Smart Email Reply System Based on Sentiment

Gireesh Babu C N
Dept of Information Science and Engineering,
BMS Institute of Technology and Management, Bengaluru, Karnataka

Abstract- *Automation plays an important role in today's world. Through automation, many things which humans can do can be done by machines. It can be done through writing scripts that performs task on behalf of humans. Automation of email can be done easily with the help of Python. In this paper, the priority is to extract information from email received, analyze body for sentiment and based on the sentiment detected, an automatic email response will be sent to the sender*

Keywords- email automation, sentiment detection, smtp, Naïve Bayes Classifier

I. INTRODUCTION

Sending an email to every individual is not possible manually. For manually it will take a lot human resources thus increasing cost to company. And also till a limited number people you can send manually even if you have human resource. So comes the automation. Automation can be done through writing scripts that performs task on behalf of humans. Automation of email can be done easily with the help of Python. Python supports many libraries that support email sending and receiving. In this seminar we are interested in responding the query we got through email. For this automation, we are using google api client for receiving emails from server and smtp to send email to the recipient. For automating email id and password is given to a script which on behalf user sends or receives email.

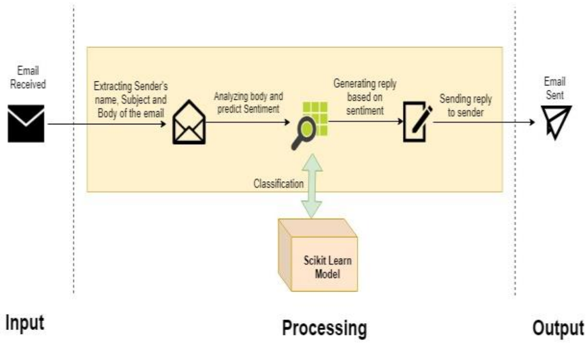
II. RELATED RESEARCH

- *Research on Sentiment Analysis*

Yuling[1] described a Convolutional Neural Networks (CNNs) model combined with SVM text sentiment analysis. The experimental results show that the proposed method improves the accuracy of text sentiment classification over tradition CNN. But SVM has one disadvantage that it takes long training time for large datasets. Jie[2] discussed about using dependency parsing with sentiment relationship migration and modified distance for sentiment analysis of short text in microblogs. One limitation is that restriction that short text in microblogs cannot exceed 140 words makes microblog short text generally consists of no more than three sentences. Emmanuel[3] uses Genetic Algorithm that learn whether words occurring in a text corpus are either sentiment or amplifier words, and their corresponding magnitude. Sentiment words, such as 'horrible', add linearly to the final sentiment. Amplifier words in contrast, which are typically adjectives/adverbs like 'very', multiply the sentiment of the following word. ChXiaojie[4] showed Multi word embedding in the semantic, syntactic, and sentiment information, followed by proceeding the word embedding fusion. K-means text clustering is applied by dividing similar text into the same cluster, thus improving the classification accuracy. Negatives and conjunctions in contexts can lead polarity shifts. Eg. A positive sentiment text may contain a negation word like not which here shifts polarity towards negative. Junaid[5] proposes a technique which uses word lexicons, emoticons, negations and intensity modifiers. Their approach follows Ekman's emotion model. Word lexicons are generated using WordNet from a set of seed words collected manually. Handling negations is a very complex task. It has achieved relatively lower precision for sadness and anger. Hongming[6] proposed in order to achieve the proposed target, the fraud type is summarised in social engineering criteria through literature review; a semantic web database is established to extract and store information; a fuzzy logic control algorithm is constructed to allocate email categories. The proposed approach will help users to distinguish the categories of emails. Nilesh[7] aims to develop automatic natural language processing system for extraction of emotions from text. Class Sequential Rules (CSR) are applied for obtaining the patterns. Algorithm developed is dataset independent. Ming-Hsiang[8] proposes a long-short term memory (LSTM)-based approach to text emotion recognition based on semantic word vector and emotional word vector of the input text. For each word in an input text, the semantic word vector is extracted from the word2vec model. Besides, each lexical word is projected to all the emotional words defined in an affective lexicon to derive an emotional word vector. R Sureshwaran[9] proposes the SMTP E-mail system protocol that includes new active monitoring algorithm architecture to improve the current E-mail system protocol functions and detect the SMTP protocol failure during the process of sending E-mail messages. Yunong[10] examined the effectiveness of emoticon emotion distribution by sentence emotion classification based on word emotion lexicon and emoticon emotion distribution. They constructed two word lexicons by using the word emotion intensities and frequency respectively, and calculate the emoticon emotion distribution by the co-occurrence of emoticon and sentence labels. After going through different papers, we looked on different insights and then we tried several algorithms and checked their accuracy and decided to use SMTP for email sending and Naïve Bayes classifier for sentiment classification.

IIII. SENTIMENT ANALYSIS USING NAÏVE BAYES CLASSIFIER

With the introduction of machine learning, machines are now becoming smarter. They are smart enough to learn from the instances they are experiencing. Sentiment analysis is an important aspect of machine learning. Sentiment analysis is capable of predicting sentiment from different types of files like audio, text, video etc. We are using Naïve Bayes Classifier for this purpose. But before applying Naïve Bayes, we are using Google API for Gmail. The system works in 3 steps as shown in below figure. Each step is explained in next three subsections.



