

**SENTIMENT LEVEL ANALYSIS FOR DETECTING SPAM
EMAIL**

NWE NWE AYE

M.I.Sc.

June, 2022

**SENTIMENT LEVEL ANALYSIS FOR DETECTING SPAM
EMAIL**

By

NWE NWE AYE

D.C.Sc.

**Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Information Science
(M.I.Sc.)**

University of Computer Studies, Yangon

June 2022

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincere thanks to those who helped me with various aspects of conducting research and writing this thesis. To complete this thesis, many things are needed like my hard work as well as the supporting of many people.

First and foremost, I would like to express my deepest gratitude and my thanks to **Dr. Mie Mie Khin**, Rector, the University of Computer Studies, Yangon, for her kind permission to submit this thesis.

My heartfelt thanks and respect go to **Dr. Soe Lin Aung**, Pro-rector, University of Computer Studies (Magway), for his invaluable and administrative support.

I would like to express my appreciation to **Dr. Moe Thu Zar Htwe**, Professor, Faculty of Computer Systems and Technologies, University of Computer Studies (Magway), for her superior suggestion, administrative supports and encouragement during my academic study.

My thanks and regards go to my supervisor, **Dr. Amy Aung**, Professor, Faculty of Information Science, University of Computer Studies (Magway), for her support, guidance, supervision, patience and encouragement during the period of study towards completion of this thesis.

I also wish to express my deepest gratitude to **Daw Aye Aye Khine**, Associate Professor, English Department, the University of Computer Studies, Yangon, for her editing this thesis from the language point of view.

Moreover, I would like to extend my thanks to all my teachers who taught me throughout the master's degree course and my friends for their cooperation.

Last but not least, I especially thank my parents, all of my colleagues, and friends for their encouragement and help during my thesis.

ABSTRACT

Since web is expanding day by day and people generally rely on web for communications, e-mails are the fastest way to send information from one place to another. Nowadays all the transactions and all the communications whether general or of business have been taking place through e-mails. E-mail is an effective tool for communication as it saves a lot of time and cost. But e-mails are also affected by attacks which include Spam Mails. Spam is the use of electronic messaging systems to send bulk data. Spam is flooding the Internet with many copies of the same message, in an attempt to force the message on people who would not otherwise choose to receive it. To avoid this problem mentioned above, this system is designed to filter the spam message by using sentiment analysis technique and machine learning approach. The proposed system uses spam words database, sentiwordnet3.0, and Naïve Bayes classifier is used for training and testing the features and also evaluating the sentimental polarity. This system is implemented by using Python3.10

Keywords: Feature Extraction, text classification, Sentiment analysis, SentiWordNet, supervised learning

CONTENTS

	Page
ACKNOWLEDGEMENTS	i
ABSTRACT	ii
CONTENTS	iii
LIST OF FIGURES	v
LIST OF TABLES	vi
LIST OF ALGORITHMS	vii
CHAPTER 1 INTRODUCTION	1
1.1 Objectives of the Thesis	1
1.2 Motivations	2
1.3 Related works	2
1.4 Organization of the Thesis	3
CHAPTER 2 BACKGROUND THEORY	5
2.1 Spam Mail Detection	6
2.2 Bayesian spam filtering	7
2.2.1 Process	7
2.2.2 Mathematical foundation	8
2.2.3 Advantages of the existing Bayesian method	8
2.2.4 Limitations of the existing Bayesian method	9
2.3 Variations on Data Classification	10
2.3.1 Rare Class Learning	10
2.3.2 Distance Function Learning	11
2.3.3 Ensemble Learning for Data Classification	12
CHAPTER 3 PROBABILISTIC MODELS FOR CLASSIFICATION	15
3.1 Naïve Bayes Classification	17

3.1.1	Bayes' Theorem and Preliminary	17
3.1.2	Naïve Bayes Classifier	20
3.1.3	Maximum Likelihood Estimates for Naïve Baye Model	22
3.2	Probabilistic and Naïve Bayes Classifiers	24
3.2.1	Bernoulli Multivariate Model	26
3.2.2	Multinomial Distribution	30
3.2.3.	Multinomial Naïve Bayes	31
3.3	Sentiment Analysis in Social Network	
3.1.1	Sentiment analysis and Natural Language Processing	34
3.1.2	Sentiment Analysis Models	35
3.1.3	Bag-of-words (BOW)	35
3.1.4	Lexicon Based Approach	35
3.1.5	Part of Speech (POS) Tagging	35
3.4	SentiWordNet3.0	36
CHAPTER 4	SYSTEM DESIGN AND IMPLEMENTATION	33
4.1	FEATURE EXTRACTION	34
4.2	Probabilistic Naïve Bayes Classification	38
4.3	Multinomial Naïve Bayes	38
4.4	Overview of the System Flow Diagram	39
4.5	System Implementation	41
4.5.1	Description of Input Data Source	52
4.5.2	Performance evaluation	60
CHAPTER 5	CONCLUSION	52
5.1	Conclusion	52
5.2	Advantages of the System	53
5.3	Limitations and Further Extensions	53

REFERENCES

LIST OF FIGURES

FIGURE		PAGE
3.1	Preparation step for sent WordNet	37
4.1	System Flow Diagram for Multinomial Naïve Bayes Classification	40
4.2	System Flow Diagram for Classifying Sentiment Score	41
4.3	SMS email message data source	43
4.4	Import Training Data	43
4.5	Before Tokenization and Stopword Removal	44
4.6	Pre-processing of Training Data	45
4.7	Accuracy Result of Multinomial Naïve Bayes Classifier	45
4.8	Import Testing Data	46
4.9	Before Tokenization and Stopword Removal for Testing Data	47
4.10	Pre-processing of Testing Data	47
4.11	Classify Result of Testing Data	48
4.12	Object Score Result by using MNB	48
4.13	Object Score Result by using SentiWordNet3.0	49
4.14	Accuracy Result for MNB for MNB and SentiWordNet3.0 Score	50

LIST OF TABLES

TABLE		PAGE
4.1	Training data of SMS spam email message	35
4.2	Tokenization, remove stopword and stemming of pre-processing training data	35
4.3	Term Frequency (TF) of each keyword	36
4.4	Inverse Document Frequency (IDF) of each keyword	36
4.5	Weight of each keyword	37
4.6	Weight of each keyword for each document	37
4.7	Total weight of keywords for each document	38

LIST OF ALGORITHMS

Algorithm	PAGE
Algorithm 3.1	32
Algorithm 3.2	36

CHAPTER 1

INTRODUCTION

Billions of individuals all over the planet utilize miniature writing for a blog locale to impart their thoughts and insights about occasions, items and people. Web-based entertainment contains billions of short casual message records, e.g., tweets, SMS messages, messages, and audits. This immense assortment of miniature sites has made the social message in region that incorporates different undertakings, e.g., feeling examination, assessment mining, spam identification and survey investigation. Feeling examination targets dissecting the casually composed text by every citizen, and removing individual conclusions about items and occasions as well as about different people groups. Nonetheless, working with informal organization messages is very difficult in light of the fact that these messages might contain spam and manhandling contents. In this period of informal organizations, plan and improvement of fitting assessment mining and spam identification devices assume a significant part in the examination of general conclusions. The framework has zeroed in on the opinion characterization in message and thinks about highlight choice in feeling examination in a significant part. This framework utilizes the spam message dataset for preparing and utilize the Naïve Bayesian method for grouping.

Email is one of the notable correspondence administrations in which a message sends electronically. Spam messages incorporate ads, free administrations, advancements, grants, and so forth. Spam words are or express that email suppliers have recognized to address dubious or vindictive movement. To safeguard the beneficiaries from such activity when they get an email, spam channels are set off to distinguish whether any spam words have been utilized. Should an email be delegated spam, it will go directly to the spam mail message characterization from Sentiment Analysis.

1.1 Objectives of the Thesis

The main objectives of this thesis are:

- To extract feature score from email messages
- To apply multinomial Naïve Bayes approach in feature selection

- To classify the emails/ messages into positive and negative senses.
- To know which emails/ messages are spam or ham (not - spam)

1.2 Motivations

Electronic Mail is the “killer network application”. It is omnipresent and unavoidable. In a generally short time span, the Internet has become unavoidably and profoundly settled in our cuttingedge society basically because of the force of its correspondence substrate connecting individuals and associations all over the planet. Much work on email innovation has zeroed in on making email simple to utilize, allowing a wide assortment of data and data types to be helpfully, dependably, sent all through the Internet. Nonetheless, the investigation of the immense storage facility of email content aggregated or created by individual clients has gotten generally little consideration other than for explicit errands, for example, spam and infection sifting. Clients in the email ceaselessly get spam and they cause problems burning through their time and furthermore hurtful messages can hurt the PCs.

This framework presents a carried out system for information mining/assessment mining conduct models from email information. Various AI and irregularity discovery calculations are implanted to the mailing frameworks by numerous specialists and the client's email conduct to group email for different errands. There are various techniques for identification of spam through email. The fundamental objective of this framework is to foster a recognition framework that beats the discovery of spam, ham and wrongly ordered spam, for example need is to work on the exactness of the proposed strategy contrasted with the other existing techniques. Thus, to restate, this proposition likewise bargains the exactness and cycle timing in light of prioritization of distinguishing email messages.

1.3 Related Works

Lop mudra Dey, "Feeling Analysis of Review Datasets Using Naïve Bayes' and K-NN Classifier", Department of Computer Science and Engineering, Heritage Institute of

Technology, Kolkata, India, 2016 [1]. This framework was to assess the presentation for feeling order with regards to exactness, accuracy and review. This framework looked at two Naïve Bayes and K-NN for feeling grouping of the film surveys and inn audits. The outcomes showed that the classifiers improved results for the film audits with the Naïve Bayes and giving above 80% exactness's and outflanking than the k-NN approach. For the inn surveys, the exactness's are a lot of lower and both the classifiers comparable outcomes. This framework can say Naive Bayes' classifier can be utilized effectively to investigate film audits.

"Execution Analysis on Mail Classification with Multilayer Perceptron (MLP)", centers around email classifier with Multilayer Perceptron (MLP) approach for spam and ham sends arrangement. The framework was utilized term recurrence and backwards archive recurrence (tf-idf) and fisher score include determination techniques at preprocessing. These strategies permitted choosing applicable elements and adding benefit as far as ad lib in precision and diminished time intricacy to email order framework [2].

Paras Seth [3], "SMS spam identification and correlation of different AI calculations", 2017, International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN). This framework grouped the SMS spam messages as spam or ham (not spam). This framework performed tests and examination with Naïve Bayes calculation, Random Forest calculation and Logistic Regression calculation. Innocent Bayes outflanks Random Forest and Logistic Regression and accomplished a high exactness of 98.445%.

1.4 Organization of the Thesis

This thesis is organized in five chapters. In Chapter **1**, the introduction of the system, objectives of the thesis, related works and thesis organization are described. **Chapter 2** presents the background theory. **Chapter 3** discusses the data classification techniques. **Chapter 4** expresses the design and implementation are presented with the figures. Moreover, Data collection, Data cleaning, Preprocessing and feature extract. Term Frequency-Inverse Document Frequency is used feature extraction and Multinomial naïve bayes classifier classifies the database on the training data into spam or ham. Finally, Chapter 5 concludes with the advantages of the system, its limitations and further extensions.

CHAPTER 2

BACKGROUND THEORY

Electronic mail is one of the most famous types of correspondences today. The shockingly quick acknowledgment of this correspondence medium is best exemplified by the sheer number of flow clients, assessed to be as near 3/4 of a billion people, and developing [1]. This type of correspondence enjoys the basic benefit of being practically quick, instinctive to utilize, and costing essentially nothing per message. The ongoing email framework depends on the SMTP convention RFC 821 and 822 created in 1982 and stretched out in RFC 2821 in 2001[2]. This framework characterizes a typical norm to join the different informing conventions in presence before 1982. It permitted clients the capacity to trade messages with each other utilizing a framework in light of the SMTP convention and email addresses. These conventions permitted messages to pass starting with one client then onto the next, making it reasonable and simple for various clients to convey autonomous of the specialist co-op or the client application.

Messages generally are held in information records or envelopes with no organized relationship (at documents), making anything over a catchphrase search extremely sluggish. Clients might decide to move messages into time-requested sub-organizers of related messages. Studies have shown that run of the mill clients rapidly produce somewhere in the range of tens to many organizers in a moderately short measure of time. Finding a specific past message across these sub-envelopes can undoubtedly transform into an overwhelming undertaking. Not exclusively is the email the subject of search, yet additionally the organizer in which it could have been set. Inside these at record organizers, connections are encoded in MIME design making examination of something besides basic filename near unimaginable. Ongoing apparatuses have been delivered which permit ordering and looking through nearby information including messages and portions of connections. Well beyond essentially sending messages, studies have shown that numerous clients have rapidly taken on email to various errands including task appointment, report chronicling, individual contact rundown, and update and planning [4].

For instance, run of the mill clients will utilize their INBOX or primary message region, as an dynamic "plan for the day", leaving current messages on the first spot on the

list. In any event, for efficient clients who generally keep up with past messages in suitable sub-envelopes, there stays the chance of margin time, and thus, over a somewhat brief timeframe, explosions of email can rapidly collect making association of these new messages a sluggish and troublesome undertaking. Notwithstanding these association issues, the Achilles impact point of the ongoing email framework is its general simplicity of misuse. The conventions depended with the understanding that email clients wouldn't mishandle the honor of sending messages to one another. The abuse and maltreatment of the email framework has taken on many structures throughout the long term. Normal abuse incorporates fashioned messages, undesirable messages (spam), fake plans, and fraud and misrepresentation through "Phishing" messages. Misuse incorporates infection and worm connections, and email DOS assaults. The shared factor among this large number of classifications is they exploit the email framework's absence of controls and confirmation of shipper and beneficiary (an acquire issue in a decentralized framework). Email isn't consent based, and one can essentially communicate something specific without earlier endorsement. Clients ought not be supposed to take care of a maintenance bill for basically opening an email which appeared to have started from a companion's email address, caricature by a victimizer.

Consequently, distinguishing spam is one of the main rules. In this proposal paper, the work is to recognize the spam in email. There are some current spam sifting strategies like Bayesian spam separating, Improved Bayesian spam separating, Naïve Bayesian spam sifting, Meta spam sifting. The researcher contrasts the current techniques and the proposed strategy and figures out the exactness and misleading positive, for example, wrongly identified spam of the proposed strategy with the previously mentioned existing techniques. It is seen that utilizing the post sifting strategy for client customization expands the precision of the proposed technique. The technique for process prioritization is additionally utilized to distinguish the exactness of the proposed strategy. In the event that a cycle can recognize spam more much of the time than the other interaction than the prioritization of the cycle is naturally refreshed and subsequently the exactness likewise increments fundamentally with the update of the interaction prioritization. To reiterate, one might say that the proposed technique for spam separating is awesome and overpowers the other existing strategies with regards to exactness and misleading positive discovery.

2.1 Spam Mail Detection

Now as the number of email users is increasing day by day, so is the number of spam in the inbox. There are different methods for detection of spam. The most well-known of these techniques are Bayesian, Improved Bayesian, Naïve Bayesian, Meta spam filtering, Grey list method for detection of spam. The advantage of Bayesian spam filtering is that it can be trained on per-user basis and it can perform particularly well in avoiding false positives, where legitimate email is incorrectly classified as spam. The main disadvantage is that spammer tactics include insertion of random innocuous words that are not normally associated with spam. For the improved Bayesian method, the main advantage is that the risk of loss factor is reduced. The disadvantage for the method is that calculating the weighting factor is time consuming and costly. The advantage of Naïve Bayesian spam filtering is that it only requires a small amount of training data to estimate the parameters [5]. The disadvantage of the method is that the dependence of the class conditional independence among these cannot be modeled. The advantage of Meta spam filtering is that TCP/IP blocking is used to find malicious email address, while the disadvantage of the method is that definitions of spam should be agreed on before testing. In the Greylist method, the advantage of the method is that it requires no additional configuration from user end while the disadvantage of the method is that it delays much of the mail from non-white listed mail servers.

So, the goal is to propose a method with higher accuracy and also provide a users' (receivers) customization in proposed model. So, this system proposed an efficient proposed method named as Multi nominal Naïve Bayes which will drive away the disadvantages of the existing method. The method has greater accuracy in order to detect email spam.

2.2 Bayesian spam filtering

It is known as statistical spam filtering method. It utilizes a guileless Bayes classifier to recognize spam email. Bayesian classifiers work by corresponding the utilization of tokens (regularly words, or once in a while different things), with spam and non-spam messages and afterward utilizing Bayesian deduction to compute a likelihood

that an email is or alternately isn't spam. Bayesian spam separating is an extremely strong strategy for managing spam, that can fit itself to the email needs of individual clients, and gives low misleading positive spam identification rates that are by and large satisfactory to clients.

