VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SMS SPAM FILTERING USING NATURAL LANGUAGE PROCESSING

Submitted by:

K. RITESH CHOWDARY	- 18BCE0609
K. SAI PREETHAM	- 18BCE0589
VINOD RONGALA	- 18BCE0042

Prepared for

NATURAL LANGUAGE PROCESSING(CSE4022)

PROJECT COMPONENT

Submitted to

Prof: Sharmila Banu K

School of Computer Science and Engineering (SCOPE)

Table of Contents

Abstract: -

The Short Message Service (SMS) have a significant financial effect for end clients and specialist organizations. Spam is a genuine all-inclusive issue that messes up practically all clients. A few examinations have been introduced, including executions of spam channels that keep spam from arriving at their objective. Navies Bayesian algorithm is one of the best methodologies utilized in separating strategies. The computational intensity of cell phones is expanding, making progressively conceivable to perform spam sifting at these gadgets as a versatile phone application, prompting better personalization and adequacy. The test of separating SMS spam is that the short messages frequently comprise of not many words made out of contractions and phrases. The proposed procedure uses a bunch of certain highlights that can be utilized as contributions to spam location model. This is to organize messages using arranged dataset that contains Phone Numbers, Spam Words, and Detectors. Our proposed procedure uses a twofold assortment of mass SMS messages Spam and Ham in the training process. We express a bunch of stages that help us to construct dataset, for example, tokenizer, stop word channel, and training process. The outcomes applied to the testing messages show that the proposed framework can group the SMS spam and ham with exact contrasted of Naïve Bayesian calculation.

Objective : -

The objective of the project was to create a model which will accurately differentiate spam messages from ham ones. In doing so, we tried to showcase the ability of the naive-bayes algorithm to accurately predict the message as spam or ham which will be used to detect unsolicited and unwanted SMS and prevent those messages from getting to a user's inbox. Like different kinds of filtering programs, a spam filter appearance for sure criteria on that it bases judgments.

SPAM :-

Today’s SPAM, also known as junk is a far cry from 1937 and has nothing to do with solving problems, and everything to do with creating them. Spammers may argue that it’s not really a problem and that you can merely delete what you don’t want, but that is naïve in the extreme. Whether received as a private user, or as a business professional, SPAM is a cyber menace for a number of reasons:

- On the most basic level, receiving a large volume of SPAM SMS wastes valuable bandwidth and time. Private user is forced to individually delete their unwanted message and administrators have to fight similar problems but on a far larger scale in an attempt to keep our
- This wasted time and effort inevitably leads to a loss in productivity as valuable resources are in-

efficiently allocated to non-profitable enterprises.

- SPAM is a prime means of transferring electronic viruses and malware infections, whether deliberately, or by accident as the direct result of generating mass SMS to and from a large number of recipients.
- SPAM is also not merely restricted to mass marketing schemes. Professional criminal organizations use it to instigate complex frauds and scams such as phishing attacks and “419” Nigerian fraud type scams that have made the news in recent years. It is incredibly cheap for the spammers to send hundreds of thousands of SMS an hour, but the cost of receiving them can be many times greater, both interms of monetary outlay and also the cost of rectifying all of the other associated problems.
- Perhaps the greatest problem is the irritation factor. SPAM is extremely annoying and there is nothing worse than accessing your SMS provider to find countless SMS advertising products you don’t need, pornography you don’t want and scams you want to avoid.

Naive Bayes spam filtering :-

Naive Bayes classifiers are a popular statistical technique of SMS filtering. They typically use bag of words features to identify spam SMS, an approach commonly used in text classification. Naive Bayes classifiers work by correlating the use of tokens (typically words, or sometimes other things), with spam and non-spam SMS and then using Bayes' theorem to calculate a probability that an SMS is or is not spam. Naive Bayesian spam filtering may be a baseline technique for coping with spam which will tailor itself to the SMS wants of individual users and provides low false positive spam detection rates that area unit typically acceptable to users. It's one in all the oldest ways that of doing spam filtering, with roots within the Nineties.

Process: -

Particular words have particular probabilities of occurring in spam SMS and in legitimate SMS. For instance, most users will frequently encounter the word "Viagra" in spam SMS, but will seldom see it in other SMS. The filter does not recognize these possibilities ahead, and should 1st be trained therefore it will build them up. To train the filter, the user must manually indicate whether a new SMS is spam or not. For all words in each training SMS, the filter will adjust the probabilities that each word will appear in spam or legitimate SMS in its database. As an example, Bayesian spam filters can generally have learned high spam chance for the words "Viagra" and "refinance", however a low spam chance for words seen solely in legitimate SMS, such as the names of friends and family members.

After training, the word probabilities (also known as likelihood functions) are used to compute the probability that an SMS with a particular set of words in it belongs to either category. Each word in the SMS contributes to the SMS's spam probability, or only the most interesting words. This contribution is termed posterior probability and is calculated using Bayes' theorem. Then, the SMS's spam probability is computed over all words in the SMS, and if the total exceeds a certain threshold (say 95%), the filter will mark the SMS as a spam.

As in any other spam filtering technique, SMS marked as spam can then be automatically moved to a "Junk" SMS folder, or even deleted outright. Some software package implements quarantine mechanisms that outline a timeframe throughout that the user is allowed to review the software's call. The initial coaching will sometimes be refined once wrong judgements from the software package area unit known (false positives or false negatives).that permits the software package to dynamically adapt to the ever-evolving nature of spam. Some spam filters mix the results of each theorem spam filtering and different heuristics (pre- outlined rules regarding the contents, viewing the message's envelope, etc.), leading to even higher filtering theorem.

