TEXT SUMMARIZATION AND CLASSIFICATION OF CONVERSATION DATA BETWEEN SERVICE CHATBOT AND CUSTOMER

1st Tanmayee Behere
Department of Computer Engineering
Cummins college of Engineering for Women
Pune, India
tanmayee.behere@cumminscollege.in

2nd Avani Vaidya
Department of Computer Engineering
Cummins college of Engineering for Women
Pune, India
avani.vaidya@cumminscollege.in

3rd Anamika Birhade
Department of Computer Engineering
Cummins college of Engineering for Women
Pune, India
anamika.birhade@cumminscollege.in

4th Komal Shinde
Department of Computer Engineering
Cummins college of Engineering for Women
Pune, India
komal.shinde@cumminscollege.in

5th Pranjali Deshpande
Department of Computer Engineering
Cummins college of Engineering for Women
Pune, India
pranjali.deshpande@cumminscollege.in

6th Sunita Jahirabadkar
Department of Computer Engineering
Cummins college of Engineering for Women
Pune, India
sunita.jahirabadkar@cumminscollege.in

***Abstract* – In any business, a humongous amount of data is generated within each fraction of a second by each and every software application. However, processing this voluminous data is a tedious task especially when the data is in textual form. In this scenario, Natural Language Processing has contributed majorly in this area of study. In NLP, keyword extraction plays a pivotal role in text processing which helps the readers to determine whether to read a document or a webpage. The system designed in this paper computed extractive text summarization using Graph-based technique and TextRank algorithm on conversation data between the user and the service chatbots which in fact an offline conversation. This summary is then consumed by a classification module which is trained using Naive Bayes classifier to evaluate in which of the three categories the conversation falls into: 1. Help 2.Complaint 3. FakeCustomer. This system can be utilized by various companies such as online shopping websites, software companies to determine in which aspect immediate attention is required. The system is also experimented using different thresholds for determining the length of summary produced and its corresponding accuracy.**

Keywords* – *Extractive Summary, TextRank, Naive Bayesian, Cosine Similarity, Page Rank, Classification, Supervised Learning.

I INTRODUCTION

Volume of text in service companies while conversing with customers is increasing day-by-day and has become a major source of information and a lot of effort and time is required to go through them , thus by providing a summary it reduces reading time and only the significant points are highlighted from the document.

Companies can detect flaws and improve quality of service by analysing complaints from customers, so helping them identify complaints from other categories is required. Training the employees or chatbot for queries by customers based on these categories would help to get a broader view about the type of customer and responses to be given. It would also help determine the new training programs required for customers. Companies can further use these training questions to add them in FAQ. It will also aid companies identify customers who may be their competitors trying to harm them or trying to gain the inside information through their chatbot.

Two major techniques have been proposed for computing text summarization: Extractive Text Summarization, which focuses on pulling key phrases and key sentences from the given data to

978-1-7281-6823-4/20/$31.00 ©2020 IEEE

create a summary without changing original text. Abstractive text summarization focuses on understanding the appropriate meaning of given natural sentences and generation of natural language.

The input data of conversations between the service chatbots and the customer, the designed system will analyze this data and compute extractive text summary to classify it as customer complaints, help for customers, fake customers, with further generation of reports for analysis of most frequently occurring terms for each category. The report will brief about the most frequent complaints registered in each category.

The paper is organized into following sections:

Section 2 depicts Literature Review, Section 3 focuses on Designed System and Section 4 is the Conclusion.

II LITERATURE REVIEW

Many techniques have been introduced to compute abstractive and extractive text summarization.

Tulasi Prasad Sariki, G. Bharadwaja Kumar, Utkarsh Shukla, Ayush Mishra [1] proposed a framework which gives better results for extractive text summarization process which combines three different methods namely Statistical, NER-based and Cue phrase method. In the proposed framework, important sentences from the text to be summarized are chosen by first scoring each sentence using a statistical approach that uses term-frequency for ranking the sentence. Secondly Named Entity Recognition is used to identify the text segments that are important. The third step in the summarization process is to identify the sentences which consist of cue phrases like “significantly” or “in particular” etc which is followed by candidate sentences. Top N sentences are chosen where N represents compression ratio, and then similarity between selected sentences is calculated using Word Movers Distance Method to remove duplication. The proposed framework requires no training overhead and also it gives better performance results by exceeding the ROGUE-L scores. However, the shortcomings of using statistical approach is that it does not provide linguistic knowledge processing or semantic relation mapping.

Wen Xiao,Giuseppe Carenini [4], provided a way of computing extractive text summarization of long documents by combining Global and Local Context in which they made use of non auto-regressive and Average Word Embeddings for sentence encoding. The proposed system consisted of three components: A. Sentence encoder which maps sentences to word embeddings to a fixed length vector, where Average Word Embedding method is selected due to its better performance, which in sentence embedding is the average of its word embeddings. B. Document Encoder which uses a bi-directional recurrent neural network used to encode all sentences sequentially forward and backward. The proposed system makes use of Gated Recurrent Units represented by standard reset, update and new gates, where the outcome of this module for each sentence is two hidden states as forward and backward respectively. Sentence representation is computed by concatenation of backward and forward hidden state for each sentence, this also represents the current state but also captures the contextual information before and after the sentence. Document representation provides the global context for the document as a whole, computed by concatenating final states of forward and backward GRU. The local context of each sentence is captured through topic segment representation using LSTM-minus method, in which each topic segment is represented as the difference between hidden states of start and end of that topic .C. Finally, Decoder is used to determine whether the sentence is to be included in summary. The representations obtained, sentence, document and topic segment are combined by two ways: 1. Concatenation, where the obtained vectors are concatenated. The proposed model is tested on PubMed and arXiv datasets and evaluation of this model is done using oracle labels by Oracle Greedy Labelling algorithm. The model attains state-of-the-art on the two testing datasets, and shows promising results for longer documents on ROUGE and METEOR scores. However, the model depicts that redundancy of sentences is present in the summary and no work has been done on the same topic.