The principal known mail-separating project to utilize a Bayes classifier was Jason Rennie's ifile program, delivered in 1996. The program was utilized to sort mail into organizers. The principal academic distribution on Bayesian spam sifting was by Sahami et al. in 1998[7]. That work was before long conveyed in business spam channels. Notwithstanding, in 2002, Paul Graham had the option to significantly work on the bogus positive rate, so it very well may be utilized all alone as a solitary spam channel.

2.2.1 Process

Particular words have particular probabilities of occurring in spam email and in legitimate email. For example, most email clients will every now and again experience "Viagra" in spam email, however will rarely see it in other email. The channel doesn't have the foggiest idea about these probabilities ahead of time, and must initially be prepared so it can develop them. To prepare the channel, the client should physically demonstrate regardless of whether another email is spam. For all words in each preparing email, the channel will change the probabilities that each word will show up in spam or real email in its data set. For example, Bayesian spam channels will commonly have taken in an extremely high spam likelihood for the words "Viagra" and "renegotiate", yet an exceptionally low spam likelihood for words seen exclusively in genuine email, like the names of loved ones.

Subsequent to preparing, the word probabilities (otherwise called probability capabilities) are utilized to figure the likelihood that an email with a specific arrangement of words in it has a place with one or the other class. Each word in the email adds to the email's spam likelihood, or just the most fascinating words. This commitment is known as the back likelihood and is registered utilizing Bayes' hypothesis. Then, at that point, the email's spam likelihood is registered over all words in the email, and assuming the all out surpasses a specific edge (say 95%), the channel will check the email as a spam.

As in some other spam sifting procedure, email set apart as spam can then be consequently moved to a "Garbage" email envelope, or even erased through and through. Some product carries out isolation systems that characterize a time period during which the client is permitted to survey the product's choice. The underlying preparation can for the most part be refined when wrong decisions from the product are distinguished (bogus upsides or misleading negatives). That permits the product to powerfully adjust to the steadily advancing nature of spam. Some spam channels consolidate the consequences of both Bayesian spam sifting and different heuristics (pre-characterized rules about the items, taking a gander at the message's envelope, and so on), bringing about considerably higher separating exactness, at times at the expense of seductive nature.

2.2.2 Mathematical foundation

Bayesian email channels exploit Bayes' hypothesis. Bayes' hypothesis involves a few times with regards to spam:

- A first time, to register the likelihood that the message is spam, realizing that a given word shows up in this message;
- All a subsequent time, to register the likelihood that the message is spam, thinking about its words (or a pertinent subset of them);
- In a last time, to manage intriguing words.

2.2.3 Advantages of the existing Bayesian method

One of the principal benefits of Bayesian spam sifting is that it tends to be prepared on a for each client premise. The spam that a client gets is many times connected with the web-based client's exercises. For instance, a client might have been bought into an internet based pamphlet that the client views as spam. This internet based pamphlet is probably going to contain words that are normal to all bulletins, for example, the name of the pamphlet and its starting email address. A Bayesian spam channel will ultimately dole out a higher likelihood in light of the client's particular examples.

The genuine messages a client gets will more often than not be unique. For instance, in a professional workplace, the organization name and the names of clients or clients will be referenced frequently. The channel will relegate a lower spam likelihood to messages the word probabilities are remarkable to every client and can develop over the long run with remedial preparation at whatever point the channel mistakenly groups an email. Accordingly, Bayesian spam sifting exactness subsequent to preparing is much of the time better than pre-characterized rules.

It can perform especially well in staying away from misleading up-sides, where genuine email is mistakenly named spam. For instance, on the off chance that the email contains "Nigeria", which is often utilized in Advance charge misrepresentation spam, a pre-characterized rules channel could dismiss it out and out. A Bayesian channel would stamp "Nigeria" as a plausible spam word, however would consider other significant words that generally show real email. For instance, the name of a mate may firmly demonstrate the email isn't spam, which could beat the utilization of the word "Nigeria."

2.2.4 Limitations of the existing Bayesian method

Depending on the implementation, Bayesian spam filtering might be vulnerable to Bayesian harming, a strategy involved by spammers trying to corrupt the viability of spam channels that depend on Bayesian sifting. A spammer rehearsing Bayesian harming will convey messages with a lot of genuine message (accumulated from authentic news or scholarly sources). Spammer strategies incorporate inclusion of irregular harmless words that are not typically connected with spam, consequently diminishing the email's spam score, making it bound to slip past a Bayesian spam channel. Anyway with (for instance) Paul Graham's plan just the main probabilities are utilized, so that cushioning the text out with non-spam-related words doesn't influence the discovery likelihood fundamentally.

Words that regularly show up in enormous amounts in spam may likewise be changed by spammers. For instance, « Viagra » would be supplanted with « Viaagra » or « Viagra » in the spam message. The beneficiary of the message can in any case peruse the changed words, yet every one of these words is met all the more seldom by the bayesian channel, which ruins its way of learning. When in doubt, this spamming strategy doesn't

function admirably, in light of the fact that the determined words end up perceived by the channel very much like the typical ones .

2.3 Variations on Data Classification

Data classification problem have many natural variations. There are relate to one or other little varieties of the standard arrangement issue or are upgrades of grouping with the utilization of extra information. Varieties of the characterization issue are those of interesting study of class and distance capability learning. Upgrades of the information order issue utilize meta-calculations, more information in strategies, for example, move studying and co-preparing, dynamic studying, and human mediation in visual learning. Furthermore, the model subject assessment is a significant one with regards to information characterization. This is on the grounds that the model issue assessment is significant for the plan of successful characterization meta-calculations.

2.3.1 Rare Class Learning

Class learning of unlikely is a significant variety of the characterization issue, and is firmly connected with exception examination. Truth be told, it tends to be viewed as a managed variety of the exception identification issue. In uncommon study of class , the appropriation of the classes is exceptionally imbalanced in the information, and it is regularly more essential to decide the good rating accurately. For instance, consider the situation where characterizing patients into dangerous and typical categories is attractive. In such instances, most of patients might be ordinary, however it is commonly significantly more exorbitant to take for a really threatening patient (misleading negative). Along these lines, misleading negatives are more expensive than bogus up-sides. The issue is firmly connected with cost-touchy studying, since the various misclassification classes has various classes.

The significant distinction with the standard order issue is that the goal capability of the issue should be altered with costs. This gives a few roads that can be utilized to tackle this issue really:

- **Model Weighting:** For this situation, the models are weighted in an unexpected way, contingent on their expense of misclassification. This prompt minor changes in most characterization calculations, which are moderately easy to execute. For instance, in a SVM classifier, the goal capability should be properly weighted with costs, though in a choice tree, the measurement of the split model requirements to weight the models with costs. In a closest neighbor classifier, the k closest neighbors are fittingly weighted while deciding the class with the biggest presence.
- **Model Re-testing:** For this situation, the models are fittingly re-examined, with the goal that uncommon classes are over-inspected, while the ordinary classes are under-examined. A standard classifier is applied to the re-tested information with practically no change. According to a specialized viewpoint, this approach is identical to model weighting. In any case, according to a computational viewpoint, such a methodology enjoys the benefit that the recently re-tested information has a lot more modest size. This is on the grounds that a large portion of the models in the information relate to the ordinary class, which is definitely under-tested, while the uncommon class is commonly just somewhat over-examined.

Numerous varieties of the uncommon class discovery issue are conceivable, in which either instances of a solitary class are accessible, or the typical class is polluted with interesting class models.

2.3.2 Distance Function Learning

Distance function learning is a significant issue that is firmly connected with information order. In this issue it is alluring to relate sets of information occasions to a distance esteem with the utilization of either managed or unaided techniques. For instance, consider the instance of a picture assortment, where the similitude is characterized based on a client focused semantic rule. In such a case, the utilization of standard distance works, for example, the Euclidian measurement may not mirror the semantic similitudes between two pictures well, since they depend on human discernment, and may try and differ from

one assortment to another. Consequently, the most ideal way to resolve this issue is to unequivocally integrate human criticism into the educational experience.

Normally, this criticism is consolidated either as far as sets of pictures with express distance values, or as far as rankings of various pictures to a given objective picture. Such a methodology can be utilized for a wide range of information spaces. This is the preparation information that is utilized for learning purposes.

2.3.3 Ensemble Learning for Data Classification

A meta-algorithm is a grouping technique that re-utilizes at least one right now existing characterization calculation by applying either numerous models for heartiness, or consolidating the consequences of similar calculation with various pieces of the information. The overall objective of the calculation is to acquire additional strong outcomes by consolidating the outcomes from different preparation models either successively or autonomously. The general mistake of an order model relies on the predisposition and difference, notwithstanding the characteristic commotion present in the information. The predisposition of a classifier relies on the way that the choice limit of a specific model may not compare to the genuine choice limit.

For instance, the preparation information might not have a direct choice limit, however a SVM classifier will expect a straight choice limit. The fluctuation depends on the arbitrary varieties in the specific preparation informational index. More modest preparation informational collections will have bigger change. Various types of outfit investigation endeavor to lessen this predisposition and change. The pursuer is alluded to for a magnificent conversation on inclination and fluctuation. Meta-calculations are utilized generally in numerous information mining issues, for example, grouping and exception examination to acquire additional precise outcomes from various information mining issues.

The area of order is the most extravagant one according to the viewpoint of meta-calculations, due to its fresh assessment models and relative simplicity in consolidating the consequences of various calculations. A few instances of well known meta-calculations are as per the following:

- **Supporting: Boosting** is a typical strategy utilized in characterization. The thought is to zero in on progressively troublesome bits of the informational index to make models that can characterize the data of interest in these parts all the more precisely, and afterward utilize the outfit scores over every one of the parts. A hold-out approach is utilized to decide the inaccurately characterized occasions for each piece of the informational index. Hence, the thought is to consecutively decide better classifiers for additional troublesome bits of the information, and afterward consolidate the outcomes to get a meta-classifier, which functions admirably on all pieces of the information.
- **Sacking: Bagging** is a methodology that works with irregular information tests, and joins the outcomes from the models developed utilizing various examples. The preparation models for every classifier are chosen by testing with substitution. These are alluded to as bootstrap tests. This approach has frequently been displayed to give prevalent outcomes in specific situations, however this isn't generally the situation. This approach isn't successful for lessening the inclination, however can decrease the change, on account of the particular arbitrary parts of the preparation information.
- **Irregular Forests: Random woods** are a technique that utilization sets of choice trees on either part with haphazardly produced vectors, or arbitrary subsets of the preparation information, and process the score as an element of these various parts. Normally, the irregular vectors are created from a decent likelihood dispersion. In this way, irregular woods can be made by either arbitrary split choice, or irregular info determination. Irregular woods are intently related to sacking, and truth be told packing with choice trees can be viewed as an extraordinary instance of arbitrary woodlands, as far as how the example is chosen (bootstrapping). On account of irregular backwoods, it is likewise conceivable to make the trees in a lethargic manner, which is custom-made to the current specific test occurrence.
- **Model Averaging and Combination:** This is perhaps of the most widely recognized model utilized in troupe examination. Truth be told, the irregular woods technique examined above is an exceptional instance of this thought.

With regards to the arrangement issue, numerous Bayesian techniques exist for the model blend process. The utilization of various models guarantees that the blunder brought about by the predisposition of a specific classifier doesn't rule the grouping results.

- **Stacking:** Methods, for example, stacking likewise consolidate various models in different ways, like involving a second-level classifier to play out the blend. The result of various first-level classifiers is utilized to make another component portrayal for the second level classifier. These first level classifiers might be picked in various ways, like utilizing different packed away classifiers, or by utilizing different preparation models. To keep away from overfitting, the preparation information should be isolated into two subsets for the first and second level classifiers.
- **Container of Models:** In this approach a "wait" piece of the informational index is utilized to choose the most proper model. The most suitable model is one in which the most elevated precision is accomplished in the held out informational index. Basically, this approach can be seen as a rivalry or prepare off challenge between the various models.

The area of meta-calculations in characterization is extremely rich, and various technique might work better in various scenarios.

CHAPTER 3

PROBABILISTIC MODELS FOR CLASSIFICATION

In machine learning, order is viewed as an occurrence of the regulated learning strategies, i.e., surmising a capability from marked preparing information. The preparation information comprise of a bunch of preparing models, where every model is a couple comprising of an information object (normally a vector) $x = _x1,x2, \dots,xd _$ and an ideal result esteem (commonly a class name) $y \in \{C1,C2, \dots,CK\}$. Given such a bunch of preparing information, the errand of a characterization calculation is to examine the preparation information and produce an induced capability, which can be utilized to order new (up to this point inconspicuous) models by doling out a right class name to every one of them. A model would dole out a given email into "spam" or "non-spam" classes.

A typical subclass of order is probabilistic characterization, and in this part we will zero in on a few probabilistic grouping techniques. Probabilistic order calculations utilize factual surmising to track down the best class for a given model. As well as just doling out the best class like other order calculations, probabilistic characterization calculations will yield a comparing likelihood of the model being an individual from every one of the potential classes. The class with the most elevated likelihood is regularly then chosen as the best class. As a rule, probabilistic characterization calculations enjoys a couple of upper hands over non-probabilistic classifiers: First, it can yield a certainty esteem (i.e., likelihood) related with its chosen class name, and subsequently it can decline in the event that its certainty of picking a specific result is excessively low. Second, probabilistic classifiers can be all the more really integrated into bigger AI undertakings, in a way that to some degree or totally dodges the issue of blunder proliferation.

Inside a probabilistic structure, the central issue of probabilistic order is to gauge the back class likelihood $p(Ck|x)$. In the wake of acquiring the back probabilities, we use choice hypothesis [5] to decide class participation for each new info x . Essentially, there are two manners by which we can gauge the back probabilities.

In the primary case, we center around deciding the class-restrictive probabilities $p(x|Ck)$ for each class Ck exclusively, and surmise the earlier class $p(Ck)$ independently. Then, at that point, utilizing Bayes' hypothesis, we can acquire the back probabilities of

class $p(C_k|x)$. Equally, we can demonstrate the joint appropriation $p(x, C_k)$ straightforwardly and afterward standardize to acquire the back probabilities. As the class-restrictive probabilities characterize the factual cycle that creates the highlights we measure, these methodologies that unequivocally or verifiably model the dissemination of contributions as well as results are known as generative models. In the event that the noticed information are genuinely examined from the statistical model, fitting the boundaries of the generative model to boost the information probability is a typical strategy. In this section, we will present two normal stational models. There are probabilistic Naive Bayes classifier and Hidden Markov model.

One more model class is to straightforwardly demonstrate the back probabilities $p(C_k|x)$ by learning a discriminative capability $f(x) = p(C_k|x)$ that guides input x straightforwardly onto a class name C_k . This approach is frequently alluded to however the discriminative model as all work seems to be put on characterizing the generally speaking discriminative capability with the class-contingent probabilities in thought. For example, on account of two-class issues, $f(x) = p(C_k|x)$ may be ceaseless worth somewhere in the range of 0 and 1, with the end goal that $f < 0.5$ addresses class C_1 and $f > 0.5$ addresses class C_2 . In this part, we will present a few probabilistic discriminative models, including Logistic Regression, a sort of summed up straight models, and Conditional Random Fields.

This section portrays a few essential models and calculations for probabilistic grouping, including the accompanying:

- Guileless Bayes Classifier. A Naive Bayes classifier is a straightforward probabilistic classifier in light of applying Bayes' hypothesis with solid (guileless) freedom suspicions. A decent utilization of Naive Bayes classifier is report grouping.
- Calculated Regression. Calculated relapse is a methodology for anticipating the result of a categorial ward variable in view of at least one noticed factors. The probabilities portraying the potential results are displayed as a component of the noticed factors utilizing a strategic capability.

- **Secret Markov Model.** A Hidden Markov model (HMM) is a basic instance of dynamic Bayesian organization, where the secret states are shaping a chain and just some conceivable incentive for each state can be noticed. One objective of HMM is to derive the secret states as indicated by the noticed qualities and their reliance connections. A vital utilization of HMM is grammatical form labeling in NLP.
- **Contingent Random Fields.** A Conditional Random Field (CRF) is an exceptional instance of Markov irregular field, yet each condition of hub is restrictive on a few noticed values. CRFs can be considered as a kind of discriminative classifiers, as they don't display the circulation over perceptions. Name element acknowledgment in data extraction is one of CRF's applications.