Mathematical foundation :-

Bayesian SMS filters utilize Bayes' theorem. Bayes' theorem is used several times in the context of spam:

- Firstly, to figure the chance that the message is spam, knowing that a given word seems during this message
- Secondly, to figure the chance that the message is spam, taking into thought all of its words (or a relevant set of them)
- sometimes a third time, to deal with rare words

Computing the probability that a message containing a given word is spam

The formula used by the software to determine that, is derived from Bayes' theorem: -

$$\Pr(S|W) = \frac{\Pr(W|S) \cdot \Pr(S)}{\Pr(W|S) \cdot \Pr(S) + \Pr(W|H) \cdot \Pr(H)}$$

Combining individual probabilities

$$p = \frac{p_1 p_2 \cdots p_N}{p_1 p_2 \cdots p_N + (1 - p_1)(1 - p_2) \cdots (1 - p_N)}$$

Where ‘p’ is the probability that the suspect message is spam;

Dealing with rare words

$$\Pr'(S|W) = \frac{s \cdot \Pr(S) + n \cdot \Pr(S|W)}{s + n}$$

Advantages :-

One of the main advantages of Bayesian spam filtering is that it is trained on a per-user basis. The word probabilities are unique to each user and can evolve over time with corrective training whenever the filter incorrectly classifies an SMS. As a result, Bayesian spam filtering accuracy after training is often superior to predefined rules.

It will perform significantly well in avoiding false positives, wherever legitimate SMS is incorrectly classified as spam. as an example, if the SMS contains the word "Nigeria", that is usually utilized in Advance fee fraud spam, a predefined rules filter may reject it outright. A Bayesian filter would mark the word "Nigeria" as a probable spam word, but would consider other important words that usually indicate legitimate e-mail. For example, the name of a spouse may strongly indicate the e-mail is not spam, which could overcome the use of the word "Nigeria." Super simple, you’re just doing a bunch of counts. If the NB conditional independence assumption really holds, a Naive Bayesian classifier can converge faster than discriminative models like supply regression, therefore you would like less coaching information. And though the NB assumption doesn’t hold, a NB classifier still typically will a good job in observe. a decent bet if need one thing quick and straightforward that performs practically. Its main disadvantage is that it can’t learn interactions between options (e.g., it can’t learn that though you're keen on movies with Brad Pitt and Tom Cruise, you hate movies wherever they’re together).

Disadvantages :-

Depending on the implementation, Bayesian spam filtering could also be at risk of Bayesian poisoning, a way utilized by spammers in an effort to degrade the effectiveness of spam filters that rely on Bayesian filtering. A spammer practicing Bayesian poisoning will send out SMS with large amounts of legitimate text (gathered from legitimate news or literary sources). Spammer tactics include insertion of random innocuous words that are not normally associated with spam, thereby decreasing the SMS's spam score, making it more likely to slip past a Bayesian spam filter. However, with (for example) Paul Graham's theme solely the foremost vital possibilities area unit used, so artifact the text out with non-spam-related words doesn't have an effect on the detection chance considerably.

Words that usually seem in massive quantities in spam may be remodeled by spammers. As an example, «Viagra» would get replaced with «Viaagra» or «V!agra» in the spam message. The recipient of the message can still read the changed words, but each of these words is met more rarely by the Bayesian filter, which hinders its learning process. As a general rule, this spamming technique does not work very well, because the derived words end up recognized by the filter just like the normal ones.

Stages : -

- Environment Setting
- Convert a corpus to a vector format: bag-of-words approach
- Massage the raw message (sequence of characters) into vectors (sequences of numbers)
- split the message into words and return a list
- remove punctuation
- remove quite common words('the','a',etc.)
- Vectorization
- vectoring our messages
- convert every message (represented as an inventory of tokens) into a vector
- count DF in the vector
- weight the counts(IDF)
- Normalize the vectors to unit length (L2 norm)

Literature Review :-

Globally, short messaging service (SMS) is one amongst the foremost widespread and additionally most cheap telecommunication service packages. However, mobile users became progressively involved relating to the protection of their shopper confidentiality. This can be primarily to the actual fact that mobile selling remains intrusive to the non-public freedom of the subscribers. SMS spamming has become a significant nuisance to the mobile subscribers given its pervasive nature. It incurs substantial value in terms of lost productivity, network information measure usage, management, and raid of non-public privacy. Thus, in short spamming threatens the profits of the service providers. Mobile SMS spams frustrate the movable users, and similar to email spams, they cause new social group frictions to mobile telephone set devices. Email spam is distributed or received via the planet Wide internet, whereas the SMS mobile spam is often broad casted via a mobile network.

Spam is represented as unwanted or uninvited electronic messages sent in bulk to a gaggle of recipients. The messages area unit characterized as electronic, uninvited, commercial, mass constitutes a growing threat primarily due to the subsequent factors:

1. the supply of inexpensive bulk SMS plans
2. dependableness (since the message reaches the mobile user)
3. low probability of receiving responses from some unsuspecting receivers
4. the message is personalized. Mobile SMS spam detection and interference isn't a trivial matter

It has taken on tons of problems and solutions inheritable from comparatively older situations of email spam detection and filtering. uninvited SMS text messages area unit a standard prevalence in our lifestyle and consume communication time, information measure and resources. though the present spam filters give some level of performance, the spams inform receivers by maneuvering information samples.

Haiyi Zhang, Di Li [1], studied how a spam email detector is developed using naive Bayes algorithm. They use pre-classified emails (priory knowledge) to train the spam email detector. With the model generated from the training step, the detector is able to decide whether an email is a spam email or an ordinary email. They used Text categorization, Detectors, Bayesian methods, Probability, Classification algorithms, Inference algorithms in the process.

Efnan Sora Gunal, Semih Ergin, Serkan Gunal, Alper Kursat Uysal[2], studied about a novel framework for SMS spam filtering is introduced in this paper to prevent mobile phone users from unsolicited SMS messages. The framework makes use of two distinct feature selection approaches based on information gain and chi-square metrics to find out discriminative features representing SMS messages. The discriminative feature subsets are then employed in two different Bayesian-based classifiers, so that SMS messages are categorized as either spam or legitimate. Moreover, the paper introduces a real-time mobile application for Android™ based mobile phones utilizing the proposed spam filtering scheme, as well. Hence, SMS spam messages are silently filtered out without disturbing phone users. Effectiveness of the filtering framework is evaluated on a large SMS message collection including legitimate and spam messages. They used unsolicited SMS messages, feature selection, information gain, discriminative feature subsets, Bayesian-based classifier, SMS message categorization, real-time mobile application, Bayes methods, feature extraction, information filtering, mobile computing, pattern classification, real-time systems in their work.

