-2%** while most CV Models being able to catch a little bit of sarcasm.
- However, in live testing it was seen that DT performs on par with SVM while the NLTK based Bernoulli's NB displayed the worst results.
- Selection of appropriate Machine Learning models is essential.
- Tedious data cleaning processes can be automated and optimised using ML models, feature reduction, etc.
- Python can be integrated along with flutter using Flask API which makes it easy to deploy ML models for public use via mobile applications.

References

[5]Sourav Kunal, Arijit Saha, Aman Varma, Vivek Tiwari , International Conference on Computational Intelligence and Data Science (ICCIDS 2018), **“Textual Dissection Of Live Twitter Reviews Using Naive Bayes”**,Procedia Computer Science Volume 132, Pages 307-313,2018

[6] Veny Amilia Fitri, Rachmadita Andreshwari and Muhammad Azani Hasibuan, “**Sentiment analysis of social media Twitter with case of Anti-LGBT campaign in Indonesia using Naive Bayes, Decision Tree & Random Firest Algorithm**,” The Fifth Information Systems International Conference 2019, Procedia Computer Science, vol 161, pp. 765–772, 2019.

[7]Robert Dzisevič and Dmitrij Šešok, “ **Text Classification using Different Feature Extraction Approaches,**” 2019 Open Conference of Electrical, Electronic and Information Sciences, 25-25 April 2019.

[8] Anton Borg and Martin Boldt., “**Using VADER Sentiment and SVM for Predicting Customer Response Sentiment**,” Expert Systems with Applications, 2020.

References

[1] Vimala Balakrishnan & Wandeep Kaur , 23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems , "**String-based Multinomial Naïve Bayes for Emotion Detection among Facebook Diabetes Community.**" Procedia Computer science , 159 , pp.30-37 (2019).

[2] Anchal Gupta and Satish Mahadevan Srinivasan , The Fifth Information Systems International Conference 2019 "**Constructing a heterogeneous training dataset for Emotion Classification.**" Procedia Computer Science , 161 , pp. 765-772 (2019)

[3] Amit Sharma ,Sandeep Chaurasia and Devesh Kumar Shrivastava, International Conference on Computational Intelligence and Data Science (ICCIDS 2019), "**Sentimental Short Sentences Classification by Using CNN DeepLearning Model with Fine Tuned Word2Vec**.",Procedia Computer Science 167, pp.1139-1147,2020.

[4] Ankita Sharma, Udayan Ghose ,International Conference on Smart Sustainable Intelligent Computing and Applications under ICITETM2020 , **"Sentimental Analysis of Twitter Data with respect to General Elections in India"** Procedia Computer Science Volume 173 , Pages 325-334 , 2020

THANK YOU !