3.1 Naive Bayes Classification

The Naive Bayes classifier is in view of the Bayes' hypothesis, and is especially fit when the dimensionality of the data sources is high. Notwithstanding its straightforwardness, the Naive Bayes classifier can frequently accomplish equivalent execution with some complex order strategies, for example, choice tree and chose brain network classifier. Gullible Bayes classifiers have additionally shown high precision and speed when applied to enormous datasets. In this part, we will momentarily survey Bayes' hypothesis, then, at that point, give an outline of Naive Bayes classifier and its utilization in AI, particularly archive order.

3.1.1 Bayes' Theorem and Preliminary

A generally involved structure for grouping is given by a basic hypothesis of likelihood known as Bayes' hypothesis or Bayes' standard. Before we present Bayes' Theorem, let us first audit two principal rules of likelihood hypothesis in the accompanying structure:

$$p(X) = \sum_Y p(X,Y) \quad (3.1)$$

$$p(X,Y) = p(Y|X)p(X). \quad (3.2)$$

where the principal condition is the total rule, and the subsequent condition is the item rule. Here $p(X, Y)$ is a joint likelihood, the amount $p(Y|X)$ is a contingent likelihood, and the amount $p(X)$ is a negligible likelihood. These two straightforward standards structure the reason for all of the probabilistic theory.

symmetry property $p(X,Y) = p(Y,X)$, so Bayes' theorem define the following equation 3.3,

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)}, \quad (3.3)$$

which assumes a focal part in AI, particularly grouping. Utilizing the total rule, the denominator in Bayes' hypothesis can be communicated as far as the amounts showing up in the numerator.

The denominator in Bayes' hypothesis can be viewed similar to the standardization steady expected to guarantee that the amount of contingent likelihood on $p(Y|X)$, Equation (3.3) over all upsides of Y approaches one.

To study a straightforward guide to all the more likely figure out the fundamental ideas of likelihood hypothesis and the Bayes' hypothesis are described in equation 3.3. Assume that we consider two boxes. There are one red and one white in the box. we have two apples, four lemons, and six oranges in red box. After that we have three apples six lemons, and one orange in the white box. In this box, we arbitrarily pick one of the containers and from that case we haphazardly select a thing, and have seen which kind of thing it is. All the while, we supplant the thing in the case from which it came, and we could envision rehashing this cycle commonly. In this case, we pick the red box 40% of the time and we pick the white box 60% of the time. So we pick a thing to desire object easily.

So probability of random variable Y to define as follow:

$$p(Y = e) = 4/10 \text{ and}$$

$$p(Y = f) = 6/10,$$

where $p(Y = a)$ is the minimal likelihood that we pick the red box, and $p(Y = b)$ is the peripheral likelihood that we pick the white box. Assume that we pick a container indiscriminately, and afterward the likelihood of choosing a thing is the negligible portion of that thing given the chose box, which can be composed as the accompanying contingent probabilities

$$p(X = a|Y = e) = 2/12 \quad (3.4)$$

$$p(X = l|Y = e) = 4/12 \quad (3.5)$$

$$p(X = o|Y = e) = 6/12 \quad (3.6)$$

$$p(X = a|Y = f) = 3/10 \quad (3.7)$$

$$p(X = l|Y = f) = 6/10 \quad (3.8)$$

$$p(X = o|Y = f) = 1/10. \quad (3.9)$$

Note that these probabilities are normalized so that

$$p(X = a|Y = e) + p(X = l|Y = e) + p(X = o|Y = e) = 1$$

and

$$p(X = a|Y = f) + p(X = l|Y = f) + p(X = o|Y = f) = 1.$$

Now assume a thing has been chosen and it is an orange, and we might want to realize which box it came from. This expects that we assess the likelihood dispersion over boxes molded on the character of the thing, while the probabilities in Equation (3.4)- (3.9) delineate the appropriation of the thing adapted on the personality of the crate. In view of Bayes' hypothesis, we can compute the back likelihood by switching the contingent probability

$$P(Y = e|X = o) = \frac{P(X = o|Y = e)P(Y = e)}{P(X = o)} = \frac{\frac{6}{12} * 4/10}{13/50} \dots = \frac{10}{13}$$

So $p(X = o)$ can be calculated by using the sum and product rules for over all probability.

$$p(X = o) = p(X = o|Y = r)p(Y = r) + p(X = o|Y = w)p(Y = w) = \frac{6}{12} \times \frac{4}{10} + \frac{1}{10} \times \frac{6}{10} = \frac{13}{50}.$$

From the sum rule, it then follows that $p(Y = w|X = o) = 1 - 10/13 = 3/13$. Overall cases, we are keen on the probabilities of the classes given the information tests.

Assume we utilize arbitrary variable Y to indicate the class name for information tests, and irregular variable X to address the component of information tests. We can decipher $p(Y = C_k)$ as the earlier likelihood for the class C_k , which addresses the likelihood that the class name of an information test is C_k before we notice the information test. When we notice the component X of an information test, we can then utilize Bayes' hypothesis to process the relating back likelihood $p(Y|X)$. The amount $p(X|Y)$ can be communicated as how plausible the noticed information X is for various classes, which is known as the probability.

Note that the probability isn't a likelihood circulation over Y , and its vital concerning Y doesn't be guaranteed to rise to one. Considering this meaning of probability, we can express Bayes' hypothesis as back \propto probability \times earlier. Since we have presented the Bayes' hypothesis, in the following subsection, we will see the way Bayes' hypothesis is utilized in the Naive Bayes classifier.

3.1.2 Naive Bayes Classifier

Naive Bayes classifier is known to be the most straightforward Bayesian classifier, and it has turned into a significant probabilistic model and has been amazingly effective practically speaking in spite of areas of strength for its suspicion. Gullible Bayes has demonstrated successful in text arrangement, clinical analysis, and PC execution the board, among different applications. In the accompanying subsections, we will depict the model of Naive Bayes classifier, and greatest probability gauges as well as its applications.

Issue setting: Let us initially characterize the issue setting as follows: Suppose we have a bunch of preparing set $\{(x^{(i)}, y^{(i)})\}$ comprising of N models, each $x^{(i)}$ is a d -layered highlight vector, and each $y^{(i)}$ indicates the class mark for the model. We expect arbitrary factors Y and X with parts X_1, \dots, X_d comparing to the mark y and the component vector $x = [x_1, x_2, \dots, x_d]$. Note that the superscript is utilized to record preparing models for $i = 1, \dots, N$, and the addendum is utilized to allude to each element or irregular variable of a vector.

As a rule, Y is a discrete variable that falls into precisely one of K potential classes $\{C_k\}$ for $k \in \{1, \dots, K\}$, and the elements of X_1, \dots, X_d can be any discrete or ceaseless attributes.

Our task is to train a classifier that will output the posterior probability $p(Y|X)$ for possible values of Y . According to Bayes' theorem, $p(Y=C_k|X=x)$ can be represented as

$$\begin{aligned} p(Y = C_k|X = x) &= \frac{p(X = x|Y = C_k)p(Y = C_k)}{p(X = x)} \\ &= \frac{p(X_1 = x_1, X_2 = x_2, \dots, X_d = x_d|Y = C_k)p(Y = C_k)}{p(X_1 = x_1, X_2 = x_2, \dots, X_d = x_d)} \end{aligned} \quad (3.10)$$

One method for learning $p(Y|X)$ is to utilize the preparation information to appraise $p(X|Y)$ and $p(Y)$. We can then utilize these assessments, along with Bayes' hypothesis, to decide $p(Y|X = x(i))$ for any new example $x(i)$.

Learning accurate Bayesian classifiers is normally immovable. Taking into account the case that Y is boolean and X is a vector of d boolean elements, we want to gauge roughly 2^d boundaries $p(X_1 = x_1, X_2 = x_2, \dots, X_d = x_d | Y = C_k)$. That's what the explanation is, for a specific worth C_k , there are 2^d potential upsides of x , which need to process $2^d - 1$ free boundaries. Given two potential qualities for Y , we want to gauge a sum of $2(2^d - 1)$ such boundaries. In addition, to get dependable appraisals of every one of these boundaries, we should notice every one of these unmistakable occasions on various occasions, which is plainly unreasonable in most viable characterization areas. For instance, in the event that X is a vector with 20 Boolean elements, we should gauge more than 1 million boundaries.

To deal with the obstinate example intricacy for learning the Bayesian classifier, the Naive Bayes classifier decreases this intricacy by making a restrictive freedom presumption that the highlights X_1, \dots, X_d are restrictively free of each other, given Y . For the past case, this contingent freedom presumption serves to decisively decrease the quantity of boundaries to be assessed for displaying $p(X|Y)$ from the first $2(2^d - 1)$ to simply 2^d . Think about the probability $p(X = x | Y = C_k)$ of Equation (3.10), we have

$$\begin{aligned}
& p(X_1 = x_1, X_2 = x_2, \dots, X_d = x_d | Y = C_k) \\
&= \prod_{j=1}^d p(X_j = x_j | X_1 = x_1, X_2 = x_2, \dots, X_{j-1} = x_{j-1}, Y = C_k) \\
&= \prod_{j=1}^d p(X_j = x_j | Y = C_k). \tag{3.11}
\end{aligned}$$

From the chain guideline, overall property of probabilities of second equation and the third line follows straightforwardly from the above contingent freedom, that the incentive for the irregular variable X_j is autonomous of any remaining element values, X_{j-} for $j_- = j$, when molded on the character of the mark Y . This is the Naive Bayes supposition. It is a generally solid and exceptionally helpful suspicion. At the point when Y and X_j are boolean factors, we just need $2d$ boundaries to characterize $p(X_j|Y=C_k)$.

we can obtain the fundamental equation for the Naive Bayes classifier in substitution of Equation (3.11) in Equation (3.10),

$$p(Y = C_k | X_1 \dots X_d) = \frac{p(Y = C_k) \prod_j p(X_j | Y = C_k)}{\sum_i p(Y = y_i) \prod_j p(X_j | Y = y_i)}. \tag{3.12}$$

In the most probable value of Y , then we have the Naive Bayes classification rule for interested object:

$$Y \leftarrow \arg \max_{C_k} \frac{p(Y = C_k) \prod_j p(X_j | Y = C_k)}{\sum_i p(Y = y_i) \prod_i p(X_j | Y = y_i)}, \tag{3.13}$$

Above formulation can be simplified to the following but denominator does not depend on C_k .

$$Y \leftarrow \arg \max_{C_k} p(Y = C_k) \prod_j p(X_j | Y = C_k). \tag{3.14}$$

3.1.3 Maximum-Likelihood Estimates for Naive Bayes Models

In numerous functional applications, boundary assessment for Naive Bayes models utilizes the technique for most extreme probability gauges. To sum up, the Naive Bayes model has two kinds of boundaries that should be assessed. The first one is

$$\pi_k \equiv p(Y=C_k)$$

for any of the possible values C_k of Y . The parameter can be interpreted as the probability of seeing the label C_k , and we have the constraints $\pi_k \geq 0$ and $\sum_{k=1}^K \pi_k = 1$. Note there are K of these parameters, $(K-1)$ of which are independent.

For the d input features X_i , suppose each can take on J possible discrete values, and we use $X_i = x_{ij}$ to denote that. The second one is

$$\theta_{ijk} \equiv p(X_i = x_{ij} | Y = C_k)$$

for each info highlight X_i , every one of its potential qualities x_{ij} , and every one of the conceivable values C_k of Y . The incentive for θ_{ijk} can be deciphered as the likelihood of element X_i taking worth x_{ij} , adapted on the hidden mark being C_k . Note that they should fulfill $\sum_j \theta_{ijk} = 1$ for each sets of i, k qualities, and there will be dJK such boundaries, and note that main $d(J-1)K$ of these are free. These boundaries can be assessed utilizing greatest probability gauges in light of computing the general frequencies of the various occasions in the information. Most extreme probability gauges for θ_{ijk} given a bunch of preparing models are

$$\hat{\theta}_{ijk} = \hat{p}(X_i = x_{ij} | Y = C_k) = \frac{\text{count}(X_i = x_{ij} \wedge Y = C_k)}{\text{count}(Y = C_k)} \quad (3.15)$$

In this case, $\text{count}(x)$ return the number of examples in the training set. This satisfy the property of x , and $\text{count}(Y = C_k) = \sum_{n=1}^N \{Y^{(n)} = C_k\}$. This is a very natural estimate: The number of times label C_k is seen in conjunction with X_i taking value x_{ij} for sample count. The number of times for count C_k is seen in total.

To stay away from the case that the information doesn't end up containing any preparation models fulfilling the condition in the numerator, it is normal to adjust a smoothed gauge that really includes some of extra fantasized models similarly over the potential upsides of X_i . The smoothed gauge is given by

$$\hat{\theta}_{ijk} = \hat{p}(X_i = x_{ij} | Y = C_k) = \frac{\text{count}(X_i = x_{ij} \wedge Y = C_k) + l}{\text{count}(Y = C_k) + lJ}, \quad (3.16)$$

where J is the quantity of unmistakable qualities that X_i can take on, and l decides the strength of this smoothing. In the event that l is defined 1, and it is called Laplace smoothing.

For π_k , maximum likelihood estimates define as following term:

$$\hat{\pi}_k = \hat{p}(Y = C_k) = \frac{\text{count}(Y = C_k)}{N}, \quad (3.17)$$

where $N = \sum_{k=1}^K$, for number of example, $\text{count}(Y = C_k)$ is defined in the training set. Similarly, we can obtain a estimate of smoothed equation as follow:

$$\hat{\pi}_k = \hat{p}(Y = C_k) = \frac{\text{count}(Y = C_k) + l}{N + lK}, \quad (3.18)$$

where we define the number of distinct values as K and that Y can desire on, and the strength of the prior assumptions relative to the observed data is determined l .

3.2 Probabilistic and Naive Bayes Classifiers

Probabilistic classifiers are intended to utilize an implied combination model for age of the hidden reports. This blend model commonly expects that each class is a part of the combination. Every combination part is basically a generative model, which gives the likelihood of inspecting a specific term for that part or class. Therefore, this sort of classifiers is much of the time likewise called generative classifiers. The gullible Bayes classifier is maybe the least complex and furthermore the most generally utilized generative classifier. It demonstrates the circulation of the records in each class utilizing a probabilistic model with freedom suppositions about the conveyances of various terms. Two classes of models are generally utilized for gullible Bayes characterization. The two models basically register the back likelihood of a class, in view of the dispersion of the words in the report. These models disregard the genuine place of the words in the record, and work with the "pack of words" presumption. The significant contrast between these two models is the presumption with regards to taking (or not taking) word frequencies into account, and the relating approach for examining the likelihood space:

- **Multivariate Bernoulli Model:** In this model, we utilize the presence or nonappearance of words in a message record as elements to address a report. Subsequently, the frequencies of the words are not utilized for the displaying a report, and the word highlights in the text are thought to be paired, with the two qualities showing presence or nonappearance of a word in text. Since the highlights to be demonstrated are twofold, the model for reports in each class is a multivariate Bernoulli model.

- **Multinomial Model:** In this model, we catch the frequencies of terms in a report by addressing a record with a pack of words. The reports in each class can then be displayed as tests drawn from a multinomial word circulation. Thus, the restrictive likelihood of a report given a class is basically a result of the likelihood of each noticed word in the relating class.

Regardless of how we model the reports in each class (be it a multivariate Bernoulli model or a multinomial model), the part class models (i.e., generative models for records in each class) can be utilized related to the Bayes rule to process the back likelihood of the class for a given report, and the class with the most elevated back likelihood can then be relegated to the report.

There has been significant disarray in the writing on the distinctions between the multivariate Bernoulli model and the multinomial model. A decent piece of the distinctions between these two models might be found. The accompanying will portray these two models in more detail.

3.2.1 Bernoulli Multivariate Model

This class of methods regards a record as a bunch of unmistakable words with no recurrence data, in which a component (term) might be either present or missing. Allow us to expect that the vocabulary from which the terms are drawn are signified by $V = \{t_1 \dots t_n\}$. Let

us expect that the sack of-words (or text record) being referred to contains the terms $Q = \{t_{i1} \dots t_{im}\}$, and the class is drawn from $\{1 \dots k\}$. Then, we want to display the back likelihood that the report (which is thought to be produced from the term circulations of one of the classes) has a place with class I, considering that it contains the terms $Q = \{t_{i1} \dots t_{im}\}$. The most ideal way to comprehend the Bayes technique is by grasping it as an examining/generative interaction from the basic combination model of classes. The Bayes likelihood of class I can be displayed by examining a bunch of terms T from the term dissemination of the classes:

Assuming we examined a term set T of any size from the term conveyance of one of the haphazardly picked classes, and the ultimate result is the set Q , then, at that point, what is the back likelihood that we had initially picked class I for inspecting? The deduced likelihood of picking class I is equivalent to its fragmentary presence in the assortment.