Steven Kay, Paul M. Baggenstoss, Haibo He, Bo Tang [3], presented a Bayesian classification approach for automatic text categorization using class- specific features. Unlike conventional text categorization approaches, their proposed method selects a specific feature subset for each class. To apply these class-specific features for classification, they follow Baggenstoss's PDF Projection Theorem (PPT) to reconstruct the PDFs in raw data space from the class-specific PDFs in low- dimensional feature subspace, and build a Bayesian classification rule. One noticeable significance of their approach is that most feature selection criteria, such as Information Gain (IG) and Maximum Discrimination (MD), can be easily incorporated into our approach. They evaluate their method's classification performance on several real-world benchmarks, compared with the state-of-the-art feature selection approaches.

Jingnian Chen, Houkuan Huang, Shengfeng Tian, Youli Qu [4] presented two feature evaluation metrics for the Naïve Bayesian classifier applied on multi- class text datasets: Multi-class Odds Ratio (MOR), and Class Discriminating Measure (CDM). Experiments of text classification with Naïve Bayesian classifiers were carried out on two multi-class texts collections. As the results indicate, CDM and MOR gain obviously better selecting effect than other feature selection approaches. They used Text classification, Feature selection, Text preprocessing, Naïve Bayes in their work.

Min-Ling Zhang, José M.Peña, Victor Robles [5] studied how learning problem is addressed by using a method called Mlnb which adapts the traditional naive Bayes classifiers to deal with multi-label instances. Feature selection mechanisms are incorporated into Mlnb to improve its performance. Firstly, feature extraction techniques based on principal component analysis are applied to remove irrelevant and redundant features. After that, feature subset selection techniques based on genetic algorithms are used to choose the most appropriate subset of features for prediction. Experiments on synthetic and real-world data show that Mlnb achieves comparable performance to other well-established multi-label learning algorithms. They used Multi-label learning, Naive Bayes, Feature selection, Principal component analysis, Genetic algorithm in their work.

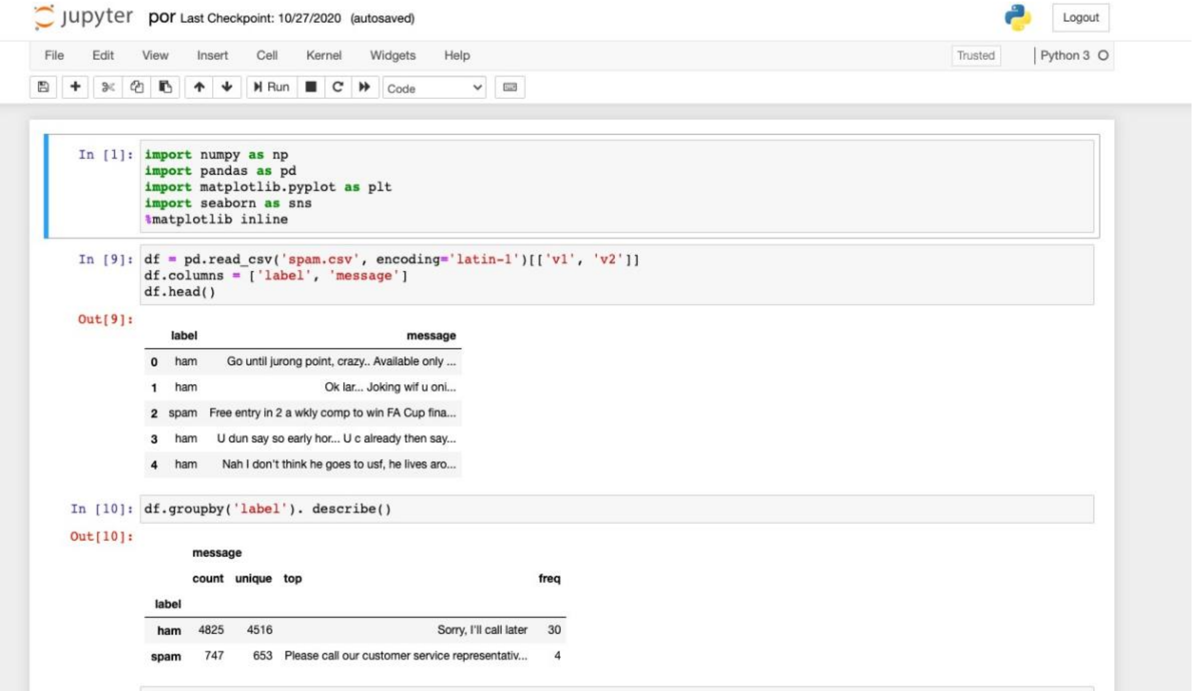
Birru Devender, Korra Srinivas, Ch.Tulasi Ratna Mani[6] developed an effective spam zombie detection system named SPOT by monitoring outgoing messages in a network. SPOT was designed based on a simple and powerful statistical tool named Sequential Probability Ratio Test to detect the compromised machines that are involved in the spamming activities. SPOT has surpassed both the false positive and false negative error rates. It also minimizes the number of required observations to detect a spam zombie. In addition, They also showed that SPOT outperforms two other detection algorithms based on the number and percentage of spam messages sent by an internal machine, respectively. The main usage of the application is sender can identify the sending mails as either spam or not and weather his system is compromised system or an uncompromised one and the user defined thresholds algorithms which are CT and PT can support the dynamic behavior to detect the spam mails associated with different address locations. They compromised machines in a network that are used for sending spam messages, spam zombie detection system, Sequential Probability Ratio Test (SPRT).