- *Google API for Gmail for Receiving Emails*

Google Gmail API is the API provided by Google for the use of gmail from python script. With the help of it you can read the gmail messages from your program. It will call method that sends request to get the received messages. For getting the received messages, it will first authenticate your gmail account. For authentication purpose, your credentials are stored in a separate JSON file. In this way your password cannot be seen from the code. It has many different functions for various gmail operations. You can get the latest email using get() method. After getting message details, pattern search is applied and then useful information is extracted like sender's email, email subject, email body etc. Then Naïve Bayes is applied to the body content for sentiment prediction.

- *Naïve Bayes Classifier to Predict Sentiment*

The Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

For some types of probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods. The type used here is Multinomial Naïve Bayes classifier.

Multinomial Naive Bayes classification algorithm tends to be a baseline solution for sentiment analysis task. The basic idea of Naive Bayes technique is to find the probabilities of classes assigned to texts by using the joint probabilities of words and classes. Given the dependent feature vector (x_1, …, xn) and the class Ck. Bayes' theorem is stated mathematically as the following relationship:

$$P(C_k \mid x_1, \ldots, x_n) = \frac{P(C_k) P(x_1, \ldots, x_n \mid C_k)}{P(x_1, \ldots, x_n)}$$

According to the "naive" conditional independence assumptions, for the given class Ck each feature of vector xi is conditionally independent of every other feature xj for i≠j.

$$P(x_i \mid C_k, x_1, \ldots, x_n) = P(x_i \mid C_k)$$

Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

We have trained the model with custom built dataset. After the model is trained, it is used to predict sentiment in the body.

- *Using SMTP to Send Respose*

After the sentiment is predicted, an email response is made based on the sentiment predicted. If sentiment is positive, response body will contain some happy line like “We are happy that you like our product” or “Thank you for choosing us followed by You will be contacted soon.” Similarly if sentiment found is negative, email response will be in apologitic manner like “Sorry for the inconvenience followed by You will be contacted shortly or soon.” This way it will make more sense to the sender according to his or her mood. After response is created it is sent back the original sender using python built-in SMTP library.

IV. RESULTS AND CONCLUSION

This paper deals with automation of email based on sentiment present in the email. This work is mainly for customer care management email automation system. Suppose a customer is happy with your product and sends you a positive feedback or query, then automated response will first greet him and say thank for choosing your product followed by they will be contacted shortly. Similarly for negative customer review or query, the email response generated will be in apologetic manner so that the customer will feel that they are treated well. Sending suitable response based on user’s feeling makes more sense. Its accuracy is pretty good but have some limitations and future scope that is discussed in next sections.

There are some limitations which we faced while implementation. For getting the latest mail you need to check again and again. Analysing body for sentiment will take some time if email body is too large. Since no mobile or web app is there so every time you need to run spyder on system.

There are certain aspects where it can be enhanced. Different emotions rather than positive and negative sentiment can also be predicted. Allow analysing email body for media files present in it like photos or videos, not only from text. Accuracy can be increased further by increasing training data.

REFERENCES

[1] Y. Chen and Z. Zhang, "Research on text sentiment analysis based on CNNs and SVM," 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, 2018, pp. 2731-2734. doi: 10.1109/ICIEA.2018.8398173J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] J. Li and L. Qiu, "A Sentiment Analysis Method of Short Texts in Microblog," 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, 2017, pp. 776-779. doi: 10.1109/CSE-EUC.2017.153

[3] Emmanuel Dufourq and Bruce A. Bassett. 2017. Automated classification of text sentiment. In Proceedings of the South African Institute of Computer Scientists and Information Technologists (SAICSIT '17). ACM, New York, NY, USA, Article 11, 10 pages. DOI: 10.1145/3129416.3129420

[4] Xiaojie Sun, Menghao Du, Hua Shi, and Wenming Huang. 2018. Text Sentiment Polarity Classification Method Based on Word Embedding. In Proceedings of the 2018 2nd International Conference on Algorithms, Computing and Systems (ICACS '18). ACM, New York, NY, USA, 99-104. DOI: 10.1145/3242840.3242850

[5] J. Akram and A. Tahir, "Lexicon and Heuristics Based Approach for Identification of Emotion in Text," 2018 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, 2018, pp. 293-297. doi: 10.1109/FIT.2018.00058

[6] S. B. Rathod and T. M. Pattewar, "Content based spam detection in email using Bayesian classifier," 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, 2015, pp. 1257-1261. doi: 10.1109/ICCSP.2015.7322709

[7] N. M. Shelke, S. Deshpande and V. Thakre, "Class sequential rules for emotion extraction from text," 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, 2017, pp. 758-763. doi: 10.1109/I-SMAC.2017.8058281

[8] M. Su, C. Wu, K. Huang and Q. Hong, "LSTM-based Text Emotion Recognition Using Semantic and Emotional Word Vectors," 2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia), Beijing, 2018, pp. 1-6. doi: 10.1109/ACIIAsia.2018.8470378

[9] R. Sureswaran, H. A. Bazar, O. Abouabdalla, A. M. Manasrah and H. El-Taj, "Active e-mail system SMTP protocol monitoring algorithm," 2009 2nd IEEE International Conference on Broadband Network & Multimedia Technology, Beijing, 2009, pp. 257-260. doi: 10.1109/ICBNMT.2009.5348490

[10] Y. Wu, X. Kang, K. Matsumoto, M. Yoshida, K. Xielifuguli and K. Kita, "Sentence Emotion Classification for Intelligent Robotics Based on Word Lexicon and Emoticon Emotions," 2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE), Lanzhou, 2018, pp. 38-41. doi: 10.1109/IRCE.2018.8492969