We indicate the class of the examined set T by CT and the relating back likelihood by $P(CT = i|T = Q)$. This is basically the thing we are attempting to find. It is critical to take note of that since we don't permit substitution, we are basically picking a subset of terms from V without any frequencies connected to the picked terms. Consequently, the set Q may not contain copy components. Under the gullible Bayes presumption of freedom between terms, this is basically comparable to either choosing or not choosing each term with a likelihood that relies on the fundamental term circulation. Moreover, it is likewise vital to take note of that this model has no limitation on the number

of terms picked. As we will see later, these presumptions are the critical contrasts with the multinomial Bayes model. The Bayes approach groups a given set Q in light of the back likelihood that Q is an example from the information dispersion of class I , i.e., $P(CT = i|T = Q)$, and it expects us to register the accompanying two probabilities to accomplish this:

What is the earlier likelihood that a set T is an example from the term circulation of class I ? This likelihood is meant by $P(CT = I)$.

If we examined a set T of any size from the term dissemination of class I , then what is the likelihood that our example is the set Q ? This likelihood is signified by $P(T = Q|CT = I)$.

We will presently give a more numerical portrayal of Bayes demonstrating. At the end of the day, we wish to show $P(CT = i|Q \text{ is tested})$. We can utilize the Bayes rule to compose this restrictive likelihood in a manner that can be assessed all the more effectively from the basic corpus. As such, we can streamline as follows:

$$\begin{aligned}
 P(CT = i|T = Q) &= \frac{P(CT = i) \cdot P(T = Q|CT = i)}{P(T = Q)} \\
 &= \frac{P(CT = i) \cdot \prod_{t_j \in Q} P(t_j \in T|CT = i) \cdot \prod_{t_j \notin Q} (1 - P(t_j \in T|CT = i))}{P(T = Q)}
 \end{aligned}$$

The last state of the above succession utilizes the gullible freedom supposition, since we are accepting that the probabilities of event of the various terms are autonomous of each other. This is essentially vital, to change the likelihood conditions to a structure that can be assessed from the basic information.

The class relegated to Q is the one with the most noteworthy back likelihood given Q. It is not difficult to see that this choice isn't impacted by the denominator, which is the minor likelihood of noticing Q. That is, we will allot the accompanying class to Q:

$$\begin{aligned}\hat{i} &= \arg \max_i P(C^T = i | T = Q) \\ &= \arg \max_i P(C^T = i) \cdot \\ &\quad \prod_{t_j \in Q} P(t_j \in T | C^T = i) \cdot \prod_{t_j \notin Q} (1 - P(t_j \in T | C^T = i))\end{aligned}$$

It is critical to take note of that all terms in the right hand-side of the last condition can be assessed from the preparation corpus. The worth of $P(C^T = I)$ is assessed as the worldwide part of archives having a place with class I, the worth of $P(t_j \in T | C^T = I)$ is the negligible portion of reports in the i th class that contain term t_j . We note that the above are all most extreme probability evaluations of the relating probabilities. By and by, Laplacian smoothing is utilized, in which little qualities are added to the frequencies of terms to keep away from no probabilities of meagerly present terms. In many uses of the Bayes classifier, we just consideration about the personality of the class with the most elevated likelihood esteem, as opposed to the real likelihood esteem related with it, which is the reason we don't have to figure the normalizer $P(T = Q)$. Truth be told, on account of parallel classes, various improvements are conceivable in registering these Bayes "likelihood" values by utilizing the logarithm of the Bayes articulation, and eliminating various terms that don't influence the requesting of class probabilities.

In spite of the fact that for grouping, we don't have to process $P(T = Q)$, a few applications require the specific calculation of the back likelihood $P(C^T = i | T = Q)$. For instance, on account of administered oddity discovery (or uncommon class recognition), the specific back likelihood esteem $P(C^T = i | T = Q)$ is required to reasonably look at the likelihood esteem over various test cases, and rank them for their irregular nature. In such

cases, we would have to figure $P(T = Q)$. One method for accomplishing this is essentially to take a total over all the classes:

$$P(T = Q) = \sum_i P(T = Q | C^T = i) P(C^T = i)$$

This depends on the restrictive freedom of elements for each class. Since the boundary values are assessed for each class independently, we might deal with the issue of information meager condition. An elective approach to registering it, which might ease the information inadequacy issue, is to additionally make the presumption of (worldwide) freedom of terms, and process it as:

$$P(T = Q) = \prod_{j \in Q} P(t_j \in T) \cdot \prod_{t_j \notin Q} (1 - P(t_j \in T))$$

where the term probabilities depend on worldwide term conveyances in every one of the classes. A characteristic inquiry emerges, regarding whether it is feasible to plan a Bayes classifier that doesn't utilize the guileless suspicion, and models the conditions between the terms during the order interaction. Techniques that sum up the credulous Bayes classifier by not utilizing the freedom suspicion don't function admirably due to the higher computational expenses and the powerlessness to gauge the boundaries precisely and heartily within the sight of restricted information. On the one limit, a suspicion of complete reliance brings about a Bayesian organization model that ends up being computationally pricey. Then again, it has been shown that permitting restricted degrees of reliance can give great tradeoffs among precision and computational expenses.

While the freedom supposition that is a down to earth guess, it has been showing that the methodology has some hypothetical legitimacy. To be sure, broad exploratory tests have would in general show that the guileless classifier functions admirably practically speaking.

The Bayes technique gives a characteristic method for integrating such extra data into the grouping system, by making new elements for every one of these qualities. The standard Bayes method is then utilized related to this expanded portrayal for order. The Bayes method has likewise been utilized related to the fuse of different sorts of area information, for example, the consolidation of hyperlink data into the grouping system.

The Bayes strategy is likewise fit to progressive order, while the preparation information is organized in a scientific classification of subjects. For instance, the Open

Directory Project (ODP), Yahoo! Scientific classification, and an assortment of information locales have immense assortments of reports that are organized into various leveled gatherings. The progressive design of the points can be taken advantage of to perform more successful arrangement, since it has been seen that setting delicate component determination can give more helpful grouping results. In progressive grouping, a Bayes classifier is worked at every hub, which then, at that point, gives us the following branch to follow for order purposes. Two such strategies are proposed, in which hub explicit elements are utilized for the arrangement cycle. Obviously, many less highlights are expected at a specific hub in the order, in light of the fact that the elements that are picked are pertinent to that branch.

3.2.2 Multinomial Distribution

This class of methods regards a record as a bunch of words with frequencies connected to each word. Subsequently, the arrangement of words is permitted to have copy components. As in the past case, we accept that the arrangement of words in archive is meant by Q , drawn from the jargon set V . The set Q contains the unmistakable terms $\{t_{i1} \dots t_{im}\}$ with related frequencies $F = \{F_{i1} \dots F_{im}\}$. We indicate the terms and their frequencies by $[Q, F]$. The complete number of terms in the archive (or record length) is signified by $L = \sum_{j=1}^m F_{ij}$. Then, we want to show the back likelihood that the record T has a place with class I , considering that it contains the terms in Q with the related frequencies F . The Bayes likelihood of class I can be demonstrated by utilizing the accompanying examining process:

If we sampled L terms sequentially from the term distribution of one of the randomly chosen classes (allowing repetitions) to create the term set T , and the final outcome for sampled set T is the set Q with the corresponding frequencies F , then what is the posterior probability that we had originally picked class i for sampling? The a-priori probability of picking class i is equal to its fractional presence in the collection.

The previously mentioned likelihood is indicated by $P(CT = i | T = [Q, F])$. A supposition that is usually utilized in these models is that the length of the report is free of the class name. While it is effectively conceivable to sum up the strategy, with the goal that

the report length is utilized as an earlier, freedom is normally expected for effortlessness. As in the past case, we want to appraise two qualities to register the Bayes back.

1. What is the prior probability that a set T is a sample from the term distribution of class i ? This probability is denoted by $P(C^T = i)$.
2. If we sampled L terms *from the term distribution of class i* (with repetitions), then what is the probability that our sampled set T is the set Q with associated frequencies F ? This probability is denoted by $P(T = [Q, F] | C^T = i)$.

$$P(C^T = i | T = [Q, F]) = \frac{P(C^T = i) \cdot P(T = [Q, F] | C^T = i)}{P(T = [Q, F])} \propto P(C^T = i) \cdot P(T = [Q, F] | C^T = i). \quad (3.19)$$

As in the past case, it isn't important to register the denominator, $P(T = [Q, F])$, to conclude the class name for Q . The worth of the likelihood $P(C^T = I)$ can be assessed as the negligible part of archives having a place with class I . The calculation of $P([Q, F] | C^T = I)$ is more confounded. At the point when we consider the consecutive request of the L various examples, the quantity of potential ways of testing the various terms in order to bring about the result $[Q, F]$ is given by $L! \prod_{i=1}^m F_i!$. The likelihood of every one of these groupings is given by $\prod_{t_j \in Q} P(t_j \in T | C^T = i)^{F_j}$, by utilizing the gullible freedom supposition. In this way, we have:

$$P(T = [Q, F] | C^T = i) = \frac{L!}{\prod_{i=1}^m F_i!} \cdot \prod_{t_j \in Q} P(t_j \in T | C^T = i)^{F_j}. \quad (3.20)$$

Substitute Condition 3.20 in Equation 3.19 to get the class with the most noteworthy Bayes back likelihood, where the class priors are figured as in the past case, and the probabilities $P(t_j \in T | C^T = I)$ can likewise be effortlessly assessed as already with Laplacian smoothing. Note that to pick the class with the most elevated back likelihood, we don't actually need to register $L! \prod_{i=1}^m F_i!$, as it is a consistent not relying upon the class name (i.e., the equivalent for every one of the classes). We additionally note that the probabilities of class nonattendance are absent in the above conditions in view of the manner by which the testing is performed.

Various varieties of the multinomial model have been proposed. In the work, it is demonstrated the way that a class order can be utilized to work on the gauge of multinomial

boundaries in the gullible Bayes classifier to further develop grouping precision fundamentally. The key thought is to apply shrinkage methods to smooth the boundaries for information scanty kid classifications with their normal parent hubs. Subsequently, the preparation information of related classes are basically "shared" with one another in a weighted way, which works on the vigor and precision of boundary assessment when there are deficient preparation information for every individual youngster classification. The work has played out a broad examination between the

Bernoulli and the multinomial models on various corpora, and the accompanying ends were introduced:

- The multi-variate Bernoulli model can in some cases perform better compared to the multinomial model at little jargon sizes.
- The multinomial model outflanks the multi-variate Bernoulli model for huge jargon sizes, and quite often beats the multi-variate Bernoulli when jargon size is decided ideally for both. On the normal a 27% decrease in mistake.

In advance of referenced results strongly imply that the two models might have various qualities, and may hence be valuable in various situations.

3.2.3. Multinomial Naïve Bayes

The Multinomial Naive Bayes algorithm is mostly used in Natural Language Processing (NLP) and it is a probabilistic method. In many texts classification, Multinomial Naive Bayes (MNB) is widely used from a Bayesian approach. To solve text classification problems, MNB is an extension of the Naïve Bayes considered. Naïve Bayes classifiers are based on Bayes theorem. Bayes theorem calculates probability $P(A|B)$ where B is spam or ham and B is $B_1, B_2, B_3 \dots B_n$ from an upload email. Naïve Bayes algorithm calculates and total probability as follow:

$$P(A|B) = \frac{P(B|A).P(A)}{P(B|A)\sum_s P(A)} \quad (3.21)$$

B is a multiset of words that occur in a document and A means that the document is spam. So, identifying an email as spam or ham is binary, we know that our class can only ever be 0 or 1, A or C. In this case, the following equation (5) describe the Multinomial Bayes equation:

$$P(B \setminus A) = \frac{\sum_B f_B}{\prod_B f_B} \prod_B P(B \setminus A)^{f_B} \quad (3.22)$$

f_B is the number of times that the word B takes place in the multiset B

\prod_B is the product for each word in B.

$P(B \setminus A)$ is the probability that a given word occurs in a spam email.

α is the smoothing parameter.

We consider Laplace smoothing following equation (3.23).

$$P(B \setminus A) = \frac{N_{w,s} + M_{w,s}}{N_s + \sum_w M_{w,s}} \quad (6)$$

$M_{w,s}$ is smoothing parameter.

$N_{w,s}$ is the number of occurrences of a word B in a spam email.

N_s is the total number of words that have been found in the spam emails up to this point by the filter.

Spam classification is described using Multinomial Naïve Bayes as following algorithm.

Algorithm

Br s = Training subset of "s" and

Hr s = Testing subset

Br= Training and Hr= Testing

$P(t_k | g)$ = conditional probability

Initial= input variables;

t =number of documents;

s = datapoints;

y =desired inputs;

for $i = 0, i < Br\ s; i=i+1$ do

 if $(i,y) = Spam$ then

$i = Spam;$

 else

$i = Ham;$

for testing

do

```

for j in sd do
    s_test and y_test = testing size;
    s_train and y_train = training size;
    for i = 0; i < H r s;
        i ++
Calculate _ P (tk |g);
Calculate the Accuracy;
return tk;

```

This number will grow as more emails are added to the filter that are classified as spam and that contain new words. In our case of email spam detection, it is most likely that the calculated probability would be compared to a certain threshold that is given by the user to classify these mail as either spam or ham. The class of the email would be decided by the maximum of the two values. Machine learning based feature extraction is used in computing term weight using Multinomial Naïve Bayes algorithm.

3.3 Sentiment Analysis in Social Network

Sentiment analysis is commonly used with Natural Language Processing (NLP), and it is also known as “conversation mining”. In recent times, research on sentiment analysis has increased considerably in business and social applications. Typically, social media stream analysis is limited to simple sentiment analysis and count-based indicators. For sentiment analysis, the most basic approaches include Lexicon-based (using a dictionary or thesaurus), Machine Learning-based (using some ML algorithm), and a hybrid of these two.

Sentiment analysis is a computational technique that describes a customer’s sentiment or view from online product reviews, and it is a supervised learning approach. For the past years, sentiment analysis has mainly relied on a body of disappeared texts extracted from the web, social media, or internal company resources. Sentiment analysis is a discipline that [aims to extract qualitative characteristics from user’s text data](#), such as sentiment, opinions, thoughts, and behavioral intent using

natural language processing methods.

The challenge of sentiment analysis is more effectively determining a text, that is expressed with regard to an object, product, event, and brand selected by the opinion. The opinion can thus have a positive, negative or neutral polarity, and it can be studied at the different document levels, sentence levels, entity levels, and aspect levels. In many fields of application, detection and classification of opinions through machine learning have many methods. Sentiment analysis has now spread from computer science to management and social sciences like marketing, finance, political science, communications, health science, and even history, because of its importance for companies and society as a whole.

As a result, Natural Language Processing for emotion-based sentiment analysis is especially beneficial. Organizations may analyze consumer emotions and respond appropriately using NLP for speech analysis paired with a sophisticated social media monitoring strategy to enhance customer experience, swiftly handle customer complaints, and shift their market position. Sentiment Analysis can use turn all of this unwanted text into desired data using NLP and open-source technologies. Spam detection is one of the common NLP problems that experts consider mostly solved.

3.1.1. Sentiment analysis and Natural Language Processing

Natural language processing (NLP) is the ability of a computer program to understand human language. It is spoken and written referred to as natural language. NLP is a component of artificial intelligence ([AI](#)), and sentiment analysis is a subset of Natural Language Processing (NLP). Traditional studies on sentiment analysis are as follows:

- aim to [detect polarity in a given text by classifying it as positive, negative, or neutral.](#)
- to [recognize multiple differentiated affective manifestations in text.](#)
- o obtain data for sentiment analysis is difficult so it requires labels for a subset of the data, with which to train the model.

3.1.2 Sentiment Analysis Models

The sentiment text analysis or classification is more exactly positive and negative this relies on using a dataset.

3.1.3 Bag-of-words (BOW)

In Natural Language Processing, Bag of words is a technique of text modelling. It is a method of feature extraction with text data, a simple and flexible way of extracting features from documents. A bag of words is a representation of text that describes the occurrence of words within a document.

3.1.4 Lexicon Based Approach

This thesis paper is going to describe the most basic approach for sentiment analysis, i.e., Lexicon-based Sentiment Analysis. A lexicon application has one of the two main approaches to sentiment analysis and it involves calculating the sentiment from the semantic orientation of words and phrases annotation. Lexicon-based approaches a piece of the text message is represented as a bag of words. Following this representation of the message, sentiment values from the dictionary are assigned to all positive and negative words or phrases within the message.