H. Shinnou, M. Sasaki[7] proposed a new spam detection technique using the text clustering based on vector space model. Their method computes disjoint clusters automatically using a spherical k-means algorithm for all spam/non- spam mails and obtains centroid vectors of the clusters for extracting the cluster description. For each centroid vector, the label (`spam' or `non-spam') is assigned by calculating the number of spam email in the cluster. When new mail arrives, the cosine similarity between the new mail vector and centroid vector is calculated. Finally, the label of the most relevant cluster is assigned to the new mail. By using their method, they can extract many kinds of topics in spam/non-spam email and detect the spam email efficiently. They describe our spam detection system and show the result of our experiments using the Ling- Spam test collection spam detection, text clustering, vector space model, centroid vectors, Ling-Spam test collection.

Nitin Jindal, Bing Liu[8] studied the issue in the context of product reviews. They will see that review spam is quite different from Web page spam and email spam, and thus requires different detection techniques. Based on the analysis of 5.8 million reviews and 2.14 million reviewers from amazon.com, they show that review spam is widespread. In this paper, they first present a categorization of spam reviews and then propose several techniques to detect them. They used product review spam detection, product reviews opinion mining, Web page spam, email spam, spam review categorization in their work.

Jieping Zhong, Jiachun Du, Qiang Yang, Evan Wei Xiang, Qian Xu[9] studied about Short Message Service text messages that are indispensable, but they face a serious problem from spamming. This service-side solution uses graph data mining to distinguish spammers from non-spammers and detect spam without checking a message's contents. They used Support vector machines, Feature extraction, Classification algorithms, Electronic mail, Telecommunications, Short message services, Unsolicited electronic mail, Short Message Service, spam detection, SMS spam, social media spam, data in their work.

Code :-



jupyter por Last Checkpoint: 10/27/2020 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        %matplotlib inline
```

```
In [9]: df = pd.read_csv('spam.csv', encoding='latin-1')[['v1', 'v2']]
        df.columns = ['label', 'message']
        df.head()
```

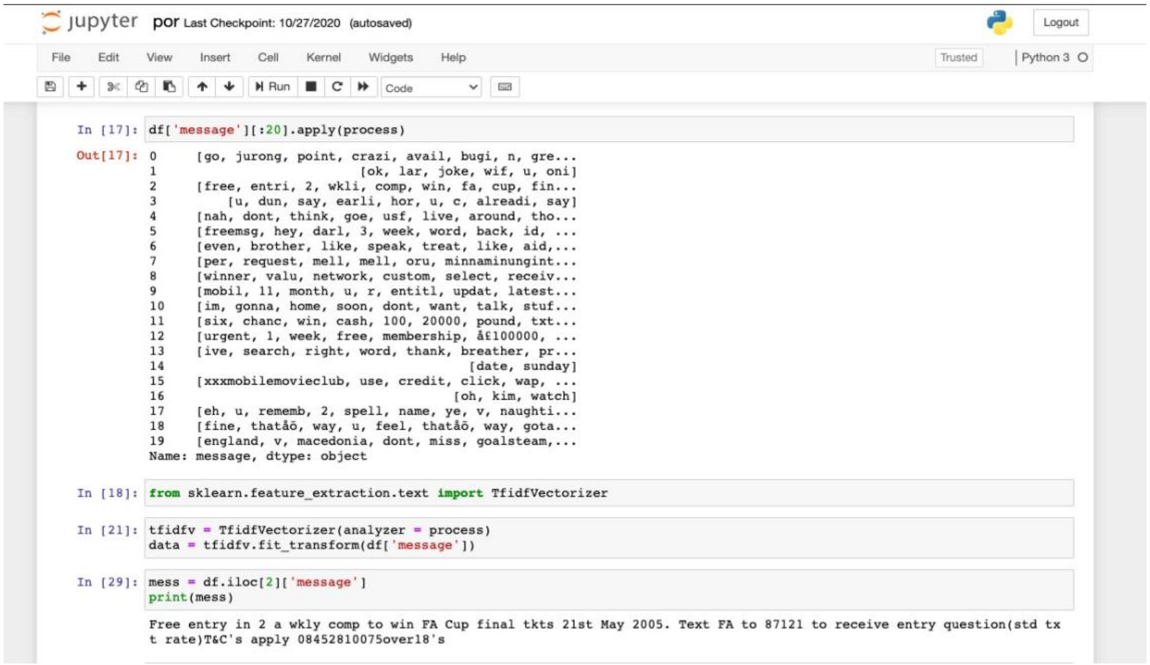
Out[9]:

	label	message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

```
In [10]: df.groupby('label'). describe()
```

Out[10]:

	message			
	count	**unique**	**top**	**freq**
label				
ham	4825	4516	Sorry, I'll call later	30
spam	747	653	Please call our customer service representativ...	4



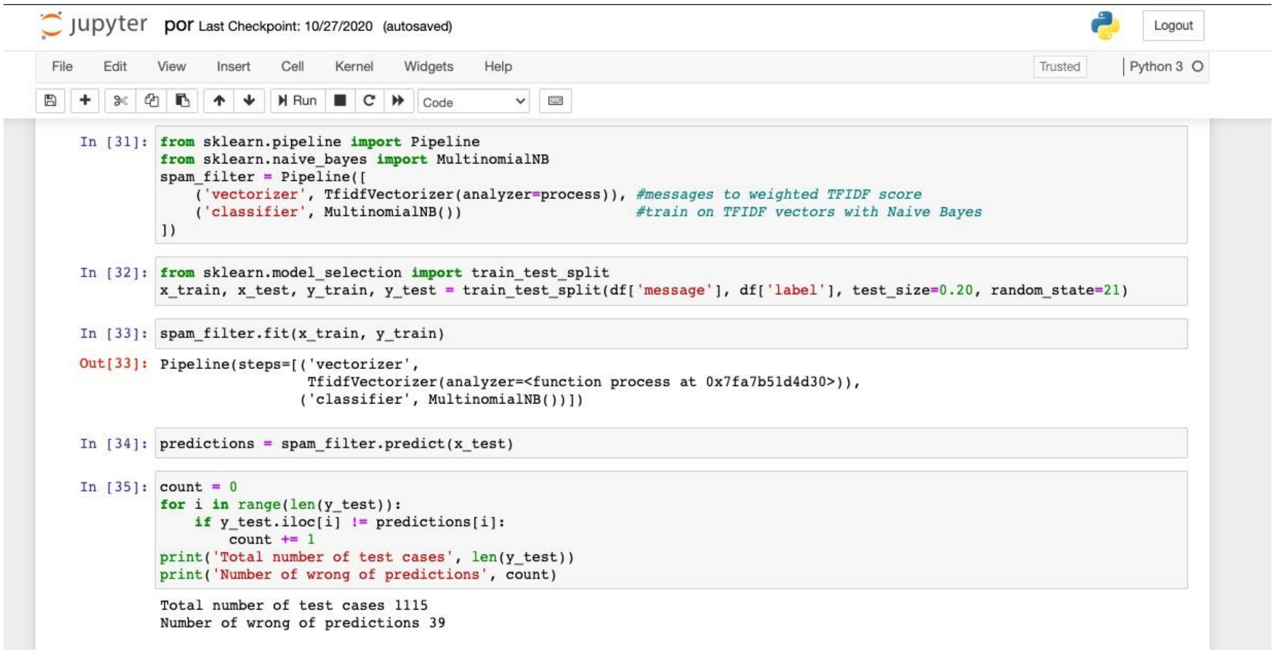
jupyter por Last Checkpoint: 10/27/2020 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [17]: df['message'][:20].apply(process)

Out[17]: 0      [go, jurong, point, crazi, avail, bugi, n, gre...
         1                          [ok, lar, joke, wif, u, oni]
         2      [free, entri, 2, wkli, comp, win, fa, cup, fin...
         3          [u, dun, say, earli, hor, u, c, alreadi, say]
         4      [nah, dont, think, goe, usf, live, around, tho...
         5      [freemsg, hey, darl, 3, week, word, back, id, ...
         6      [even, brother, like, speak, treat, like, aid,...
         7      [per, request, mell, mell, oru, minnaminungint...
         8      [winner, valu, network, custom, select, receiv...
         9      [mobil, 11, month, u, r, entitl, updat, latest...
         10     [im, gonna, home, soon, dont, want, talk, stuf...
         11     [six, chanc, win, cash, 100, 20000, pound, txt...
         12     [urgent, 1, week, free, membership, å£100000, ...
         13     [ive, search, right, word, thank, breather, pr...
         14                                       [date, sunday]
         15     [xxxmobilemovieclub, use, credit, click, wap, ...
         16                                     [oh, kim, watch]
         17     [eh, u, rememb, 2, spell, name, ye, v, naughti...
         18     [fine, thatåõ, way, u, feel, thatåõ, way, gota...
         19     [england, v, macedonia, dont, miss, goalsteam,...
         Name: message, dtype: object
```

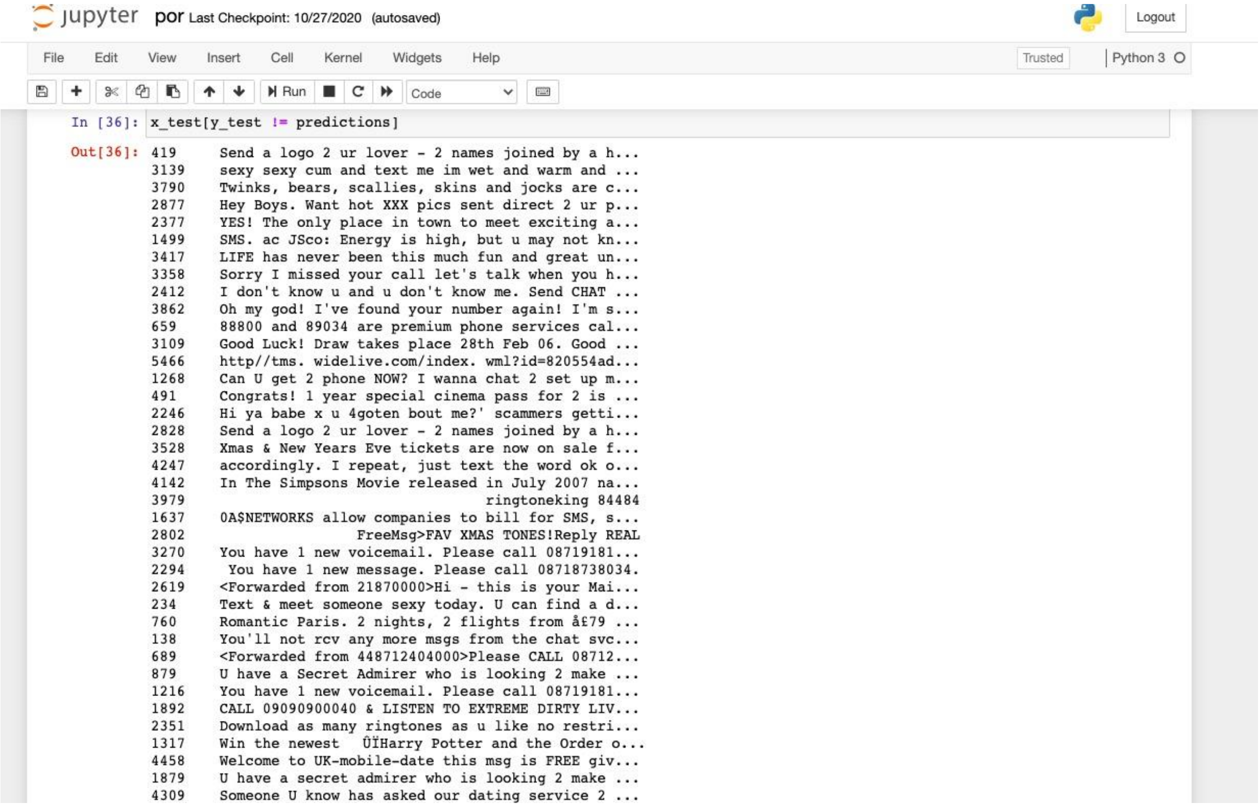
```
In [18]: from sklearn.feature_extraction.text import TfidfVectorizer
```

```
In [21]: tfidfv = TfidfVectorizer(analyzer = process)
         data = tfidfv.fit_transform(df['message'])
```

```
In [29]: mess = df.iloc[2]['message']
         print(mess)

         Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std tx
         t rate)T&C's apply 08452810075over18's
```