3.1.5 Part of Speech (POS) Tagging

A word can be categorized into one or more of a set of lexical or part-of-speech classes such as Nouns, Verbs, Adjectives and Articles, to name a few. A POS tag is a symbol representing such a lexical category – NN (Noun), VB (Verb), JJ (Adjective), AT (Article). One of the oldest and most commonly used tag sets is the Brown Corpus tag set.

Given a sentence and a set of POS tags, a mutual language processing task is to automatically specify POS tags to each word in the sentences. For example, given the sentence "I plane to give on this month end", the output of a POS tagger would be I /PRP plane/VB to/TO give/VB on/IN this/ DD month/NN end/NN. Tagging text with part-of-speech turns out to be much more beneficial for more complicated NLP tasks such as parsing and machine translation.

3.4. SentiWordNet3.0

This thesis paper describes SENTIWORDNET 3.0, a lexical resource explicitly devised for supporting sentiment classification. SentiWordNet is one of the lexicons approaches that assigns to each synset of WordNet, and the three sentiment numerical scores are positivity, negativity, and objectivity. The SentiWordNet (SWN) plays an important role in extracting opinions from texts. It is a publicly available sentiment measuring tool used in sentiment classification and opinion mining applications, adding information on the sentiment-related properties of the terms in the text. Following figure 1 is depicted the preparation of steps for generating Arabic SentiWordNet.

SentiWordNet is used to find the sentiment of words by using the following algorithm.

Algorithm

Input: Noun, Adjective, Adverbs from a sentence in the dataset of the sentence.

Output: positive, negative, and neural

Variable: A= List of words files, B=file in A, C=file in SentiWordNet, D=word in B, E= word in C

for B in A

 for D in B

 for C in SentiWordNet

 for E in C

 if D==E then,

 write D to output file with the same name as B

 flag=1

```
break
    for flag==1 then
break
```

Following figure 1 depicted the preparation step for generating Arabic SentiWordNet.

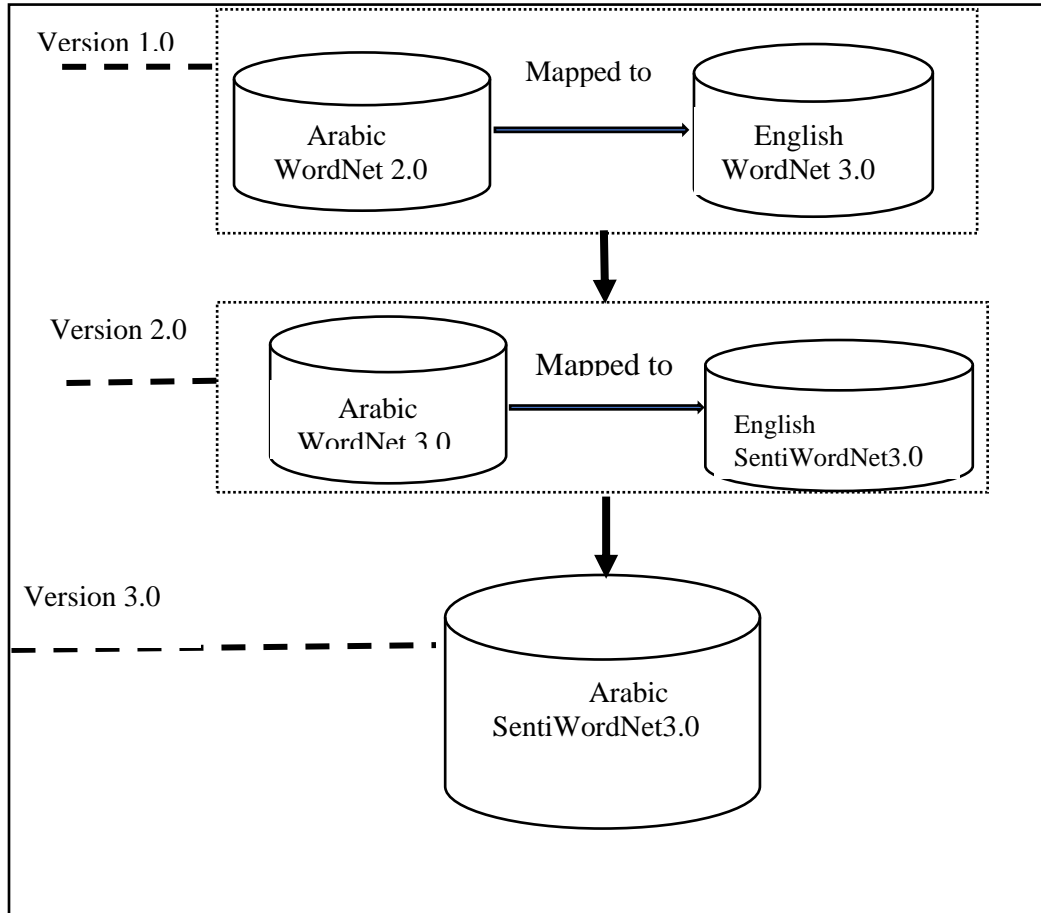


Figure 3.1 Preparation step for sent WordNet

CHAPTER 4

SYSTEM DESIGN AND IMPLEMENTATION

Sentiment Analysis also known as “opinion mining” is a field of study that aims at extracting opinions and sentiments from natural language text using computational methods. It is widely applied to email messages, reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service. Sentiment analysis refers to the inference of people’s position and attitude in their written or spoken words. Before the coming of the term, the field was studied under names such as point of views and opinion mining.

3.4. Sentiment Classification Using Machine Learning Approach

Nowadays, many new researchers have been influenced by online & public forum site data. To derive precious information of raw data sources used many machine learning algorithms. Machine learning is an application of AI that enables systems to learn and improve from experience without being explicitly programmed which aims to develop an algorithm in order to optimize the performance of the system by using example data. For AI technologies, machine learning gives clear benefits. There are many machine learning methods to choose from including:

- ❖ supervised learning
- ❖ unsupervised learning
- ❖ semi-supervised learning

Supervised machine learning algorithms a subcategory of machine learning and artificial intelligence, and use what has been learned in the past to new data using labeled examples to predict future events. Its algorithm gives a decision function to predict output values by analyzing a known training dataset. The system can provide desirable any new input after sufficient training. Supervised machine learning uses training data sets to achieve desired results and Regression, Logistic Regression, Classification, Naïve Bayes Classifiers, Decision Trees, and Support Vector Machines are types of supervised machine

learning. The advantages of supervised learning allow you to collect data or produce a data output from the previous experience, help you to optimize performance criteria using experience, and help you to solve types of the various real problem.

Unsupervised Learning is a machine learning technique in which the users do not need to supervise the model. Unsupervised Learning Algorithms allow users to perform more complex processing tasks compared to supervised learning and which algorithm include clustering, anomaly detection, neural networks, etc. When the information used to train is neither classified nor labeled, unsupervised machine learning algorithms are used. How systems can infer a function to describe a hidden structure from unlabeled data can be studied from unsupervised learning. The most three popular unsupervised learning tasks are dimensionality reduction, anomaly detection, and clustering. Other unsupervised learning tasks are density estimation and rule learning.

Semi-supervised learning is a type of machine learning that is between supervised and unsupervised learning and involves a small portion of labeled examples and a large number of unlabeled examples. The following facts explained the whole work of semi-supervised learning. Firstly, it trains the model with less amount of training data similar to the supervised learning models. The training continues until the model gives accurate results. in the next step, algorithms use the unlabeled dataset with pseudo labels but the result may not be accurate. Labeled training data and pseudo labels data are linked together currently. Then, the input data in labeled training data and unlabeled training data are also linked. Finally, again train the model with the new combined input as done the first step. It will reduce errors and improve the accuracy of the model.

Machine learning techniques are generally used for binary classification and predictions of sentiments as either positive or negative. The solution that machine learning provides for sentiment analysis involves two main steps. The first step is to” learn” the model from the training data and the second step is to classify the unseen data.

Three different levels of sentiment classification are:

- Document level
- Sentence level
- Feature or Aspect level

Domain areas for sentiment analysis are:

- Data service email messages
- Business Intelligent service enhancement
- Finance and Stock Monitoring

Spam Mail Detection and Sentiment Score Classification

In statistics, Naive Bayes classifiers are a family of simple “probabilistic classifiers” based on applying Bayes’ theorem with strong independence assumptions between the features:

- Multinomial Naive Bayes
- Multivariate Bernoulli Naive Bayes
- Gaussian Naive Bayes

4.1 Feature Extraction

Text feature extraction plays an essential role in text classification and the text features usually use a keyword set. It means that on the basis of a group of predefined keywords, Feature extraction used to compute weights of the words in the text by certain methods and then form a digital vector, which is the feature vector of the text. Existing text feature extraction method for clustering approach which is particularly suitable for large-scale text feature extraction. Text feature extraction mainly has Bag of Words, Word to Vector, N-grams, TF-IDF (Term Frequency-Inverse Document Frequency), information gain (IG), and mutual information (MI) method, etc.

After completing the text preprocessing the next step is weighting the term using TF-IDF which is designed to reflect how important a word is to a document or a set of documents. Term Frequency (TF) is term weighting based on the words frequency that appear in a document. The higher the TF value of a word in a document, the higher the effect of the term on the document. Inverse Document Frequency (IDF) is weighting method based on number of words that appear through all the documents.

The important value (or weighting) of a word increases proportionally to the number of times it appears in the document (Term Frequency). The weighting is offset by the number of documents in the set containing the word (Inverse Document Frequency).

Some words appear more generally across documents and hence are less unique identifiers. So, their weighting is lessened.

TF-IDF: Term Weighting Schemes (Term Frequency-Inverse Document Frequency)

The formulation of this method is as follows:

$$W(d, t) = tf(t, d) * \log(N / n_t) \tag{4.1}$$

- Where: $w(t,d)$ = term weight in document d
- $tf(t,d)$ = term frequency in document
- N = the total number of documents
- n_t = number of documents that have term t

TF-IDF is one of the simplest and strongest weighting schemes to evaluate how relevant a word is to a document in a collection of documents. TF-IDF and its algorithm versions are default choice in text categorization because of its simple formulation and good performance on a number of various data sets.

Table 4.1 Training data of SMS spam email message

No. Doc	Messages	Class
1	Call Free PHONE 0800 542 0578 now!	spam
2	Win a £ 1000 cash prize or prize worth £5000	spam
3	08714712388 BETWEEN 10AM-7PM Cost 10pm	spam
4	Ï Predict wait time ï...’ll finish buying?	ham
5	Both:) I shoot big loads so get ready!	ham
6	You have won? 1000 cash or a ? 2,000 prize ! call 09050000327	?
7	Î’ll finished buying? So get ready!	?

Table 4.2 Tokenization, remove stopword and stemming of pre-processing training data

No. Doc	Messages	Class
1	Call, free phone	spam
2	Win, cash, prize, prize, worth	spam
3	cost	spam
4	Predict, wait, time, finish, buying	ham
5	Shoot, big, loads, get, ready	ham

After extraction, this system calculates the weight of each feature words by using TF-IDF (Term Frequency-Inverse Document Frequency) methods. Term frequency (TF) result for training data in spam document is shown in Table (4.2).

After extraction that system calculate the weight of each feature words by using TF-IDF methods. Term frequency (TF) result for training data in ham document is shown in Table (4.3).

Table 4.3 Term Frequency (TF) of each keyword

Feature Word	Count of Feature Words	Maximum count of the feature words in the Ham document	Term Frequency of Feature word
call	1	2	0.5
Free	1	2	0.5
phone	1	2	0.5
win	1	2	0.5

cash	1	2	0.5
prize	2	2	1
worth	1	2	0.5
cost	1	2	0.5
predict	1	2	0.5
wait	1	2	0.5
finish	1	2	0.5
buying	1	2	0.5
shoot	1	2	0.5
big	1	2	0.5
load	1	2	0.5
get	1	2	0.5
ready	1	2	0.5

According to the TF_IDF method, the Inverse Document Frequency (IDF) result for all document is shown in Table (4.4).

Table 4.4 Inverse Document Frequency (IDF) of each keyword

Feature Word	Count of document in Feature Word					Count of Feature Word	Inverse document Frequency of feature
	Doc1	Doc2	Doc3	Doc4	Doc5		
call	1	0	0	0	0	1	$\text{Log}_5/1 = 0.698$
Free	1	0	0	0	0	1	$\text{Log}_5/1 = 0.698$
phone	1	0	0	0	0	1	$\text{Log}_5/1 = 0.698$

win	0	1	0	0	0	1	$\text{Log}5/1 = 0.698$
cash	0	1	0	0	0	1	$\text{Log}5/1 = 0.698$
prize	0	2	0	0	0	2	$\text{Log}5/ = 0.698$
worth	0	1	0	0	0	1	$\text{Log}5/1 = 0.698$
cost	0	0	1	0	0	1	$\text{Log}5/1 = 0.698$
predict	0	0	0	1	0	1	$\text{Log}5/1 = 0.698$
time	0	0	0	1	0	1	$\text{Log}5/1 = 0.698$
wait	0	0	0	1	0	1	$\text{Log}5/1 = 0.698$
finished	0	0	0	1	0	1	$\text{Log}5/1 = 0.698$
buying	0	0	0	1	0	1	$\text{Log}5/1 = 0.698$
shoot	0	0	0	0	1	1	$\text{Log}5/1 = 0.698$
big	0	0	0	0	1	1	$\text{Log}5/1 = 0.698$
load	0	0	0	0	1	1	$\text{Log}5/1 = 0.698$
get	0	0	0	0	1	1	$\text{Log}5/1 = 0.698$
ready	0	0	0	0	1		$\text{Log}5/1 = 0.698$

By using term frequency (TF) and Inverse document frequency (IDF) results, the system calculates the weight of each feature words of in all messages document. Table (4.5) shows in the weight result for document.

Table 4.5 Weight of each keyword

Feature Word	Term Frequency of Each Feature word	Inverse Document of each feature word	Weight of each word
call	0.33	0.698	0.2303
Free	0.33	0.698	0.2303
phone	0.33	0.698	0.2303
win	0.33	0.698	0.2303
cash	0.33	0.698	0.2303
prize	0.66	0.698	0.4606
worth	0.33	0.698	0.2303
cost	0.33	0.698	0.2303
predict	0.33	0.698	0.2303
time	0.33	0.698	0.2303
wait	0.33	0.698	0.2303
finished	0.33	0.698	0.2303
buying	0.33	0.698	0.2303
shoot	0.33	0.698	0.2303
big	0.33	0.698	0.2303
load	0.33	0.698	0.2303
get	0.33	0.698	0.2303
ready	0.33	0.698	0.2303

Table 4.6 Weight of each keyword for each document

No. Doc	Feature Word	Count of feature word	Weight of feature word
Doc1	Call	1	0.2303
	free	1	0.2303
	phone	1	0.2303
Doc2	win	1	0.2303
	cash	1	0.2303
	Prize	2	0.4606
	worth	1	0.2303
Doc3	cost	1	0.2303
Doc4	predict	1	0.2303
	wait	1	0.2303
	time	1	0.2303
	finished	1	0.2303
	buying	1	0.2303
Doc5	shoot	1	0.2303
	big	1	0.2303
	load	1	0.2303
	get	1	0.2303
	ready	1	0.2303

Table 4.7 Total weight of keyword for each document

No.	Feature Words in Document	Feature Words count in Document	D1	D2	D3	D4	D5
1	call	1	0.2303				
2	Free	1	0.2303				
3	phone	1	0.2303				
4	win	1		0.2303			
5	cash	1		0.2303			
6	prize	2		0.4606			
7	worth	1		0.2303			
8	cost	1			0.2303		
9	predict	1				0.2303	
10	time	1				0.2303	
11	wait	1				0.2303	
12	finished	1				0.2303	
13	buying	1				0.2303	
14	shoot	1					0.2303
15	big	1					0.2303
16	load	1					0.2303
17	get	1					0.2303
18	ready	1					0.2303
Total			0.69	1.15	0.23	0.69	1.15

4.2 Probabilistic Naïve Bayes Classification

In statistics, Naive Bayes classifiers are a sub-category of simple “probabilistic classifiers” based on applying Bayes’ theorem with strong independence assumptions between the features.

Multinomial Naive Bayes: This is mostly used for document classification problem, i.e., whether a document belongs to the category of sports, politics, technology etc. The features/predictors used by the classifier are the frequency of the words present in the document. Feature vectors represent the frequencies with which certain events have been generated by a multinomial distribution. This is the event model typically used for document classification.

Multivariate Bernoulli Naive Bayes: This is similar to the multinomial naive bayes but the predictors are Boolean variables. For instance, if a word occurs in the text or not. This model is popular for document classification tasks, where binary term occurrence (i.e., a word occurs in a document or not) features are used rather than term frequencies (i.e., frequency of a word in the document).

Gaussian Naive Bayes, is continuous values associated with each feature are assumed to be distributed classifier according to a Gaussian distribution.

4.3 Multinomial Naïve Bayes

Multinomial naïve bayes is a popular method for document classification. It is used for discrete counts. It determines the occurrence probability of an event. Its task is considering the probability of an occurred event. Multinomial naïve bayes classifiers use multinomial distribution for each one of the features on data.

$$\begin{aligned} P(w_i | c) &= \frac{\text{count}(w_i, c)}{\sum_{w \in N} (\text{count}(w, c))} \\ &= \frac{\text{count}(w_i, c) + 1}{(\sum_{w \in N} (\text{count}(w, c)) + |N|)} \end{aligned} \quad 4.2$$

Where: W = word/ term / feature
 C = class/category
 W_i = no of occurrence of feature in all document
 N = Total number of unique features in all document

4.4 Overview of the System Flow Diagram

The overview of the proposed system consists of two main processes such as Multinomial Naïve Bayes Classification and Sentimental Score Classification. In the first system flow diagram, there are four steps such as load input email messages, text pre-processing, TF-IDF feature extraction and Multinomial Naïve Bayes Classification. In the first step, input email messages are imported as training data and testing data. Each email message used the text structure of document level. The text pre-processing step includes three sub-stages of tokenization, stopword removal and stemming. In feature extraction step, keywords are selected and then classification technique is applied on extracted features to classify them into its TF-IDF weight calculation score. In classification step, this system used Multinomial Naïve Bayes Classifier. Finally, testing email message is classified as a result of Spam class or Ham class. For the proposed Sentiment Level Analysis of Detecting Spam Email System, there would be the system flow diagram for Multinomial Naïve Bayes classification and the system flow diagram for Classifying Sentiment Score as shown in figure (4.1) and figure (4.2).

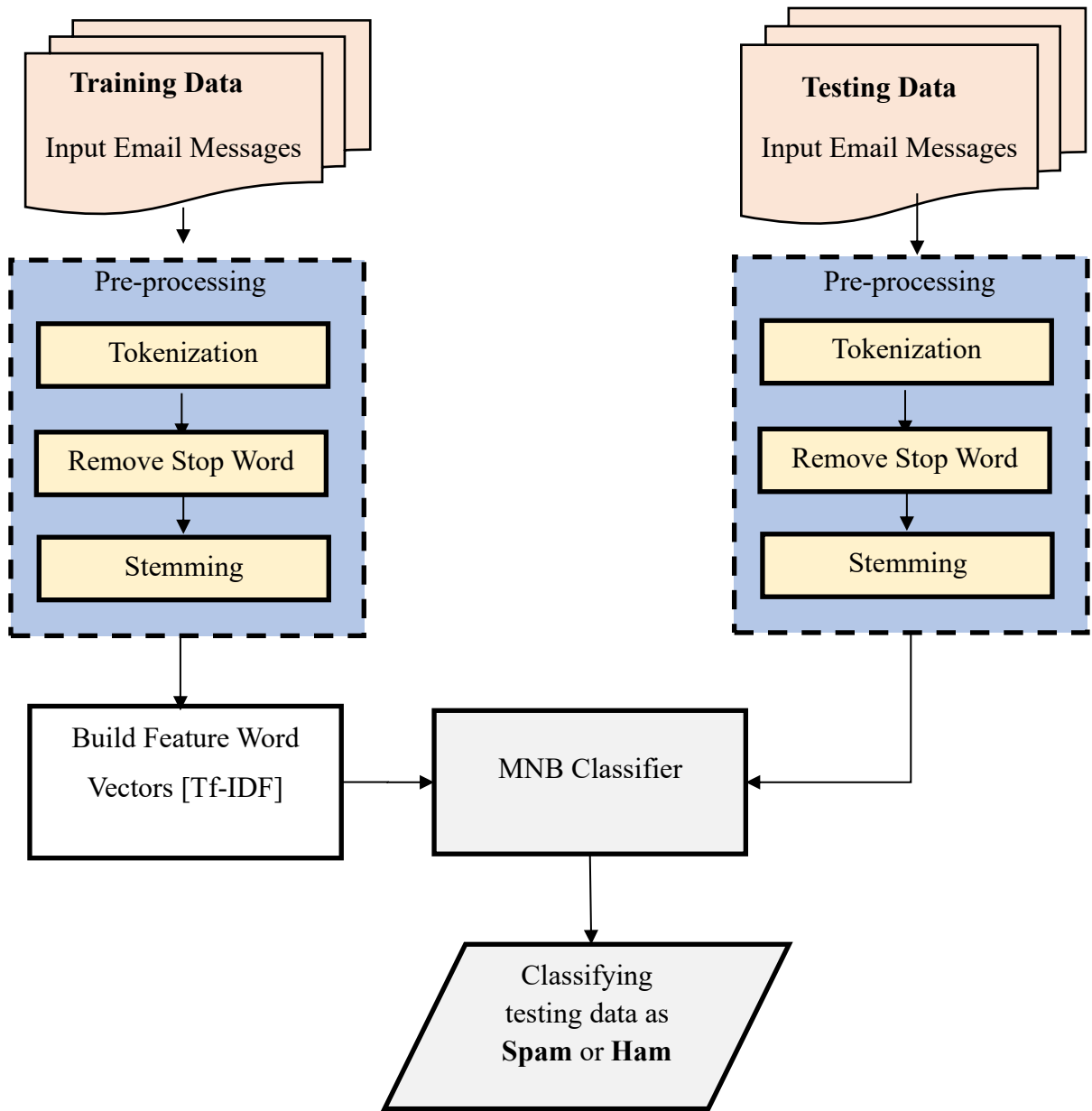


Figure 4.1 System Flow Diagram for Multinomial Naïve Bayes Classification

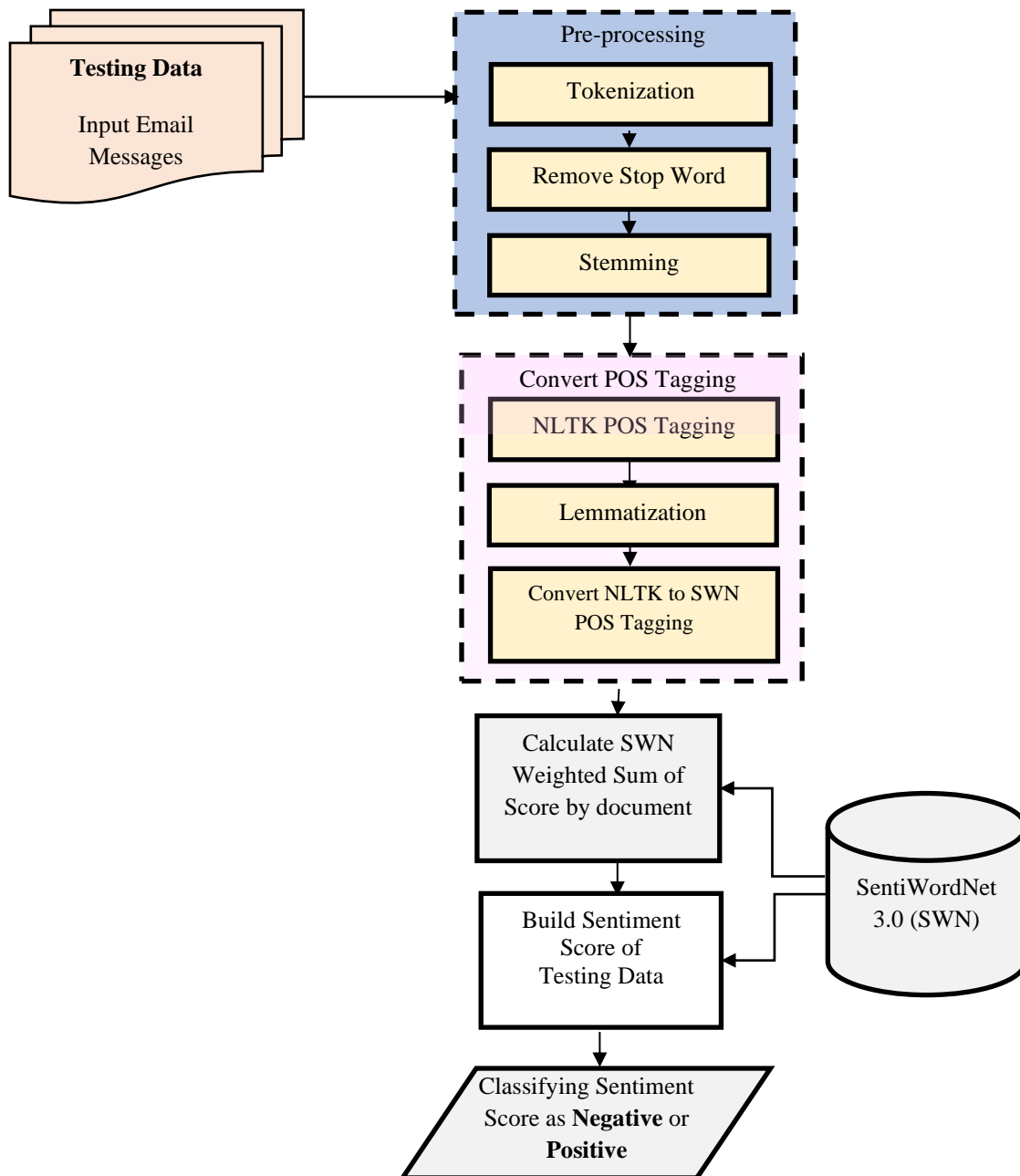


Figure 4.2. System Flow Diagram for Classifying Sentiment Score

4.5 System Implementation

The system consists of eight processes: “Import Training Data”, “Pre-processing on Training Data”, “Calculation of Naïve Bayes classifier Accuracy Result”, “Import Testing Data”, “Pre-processing on Testing Data”, “Naïve Bayes classification”, “Object Score for Testing Data” “Classifying Sentiment Score”. “Import Training Data” is used to

import the training data to the system. “Pre-processing Training Data” is used to make the pre-processing task (such as tokenization, removing stop-word and stemming) on the data of the system. “Import Testing Data” is supported to import the desire testing data to the system. “Pre-processing on Testing Data” task is same as the pre-processing of the training data. “Object Score” is used to process the sentiment analysis on the data source of SMS email messages by using SentiwordNet3.0 and Multinomial Naïve Bayes approach.

4.5.1 Description of Input Data Source

The SMS Email Message Collection is a set of SMS-tagged messages that have been collected from Kaggle and this site is one of the world’s largest community of data scientists and machine learning specialists. SMS Spam Mail Collection from www.kaggle.com and it contains one set of SMS messages in English of 5,574 messages, tagged according being ham (legitimate) or spam. The files contain one document as a one message per line. Each line is composed of two columns: v1 contains the label (ham or spam) and v2 contains the raw text. This corpus has been collected from free or free research sources at on internet.

A collection of 425 SMS spam messages was manually extracted from the Grumbletext Web site. This is a UK forum in which cell phone users make public claims about SMS spam messages, most of them without reporting the very spam message received. The identification of the text of spam messages in the claims is a very hard and time-consuming task, and it involved carefully scanning hundreds of web pages. The Grumbletext Web site is: [\[WebLink\]](#). A subset of 3,375 SMS randomly chosen ham messages of the NUS SMS Corpus (NSC), which is a dataset of about 10,000 legitimate messages collected for research at the Department of Computer Science at the National University of Singapore. The messages largely originate from Singaporeans and mostly from students attending the University. These messages were collected from volunteers who were made aware that their contributions were going to be made publicly available. The NUSSMS Corpus is available: [\[WebLink\]](#). A list of 450 SMS ham messages collected from Caroline Tag's Ph.D. This is available at [\[Web Link\]](#). Finally, we have incorporated

the SMS Spam Corpus v.0.1 Big. It has 1,002 SMS ham messages and 322 spam messages and it is publicly available at: [\[Web Link\]](#).

The following figure (4.3) is input data source of email message for training.

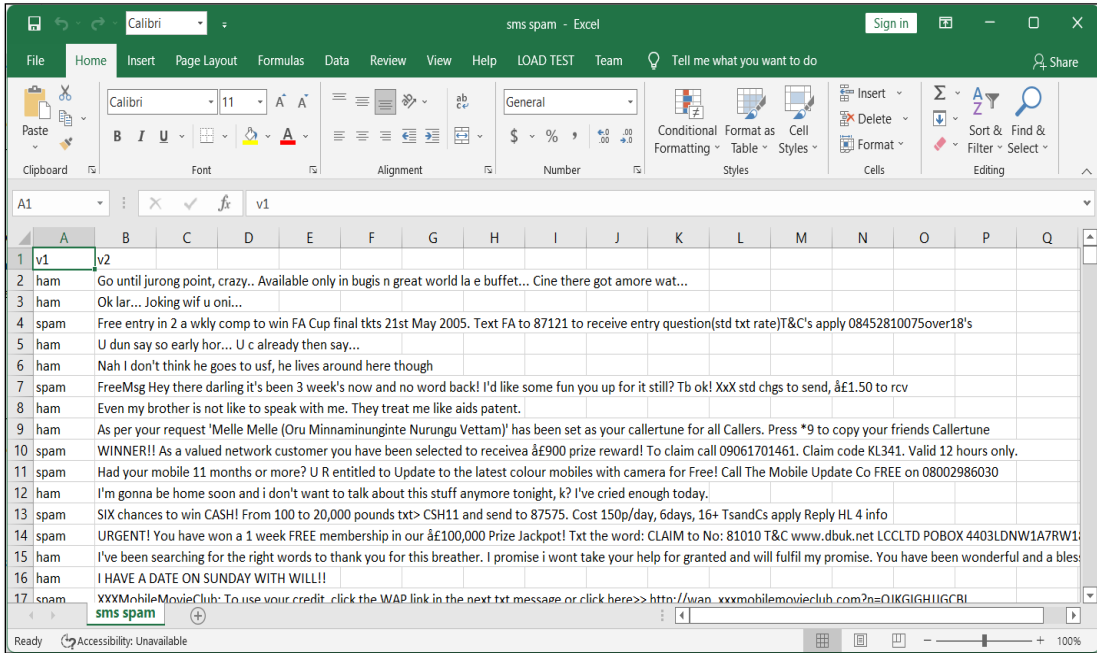


Figure 4.3 SMS email message data

This entry section describes the spam detection in sentiment analysis. To label for training data, firstly the user may import training data of SMS email messages.

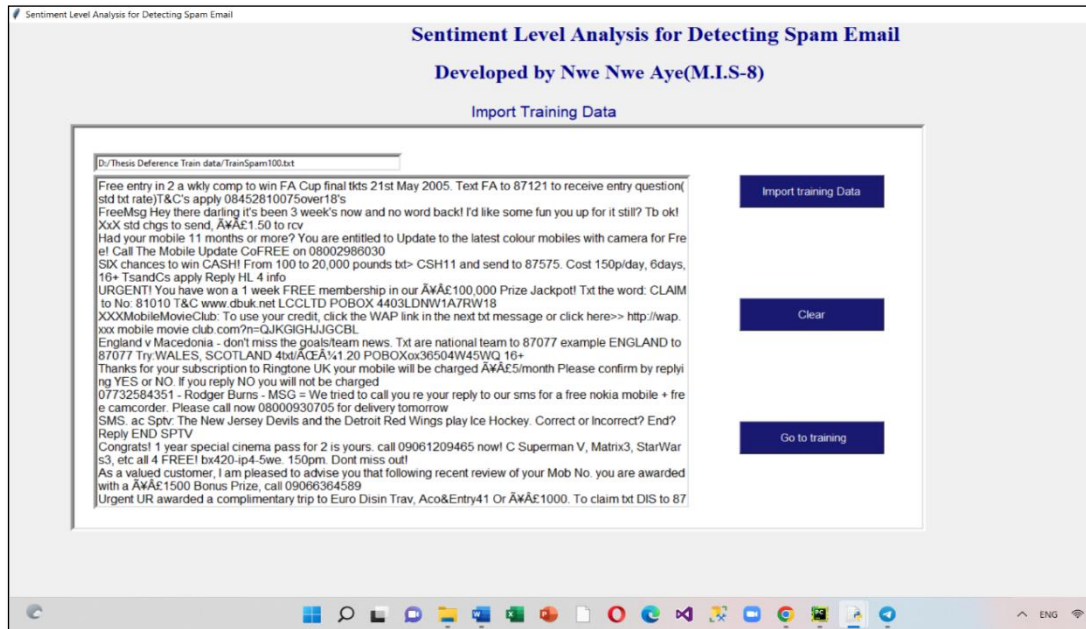


Figure 4.4 Import Training Data

Secondly, if the user clicks go to training, before tokenization and stopword removal as shown in figure (4.5), dialog box will appear for training data.