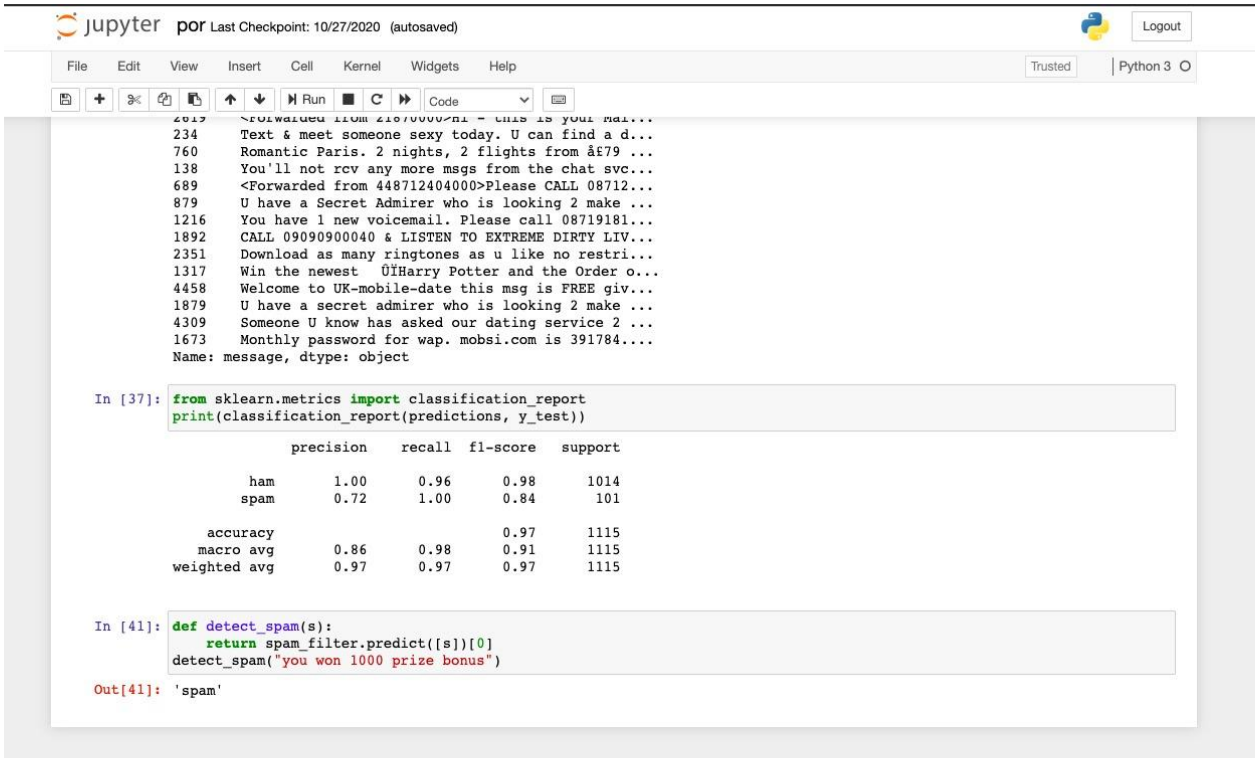


```
In [31]: from sklearn.pipeline import Pipeline
         from sklearn.naive_bayes import MultinomialNB
         spam_filter = Pipeline([
             ('vectorizer', TfidfVectorizer(analyzer=process)), #messages to weighted TFIDF score
             ('classifier', MultinomialNB())                    #train on TFIDF vectors with Naive Bayes
         ])

In [32]: from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test = train_test_split(df['message'], df['label'], test_size=0.20, random_state=21)

In [33]: spam_filter.fit(x_train, y_train)

Out[33]: Pipeline(steps=[('vectorizer',
                          TfidfVectorizer(analyzer=<function process at 0x7fa7b51d4d30>)),
                         ('classifier', MultinomialNB())])

In [34]: predictions = spam_filter.predict(x_test)

In [35]: count = 0
         for i in range(len(y_test)):
             if y_test.iloc[i] != predictions[i]:
                 count += 1
         print('Total number of test cases', len(y_test))
         print('Number of wrong of predictions', count)

         Total number of test cases 1115
         Number of wrong of predictions 39
```



```
In [36]: x_test[y_test != predictions]

Out[36]: 419     Send a logo 2 ur lover - 2 names joined by a h...
         3139    sexy sexy cum and text me im wet and warm and ...
         3790    Twinks, bears, scallies, skins and jocks are c...
         2877    Hey Boys. Want hot XXX pics sent direct 2 ur p...
         2377    YES! The only place in town to meet exciting a...
         1499    SMS. ac JSco: Energy is high, but u may not kn...
         3417    LIFE has never been this much fun and great un...
         3358    Sorry I missed your call let's talk when you h...
         2412    I don't know u and u don't know me. Send CHAT ...
         3862    Oh my god! I've found your number again! I'm s...
         659     88800 and 89034 are premium phone services cal...
         3109    Good Luck! Draw takes place 28th Feb 06. Good ...
         5466    http//tms. widelive.com/index. wml?id=820554ad...
         1268    Can U get 2 phone NOW? I wanna chat 2 set up m...
         491     Congrats! 1 year special cinema pass for 2 is ...
         2246    Hi ya babe x u 4goten bout me?' scammers getti...
         2828    Send a logo 2 ur lover - 2 names joined by a h...
         3528    Xmas & New Years Eve tickets are now on sale f...
         4247    accordingly. I repeat, just text the word ok o...
         4142    In The Simpsons Movie released in July 2007 na...
         3979                                   ringtoneking 84484
         1637    0A$NETWORKS allow companies to bill for SMS, s...
         2802                    FreeMsg>FAV XMAS TONES!Reply REAL
         3270    You have 1 new voicemail. Please call 08719181...
         2294     You have 1 new message. Please call 08718738034.
         2619    <Forwarded from 21870000>Hi - this is your Mai...
         234     Text & meet someone sexy today. U can find a d...
         760     Romantic Paris. 2 nights, 2 flights from å£79 ...
         138     You'll not rcv any more msgs from the chat svc...
         689     <Forwarded from 448712404000>Please CALL 08712...
         879     U have a Secret Admirer who is looking 2 make ...
         1216    You have 1 new voicemail. Please call 08719181...
         1892    CALL 09090900040 & LISTEN TO EXTREME DIRTY LIV...
         2351    Download as many ringtones as u like no restri...
         1317    Win the newest  ÛÏHarry Potter and the Order o...
         4458    Welcome to UK-mobile-date this msg is FREE giv...
         1879    U have a secret admirer who is looking 2 make ...
         4309    Someone U know has asked our dating service 2 ...
```