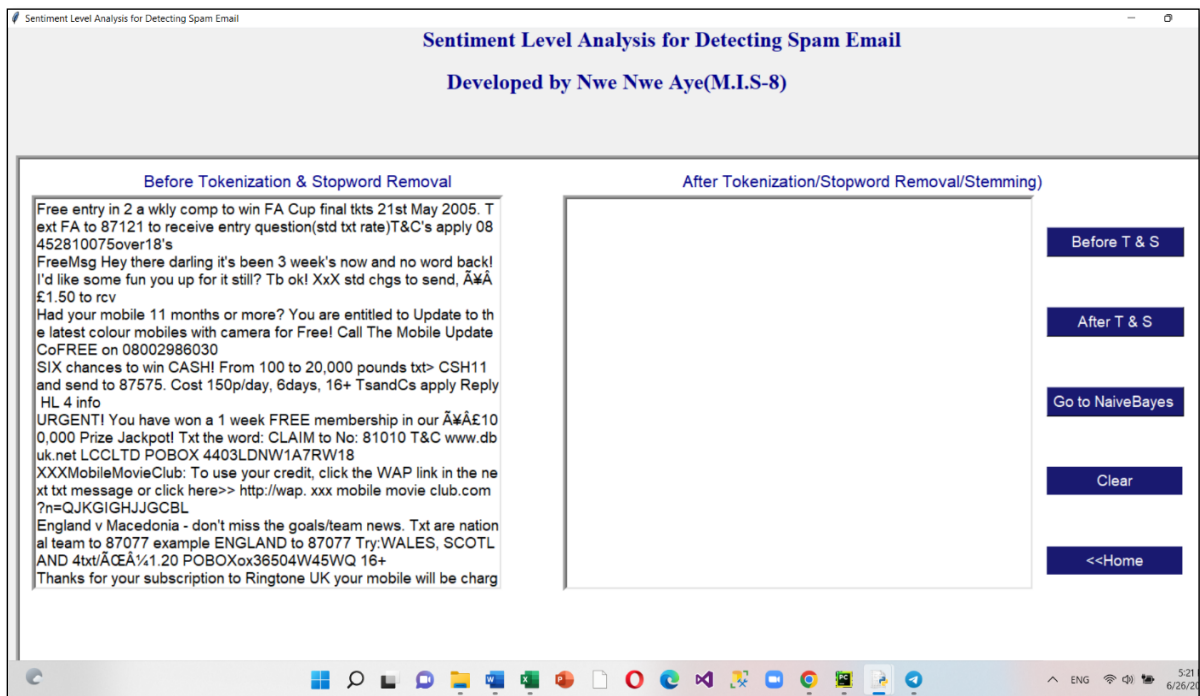


Figure 4.5 Before Tokenization and Stopword Removal

Next, after clicking the token and stopword removal button, the user would have seen in figure 4.6 dialog box which has appeared the pre-processed result documents by tokenizing, stopword removing and stemming for each word text respectively.

Firstly, the individual word tokenizing means split the words from the training data that formed the clean data. Tokenizing is done by NLTK string tokenizer. For instance, the input text “Fall in love” would be tokenized as “fall, in, love”.

In this figure, the step of removing stop-words that carry no particular meaning such as “a”, “and”, “the” and some other common word should be eliminate. The system removes the stopwords and stemming which is the process of reducing a word into its stem, i.e. its root form. The root form is not necessarily a word by itself, but it can be used to generate words by concatenating the right suffix. For example, the words “fish, fishes and fishing” all stem from fish, which is a correct word. On the other side, the words “study, studies and studying” stems from **studi**, which is not an English word.

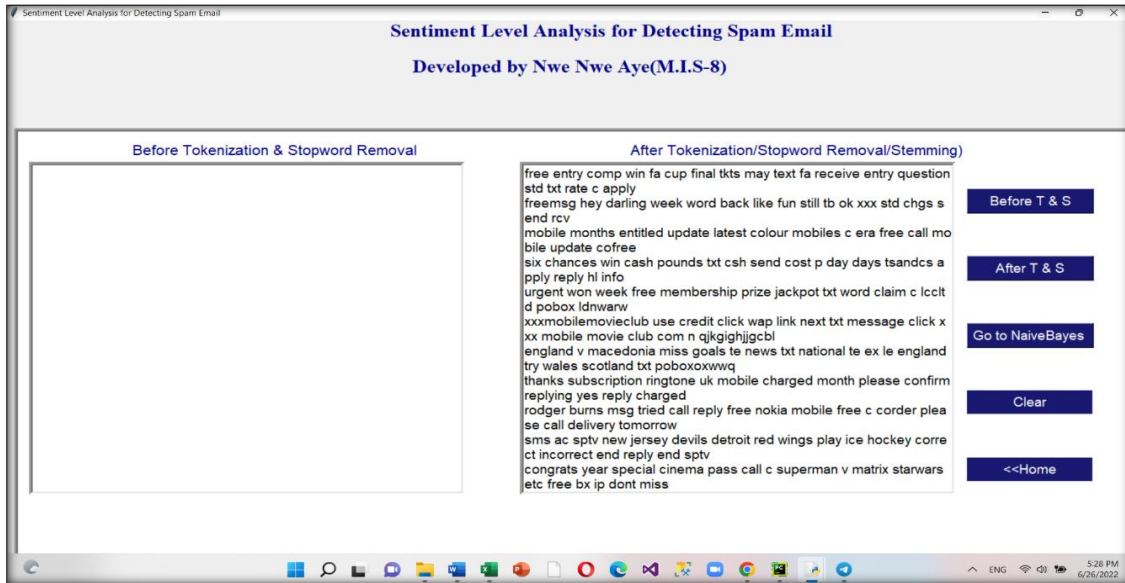


Figure 4.6 Pre-processing of Training Data

For training data, the result has been detected. If the user may click Go to Naïve Bayes button, the features/predictors used by the Multinomial Naïve Bayes classifier are the frequency of the words that have been presented in that training documents. Feature vectors represent the frequencies with which certain words have been generated by a multinomial distribution.

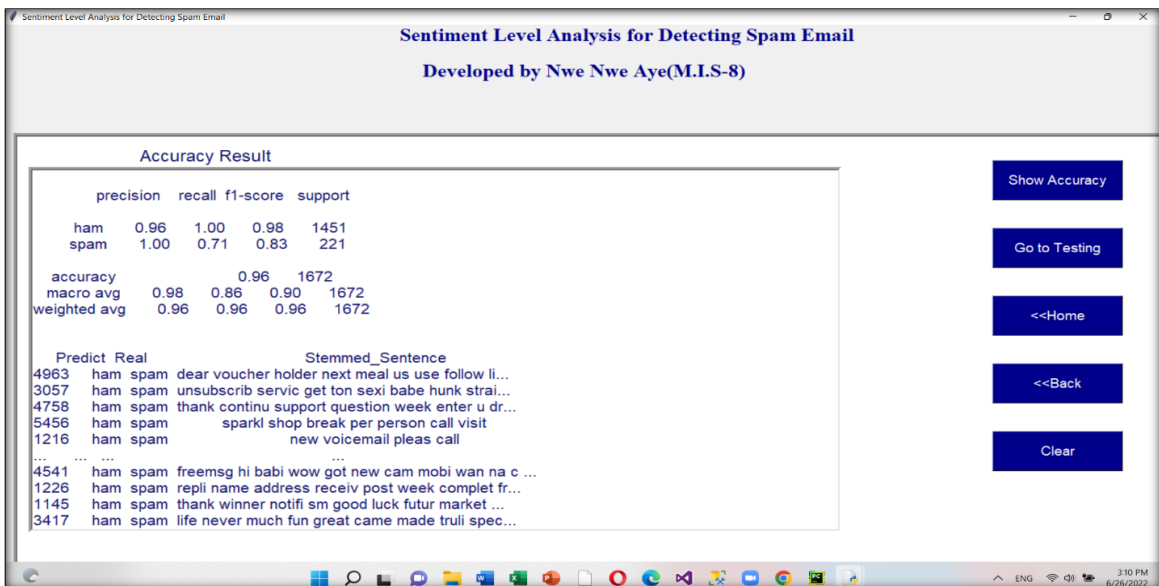


Figure 4.7. Accuracy Result of Multinomial Naïve Bayes Classifier

In this section, the user would have seen Show Accuracy button which have been processed as a result of Multinomial Naïve Bayes Classification Accuracy. For Training

data, the classifier's accuracy result has been detected and performed by supporting the terms of precision, recall, f-measure and accuracy rate.

Next, if the user may click Go to Testing button, the user would have seen Import Testing Data as presented by figure (4.8). To predict the spam or ham email class label for testing data, firstly import testing data of SMS email messages.

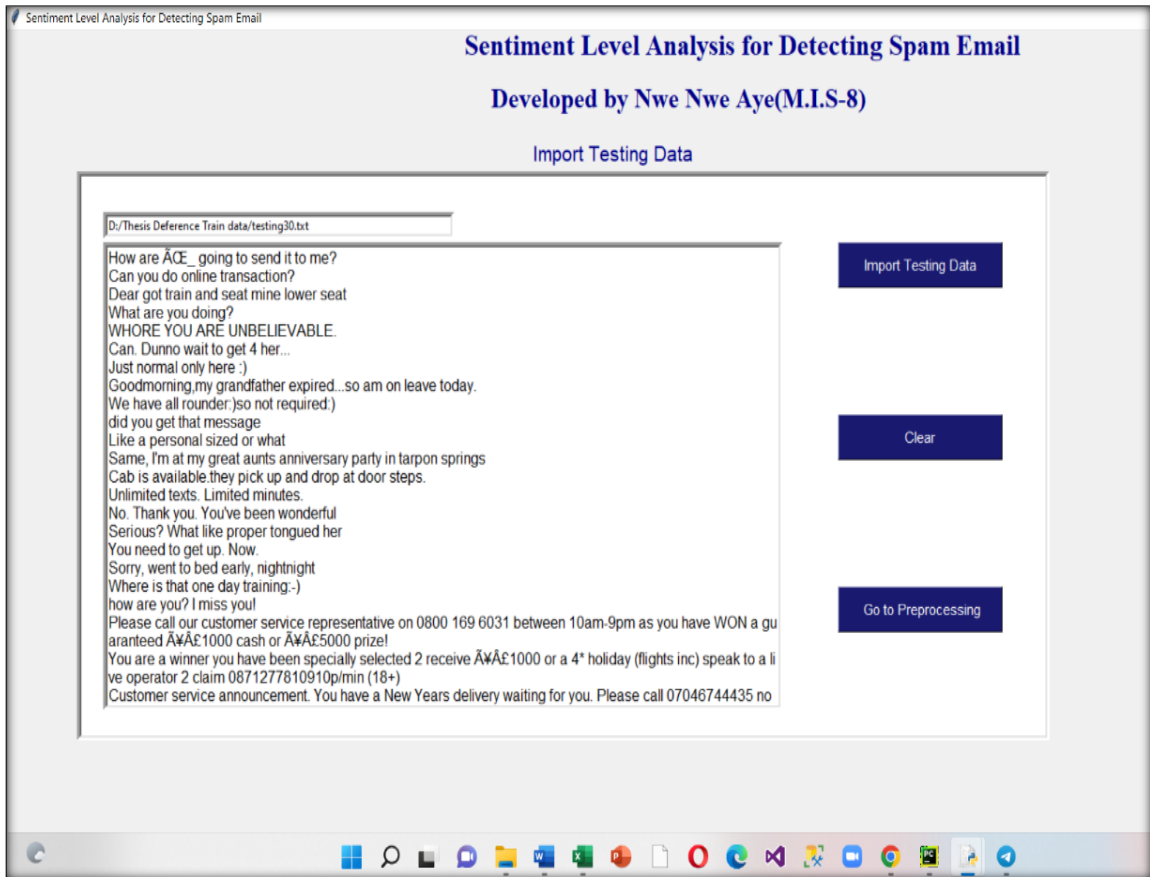


Figure 4.8 Import Testing Data

Preprocessing is important before extracting the feature words in each documents. Removing the stopwords which include verb to be, pronouns, prepositions and conjunctions not to deliver significant statistics for Sentiment analysis. So, the stopwords are eliminated. The steps of preprocessing are as follows:

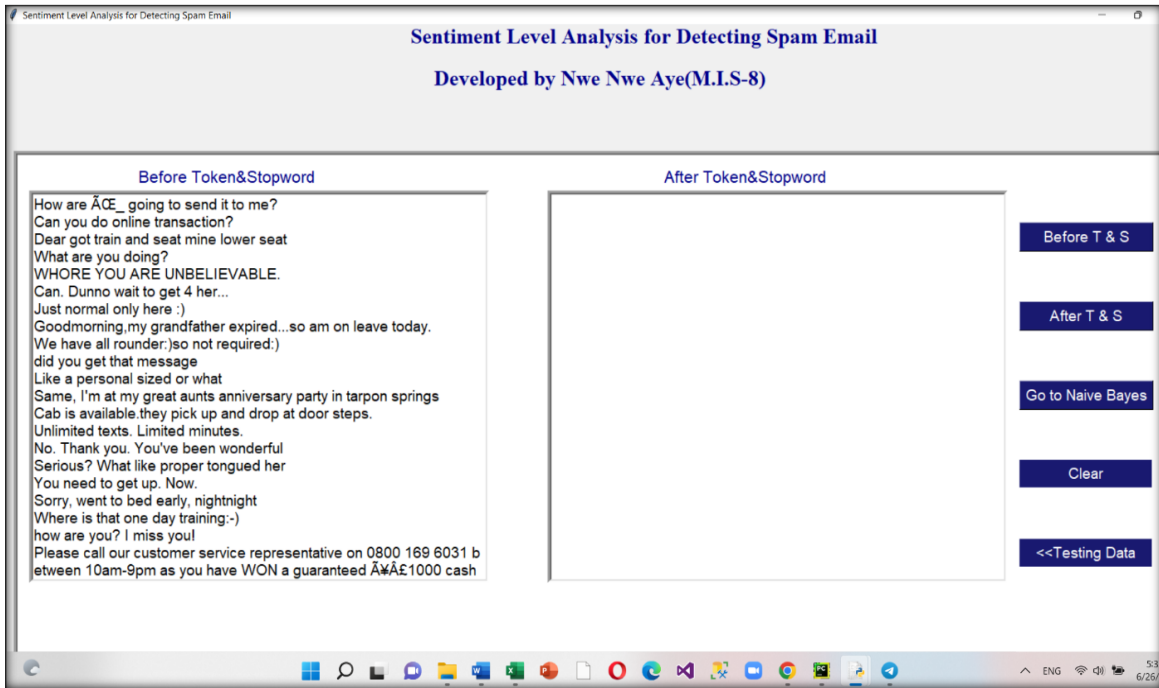


Figure 4.9 Before Tokenization and Stopword Removal for Testing Data

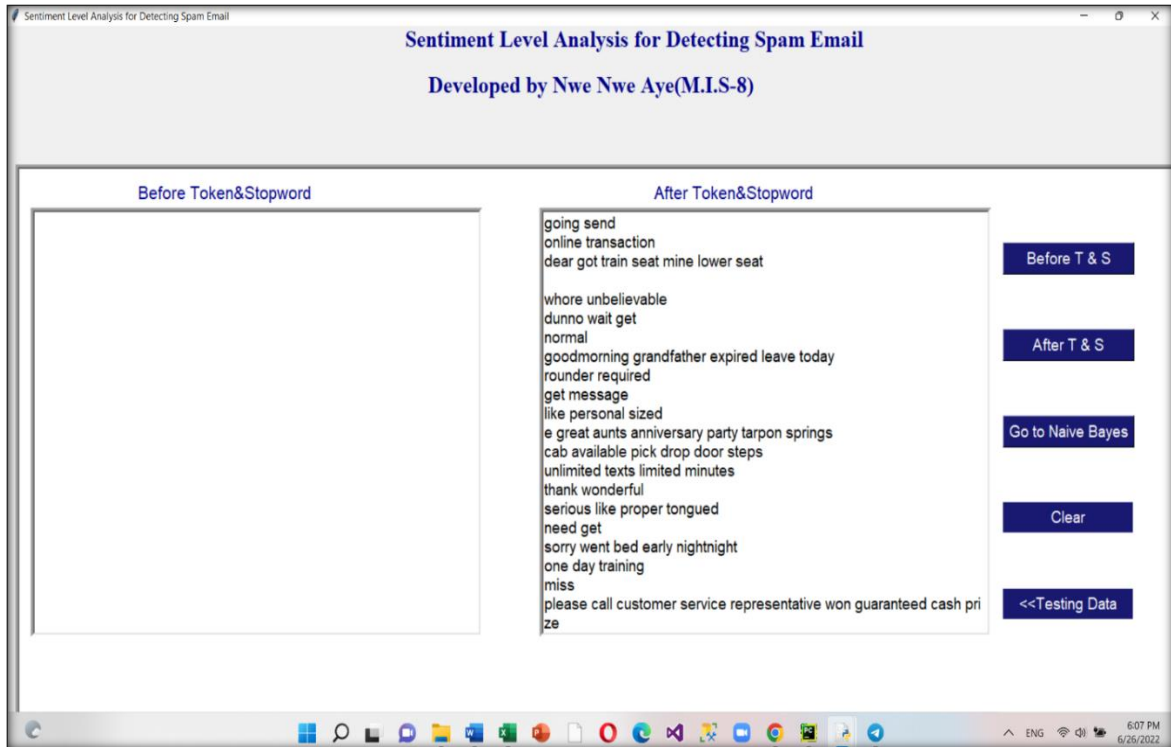


Figure 4.10 Pre-processing of Testing Data

And then to classify for testing data, the user may firstly import testing data, secondly click After Token and Stopword button which removes the stopwords, tokenization and stemming. And go to naïve bayes button to predict and classify spam or ham mail by using the Multinomial Naïve Bayes classifier which gives the most probable result of each document.