RESULTS :- Spam and Ham messages are successfully identified

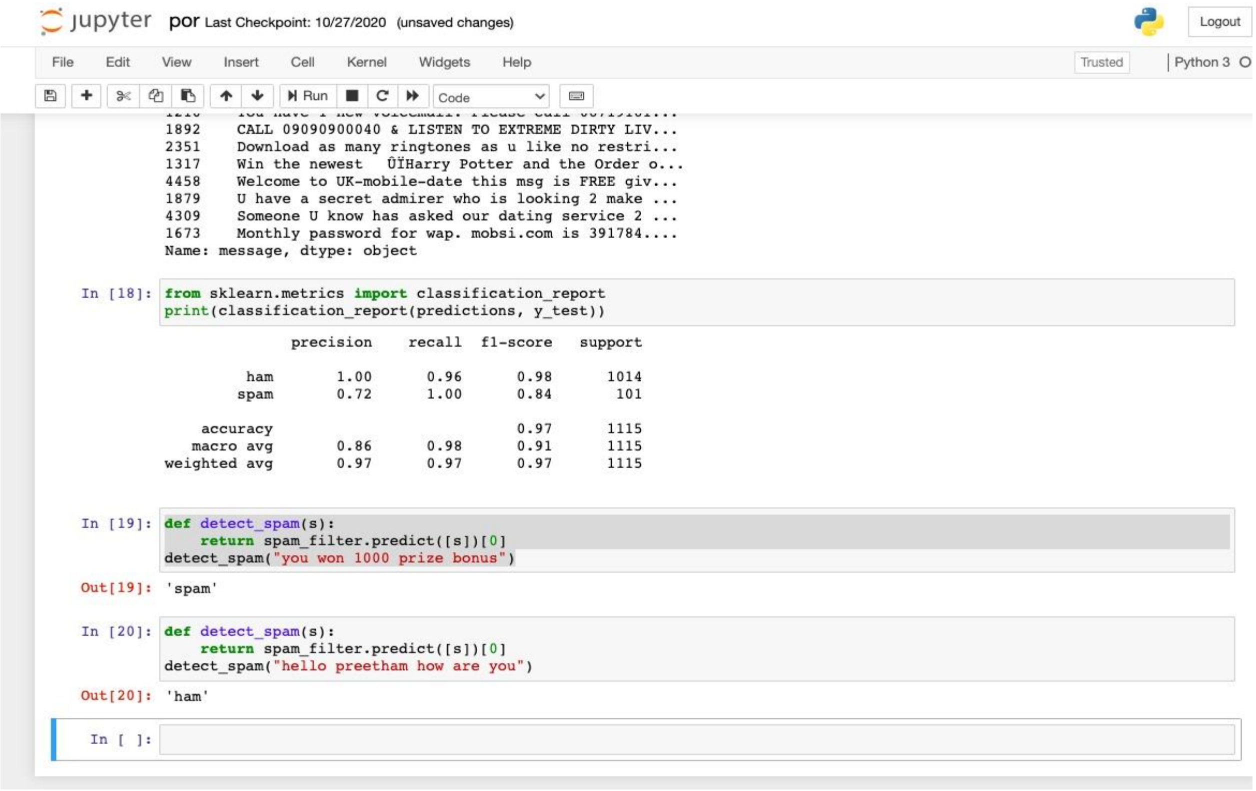


```
2619     <Forwarded from 21870000>Hi - this is your Mai...
234      Text & meet someone sexy today. U can find a d...
760      Romantic Paris. 2 nights, 2 flights from å£79 ...
138      You'll not rcv any more msgs from the chat svc...
689      <Forwarded from 448712404000>Please CALL 08712...
879      U have a Secret Admirer who is looking 2 make ...
1216     You have 1 new voicemail. Please call 08719181...
1892     CALL 09090900040 & LISTEN TO EXTREME DIRTY LIV...
2351     Download as many ringtones as u like no restri...
1317     Win the newest   ÛÏHarry Potter and the Order o...
4458     Welcome to UK-mobile-date this msg is FREE giv...
1879     U have a secret admirer who is looking 2 make ...
4309     Someone U know has asked our dating service 2 ...
1673     Monthly password for wap. mobsi.com is 391784....
Name: message, dtype: object
```

```
In [37]: from sklearn.metrics import classification_report
         print(classification_report(predictions, y_test))
```

```
              precision    recall  f1-score   support

         ham       1.00      0.96      0.98      1014
        spam       0.72      1.00      0.84       101

    accuracy                           0.97      1115
   macro avg       0.86      0.98      0.91      1115
weighted avg       0.97      0.97      0.97      1115
```

```
In [41]: def detect_spam(s):
             return spam_filter.predict([s])[0]
         detect_spam("you won 1000 prize bonus")

Out[41]: 'spam'
```



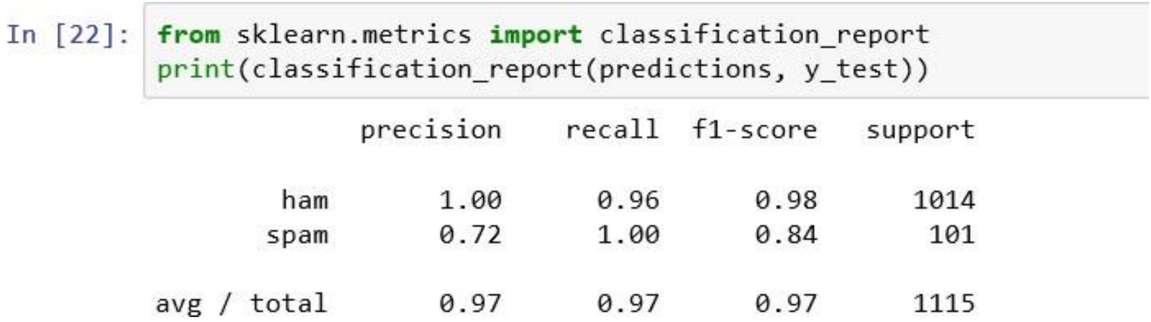
```
1216     You have 1 new voicemail. Please call 08719181...
1892     CALL 09090900040 & LISTEN TO EXTREME DIRTY LIV...
2351     Download as many ringtones as u like no restri...
1317     Win the newest   ÛÏHarry Potter and the Order o...
4458     Welcome to UK-mobile-date this msg is FREE giv...
1879     U have a secret admirer who is looking 2 make ...
4309     Someone U know has asked our dating service 2 ...
1673     Monthly password for wap. mobsi.com is 391784....
Name: message, dtype: object
```

```
In [18]: from sklearn.metrics import classification_report
         print(classification_report(predictions, y_test))
```

```
              precision    recall  f1-score   support

         ham       1.00      0.96      0.98      1014
        spam       0.72      1.00      0.84       101

    accuracy                           0.97      1115
   macro avg       0.86      0.98      0.91      1115
weighted avg       0.97      0.97      0.97      1115
```

```
In [19]: def detect_spam(s):
             return spam_filter.predict([s])[0]
         detect_spam("you won 1000 prize bonus")

Out[19]: 'spam'

In [20]: def detect_spam(s):
             return spam_filter.predict([s])[0]
         detect_spam("hello preetham how are you")

Out[20]: 'ham'

In [ ]:
```

Accuracy :-



```
In [22]: from sklearn.metrics import classification_report
         print(classification_report(predictions, y_test))
```

```
             precision    recall  f1-score   support

        ham       1.00      0.96      0.98      1014
       spam       0.72      1.00      0.84       101

avg / total       0.97      0.97      0.97      1115
```

Conclusion

In this paper it has been shown that it is possible to attain very good classification performance employing a word-position-based variant of naive Bayes. The simplicity and low time complexness of the algorithm makes naive Bayes a decent choice for end-user applications. This spam classifier is implemented by Naive Bayes Model, an easy but very efficient solution in spam classification problem. In brief, Naive Bayes treats each feature independent from one another, making inference very efficient. This code runs quite well with the accuracy of 98.31% on training sample and 97.81% on test sample. We also tried with Random Forest and XGBoost, but the accuracy was low, so the conclusion is that the Naïve Bayes is the best classifier till now.

LINK FOR THE CODE AND DATASET :

https://github.com/vinod1209op/NATURAL_LANGUAGE_PROCESSING_PROJECT

References

[1] H. Zhang and D. Li, "Naïve Bayes Text Classifier," *2007 IEEE International Conference on Granular Computing (GRC 2007)*, Fremont, CA, 2007, pp. 708-708, doi: 10.1109/GrC.2007.40.

[2] A. K. Uysal, S. Gunal, S. Ergin and E. S. Gunal, "A novel framework for SMS spam filtering," 2012 International Symposium on Innovations in Intelligent Systems and Applications, Trabzon, 2012, pp. 1-4, doi: 10.1109/INISTA.2012.6246947.

[3] B. Tang, H. He, P. M. Baggenstoss and S. Kay, "A Bayesian Classification Approach Using Class-Specific Features for Text Categorization," in IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 6, pp. 1602-1606, 1 June 2016, doi: 10.1109/TKDE.2016.2522427.

[4] Chen, Jingnian, et al. "Feature selection for text classification with Naïve Bayes." Expert Systems with Applications 36.3 (2009): 5432-5435. https://doi.org/10.1016/j.eswa.2008.06.054

[5] Zhang, Min-Ling, José M. Peña, and Victor Robles. "Feature selection for multi-label naive Bayes classification." Information Sciences 179.19 (2009): 3218-3229. https://doi.org/10.1016/j.ins.2009.06.010

[6] Devender, Birru, Korra Srinivas, and Ch Tulasi Ratna Mani. "Detecting Spam Zombies By Monitoring Outgoing Messages." May 2016

[7] M. Sasaki and H. Shinnou, "Spam detection using text clustering," 2005 International Conference on Cyberworlds (CW'05), Singapore, 2005, pp. 4 pp.-319, doi: 10.1109/CW.2005.83.

[8] N. Jindal and B. Liu, "Analyzing and Detecting Review Spam," Seventh IEEE International Conference on Data Mining (ICDM 2007), Omaha, NE, 2007, pp. 547-552, doi: 10.1109/ICDM.2007.68.

[9] Q. Xu, E. W. Xiang, Q. Yang, J. Du and J. Zhong, "SMS Spam Detection Using Non-content Features," in IEEE Intelligent Systems, vol. 27, no. 6, pp. 44-51, Nov.-Dec. 2012, doi: 10.1109/MIS.2012.3.

GITHUB REPO FOR HANDS ON SESSION:

VINOD: https://github.com/vinod1209op/natural_language_processing

PREETHAM: https://github.com/preetham1902/NLP

RITESH: https://github.com/KRiteshchowdary/NLP-Hands-ON

THANK YOU SHARMILA BANU K MA’AM