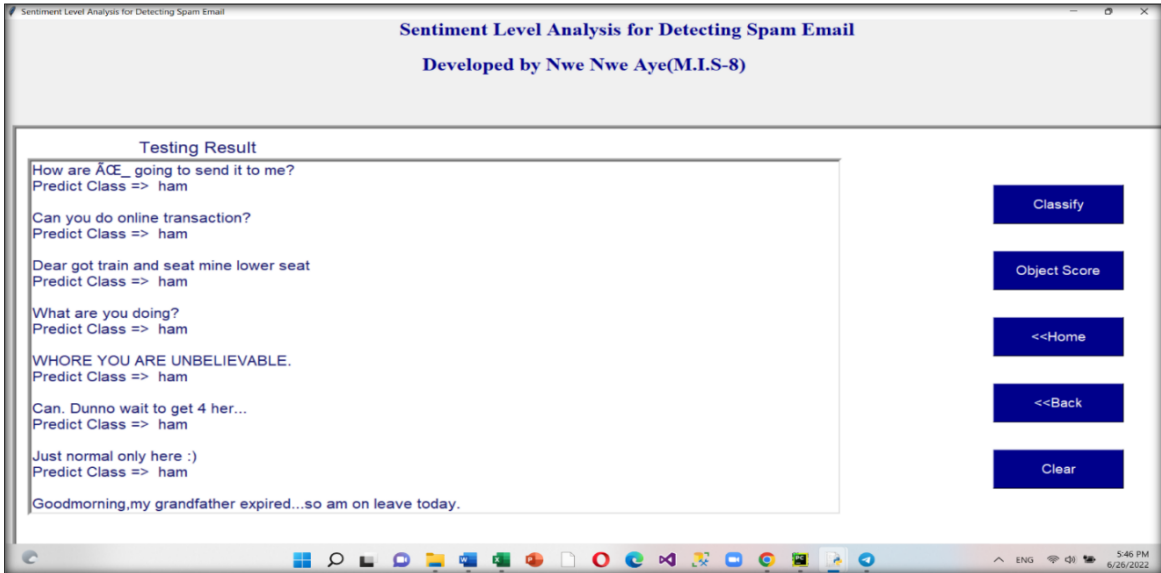


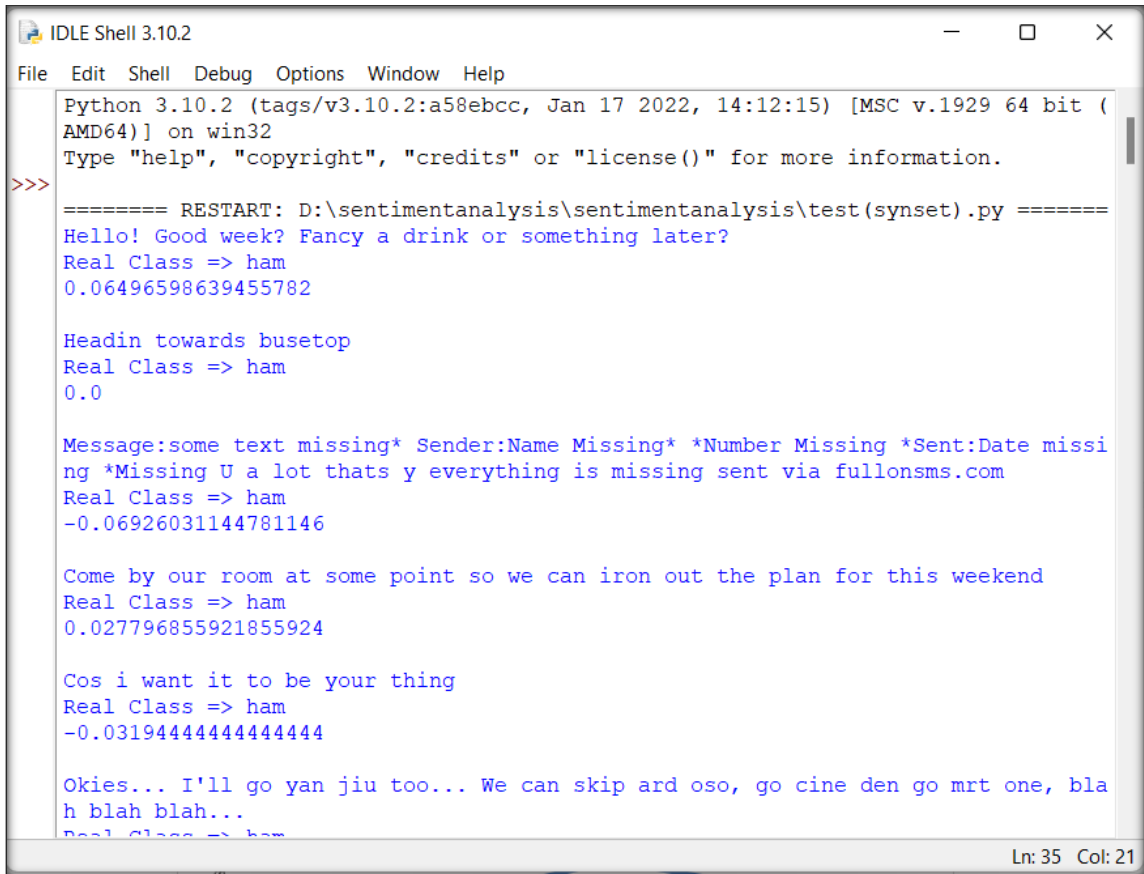
Figure 4.11 Classify Result of Testing Data

Object Score calculation results of testing documents would be processed by using the Multinomial Naïve Bayes Classifier also as shown in the following figure (4.12).



Figure 4.12 Object Score Result by using MNB

Sentiment Score classifying result of testing documents would be processed by using SentiWordNet3.0 as shown in the following figure (4.13).



```
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\sentimentanalysis\sentimentanalysis\test(synset).py =====
Hello! Good week? Fancy a drink or something later?
Real Class => ham
0.06496598639455782

Headin towards busetop
Real Class => ham
0.0

Message:some text missing* Sender:Name Missing* *Number Missing *Sent:Date missing *Missing U a lot thats y everything is missing sent via fullonsms.com
Real Class => ham
-0.06926031144781146

Come by our room at some point so we can iron out the plan for this weekend
Real Class => ham
0.027796855921855924

Cos i want it to be your thing
Real Class => ham
-0.031944444444444444

Okies... I'll go yan jiu too... We can skip ard oso, go cine den go mrt one, blah blah blah...
Real Class => ham
```

Figure 4.13 Object Score Result by using SentiWordNet 3.0

Figure 4.14 is shown as the accuracy of various training data and testing data by classifying with multinomial naïve bayes. The accuracy result for 140 training data and 60 testing data is 90%. This system is evaluated 97% of accuracy for 3901 training data and 1672 testing data.

The more training data, the higher accuracy by using Multinomial Naïve Bayes Classifier. If the total number of spam and ham are same, the class result is correct for classification.

Finally, for the aim of sentiment level analysis for detecting spam email performance evaluation, this system will compare the Multinomial Naïve Bayes classification result with the evaluated result of Classifying Sentiment Score result as the following figure 4.15.

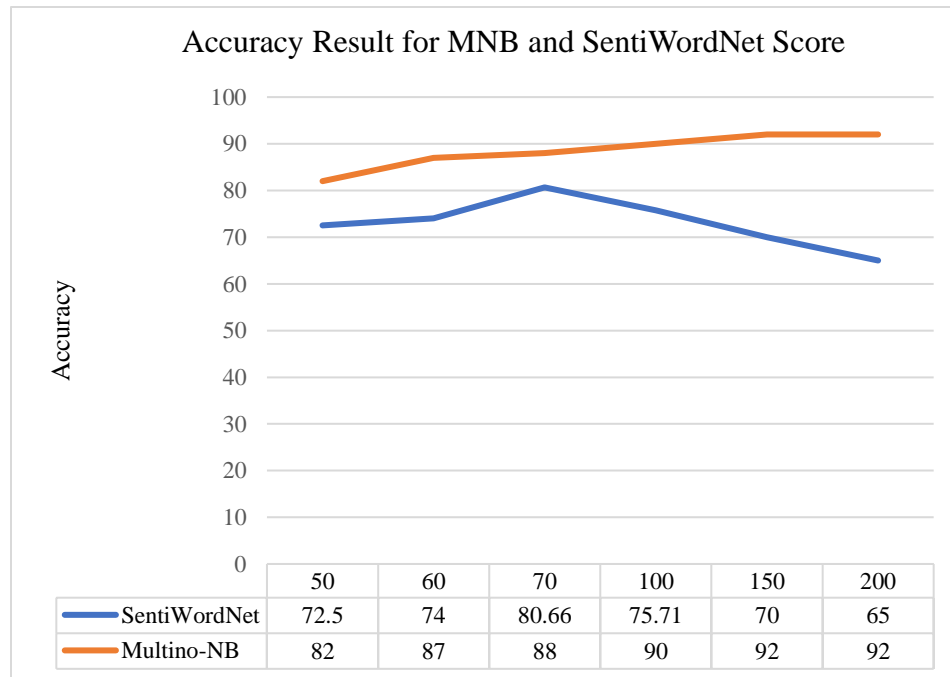


Figure 4.14 Accuracy Result for MNB and SentiWordNet Score

CHAPTER 5

CONCLUSION AND FURTHER EXTENSION

5.1 Conclusion

Email is an excellent initial point for social media analysis. In this system, the collected spam mails are preprocessed using Natural Language Toolkit techniques. The features of the spam mail are selected based on Multinomial Naïve Bayes classifier is used to classify the spam mail as spam and ham. This proposed system would be easy for user to obtain the summarized report about the opinion from spam mails. It is also used to powerful in decision making process in their daily life activities. Sentiment analysis is very powerful to extract the features from the data. The system can also be used for social media, life science and personal filtering.

For SMS message of spam filter, a successful implementation uses the multinomial Naive Bayes algorithm by the application of the concepts of conditional probability. Spam emails have become a major concern for the internet community. Email filtering is very essential necessary for email communication. The Multinomial Naive Bayes algorithm is a powerful tool that can be used to detection of spam emails. It is relatively simple to program and use in the real world. The key advantage of this system is its highly scalable with the number of predictors and data points, is very effective and is also adaptive. Furthermore, it is not sensitive to irrelevant features, it doesn't require as much training data, learn new spammer tactics automatically.

5.2 Limitation and Further Extension

The limitation of this system is hard to get good emails corpora, need huge attributes, and need lots of training data. All features are independent which may not be true in reality. With small data sets, the precision will be less, requires training data sets in order to train the model, and spammers keep improvising, this model would need to be

periodically updated. Furthermore, If the user test data set has a categorical variable of a category that wasn't present in the training data set, the Naive Bayes model will assign it zero probability and won't be able to make any predictions in this regard.

Reference

- [1] P. D. Turney, "Mining the web for synonyms: PMI-IR versus LSA on TOEFL," *Proc. 12th Eur. Conf. Mach. Learn. (ECML-2001), Freiburg, Ger.*, pp. 491–502, 2001.
- [2] Q. Ye, B. Lin, and Y. Li, "Sentiment classification for Chinese reviews: A comparison between SVM and semantic approaches," *Mach. Learn. Cybern. 2005. Proc. 2005 Int. Conf.*, vol. 4, no. August, pp. 2341–2346, 2005.
- [3] Y. Zhu, Z. Wen, P. Wang, and Z. Peng, "A method of building Chinese basic semantic lexicon based on word similarity," *Pattern Recognition, 2009. CCPR 2009. Chinese Conf.*, pp. 1–4, Nov. 2009.
- [4] K. S. Vishnu, T. Apoorva, and D. Gupta, "Learning domain-specific and domain-independent opinion oriented lexicons using multiple domain knowledge," *Contemp. Comput. (IC3), 2014 Seventh Int. Conf.*, pp. 318–323, 2014.
- [5] G. Demiroz, B. Yanikoglu, D. Tapucu, and Y. Saygin, "Learning domain-specific polarity lexicons," *Data Min. Work. (ICDMW), 2012 IEEE 12th Int. Conf.*, pp. 674–679, Dec. 2012.
- [7] Dipti Sharma¹, Dr. Munish Sabharwal², Dr. Vinay Goyal³, and Dr. Mohit Vij⁴, "SA Techniques for Social Media Data: A Review", Chandigarh University, Mohali, Punjab, India, 2019.
- [8] J Kaur & M Sabharwal. "Spam Detection in Online Social Networks Using Feed Forward Neural Network". In RSRI Conference on Recent Trends in Science and Engineering, vol. 2, pp. 69-78, 2018.

- [9] Goel, A., Gautam, J., & Kumar, S. "Real time Sentiment Analysis of tweets using Naïve Bayes". In 2nd International Conference on Next Generation Computing Technologies (NGCT), pp. 257-216, 2016.IEEE.
- [10]. Al-Smadi, M., Al-Ayyoub, M., Jararweh, Y., Qawasmeh, O.: Enhancing aspect-based Sentiment Analysis of Arabic hotels' reviews using morphological, syntactic and semantic features. *Inf. Process. Manag.* (2018).
- [11]. Zainuddin, N., & Selamat, A. "Sentiment Analysis using Support Vector Machine". In International Conference on Computer, Communications, and Control Technology (I4CT), pp. 333-337, 2014, IEEE.
- [12]. Sachdeva K., Kaur A. & M. Sabharwal. "Face Recognition using Neural Network with SURF Technique". In International Conference on Futuristic Trends in Computing and Networks, vol. 2(1), pp. 256-261, 2018.
- [13]. Vega, L., & Mendez-Vazquez, A. "Dynamic Neural Networks for Text Classification". In International Conference on Computational Intelligence and Applications (ICCIA), pp. 6- 11, 2016, IEEE.
- [14]. Patil, S., Gune, A., & Nene, M. "Convolutional neural networks for text categorization with latent semantic analysis". In International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), pp. 499-503, 2017, IEEE.
- [15]. Kotenko, I., Chechulin, A., & Komashinsky, D. "Evaluation of text classification techniques for inappropriate web content blocking". In 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), pp. 412-417, 2015, IEEE.

[16]. Xia R, Xu F, Yu J, Qi Y, and Cambria E. "Polarity shift detection, elimination and ensemble: a three-stage model for document-level Sentiment Analysis ". In *Information Processing and Management*, vol. 52, pp. 36–45, 2016.

[17]. Buddeewong, S., & Kreesuradej, W. "A new association rule-based text classifier algorithm". In *17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05)*, 2005.

[18]. Unnisa, M., Ameen A., & Raziuddin, S. "Opinion Mining on Twitter Data using Unsupervised Learning Technique". In *International Journal of Computer Applications*, pp.0975 – 8887, Vol. 148, 2016.

[19]. Park, S., & Kim, Y. "Building thesaurus lexicon using dictionary-based approach for sentiment classification". In *IEEE 14th International Conference on Software Engineering Research, Management and Applications (SERA)*, 2016.

[20] Das, S. R. *News Analytics: Framework, Techniques and Metrics*, chapter 2. Wiley Finance, 2010. *The Handbook of News Analytics in Finance*.

[21] Liu, B. Sentiment Analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167, 2012.