Jiacheng Xu and Greg Durrett[5], presents a neural model for single document summarization based on joint extraction and syntactic compression. Their model initially encodes the source document and sequentially selects the sentences for further compression. Training of this model is executed using oracle extractive-compressive summaries which also depicts a major challenge. The proposed model uses Bidirectional LSTM for computing sentence and document encoding where each word in a sentence is encoded using BiLSTM followed by the application of several convolution layers and increased pooling layers to create sentence representations.This process is followed by a decoder

stage which used a sequential LSTM decoder where, at any moment, sentence representation h of previous chosen sentence , the overall document vector and recurrent state is used to produce a distribution of the remaining sentences. The process till this stage depicts in fact the extraction process. After this, the text compression module computes compression on the selected sentences by evaluating the discrete options and decides whether to remove certain phrases. The given network uses a feedforward network which combines the encoded sentences and their compression to determine whether to delete the span.However, this model requires extensive datasets and a training model for making the network learn the compression rules which poses a primary disadvantage. As a matter of fact, the model has outperformed the work on CNN/Daily Mail corpus in terms of ROUGE and accomplished considerable achievement over extractive model and emerge to have adequate grammaticality conferring to human assessments.

Neelima G, Veeramanickam M.R.M, Sergey Gorbachev, Sandip A. Kale [6] proposed a deep natural language fuzzy processing method, to generate extractive text summary. The steps that are proposed in the system are as follows: The document is subjected to sentence-level tokenization that returns the sentences in the text and word-level tokenization that returns the words in those sentences. The second step is to preprocess the document that removes noise like new lines, brackets, special symbols etc. It also involves removal of stop words and performing stemming. Now, the third step is to pull important sentences from the document using fuzzy processing in which weighted frequency of each word is calculated. Last step is to score the sentences by adding the weighted frequency of each word present in the sentence and top n sentences form the summary. The proposed method helps in summarizing large documents into shorter texts and any webpage. It saves time of the user and helps to work with the relevant data. However, the shortcomings of approach as stated is that sentences with longer lengths and unrelated sentences from different parts of the document can be included in the summary which reduces the quality of summarized text.

El-Refaiy, Ahmed & Abas, A.R. & Elhenawy, Ibrahim[7] stated different methods for Extractive text summarization. One of the techniques is the Latent Semantic Analysis that helps us to understand the meaning of the words and the sentences in the context in which they are used with the help of statistical computations. The LSA technique comprises three main steps: The first one is to build an input matrix of the document to be summarized where columns represent the sentences and rows represent the words in the document. The cells represent the importance of the words in the sentence. The cell value is calculated using different approaches like calculating the number of times a word appears in a sentence, assigning 1/0 based on the existence of the word in the sentence, word's tf-idf value etc. The second step is the Singular Value Decomposition (SVD) that divides the input matrix into three other matrices and also, depicts the relationship between words and sentences. The first matrix depicts words against retrieved concepts. Second one represents scaling values and Third matrix depicts sentences against retrieved concepts. The third step after calculating SVD results is to select sentences using different algorithms as mentioned [7] to form the summary. LSA is unsupervised with no training required, that provides semantic representation of the text and also provides noise free information. However, the shortcomings is that it is difficult to map and maintain the correct meaning of each word to the context in which it is used.

Omar Al,Hassan K,Tarek F,Ahmed N,Khaled S [8] presented a technique computing extractive summarisation which produces 3-graphlets and 4-graphlets from the given dataset.The system initiated by first processing the abstracts by removing stopwords,words that are not nouns,adjectives and with length less than 5 characters. It also removes abstracts with less than 3 keywords, then tokenize the abstract using NLTK Abstract graphs are then computed with nides as words and edges as co-occurrence of two adjacent words. Subgraphs are generated using Depth First Search which depicts that for each word in the abstract 3 3-graphlets and 11 4-graphalets are possible. These graphlets are represented using 3 digit code with the first digit as the degree of the vertex, second digit as the addition of degrees of all vertices straight connected to the first vertex and third digit as the addition of degrees of vertices that are indirectly connected to the first. Naive Bayes Classifier makes use of predefined keywords for training. Hence this system used subgraph technique to primarily extract the keywords from the database of abstracts and then used Naive Bayes classifier to determine whether the chosen keyword is finally to be considered as a keyword or not. This classifier is built using Rapid Miner. The results produced by this technique were compared by a baseline generated by applying the traditional statistical method of TF/IDF on the same database

which showed promising outcomes in three lines of aspects Precision (76.32%) which indicated the number of extracted keywords that are useful ,Recall(62.88%) which is a Type I error that describes keywords extracted by this model but are not included the abstract's OT and 60.18% for F-measure which is a Type II error which depicts the keywords that were considered as abstract's OT but were not extracted by the proposed system.

Jilei Li with other researchers worked on classification methods for expert academic personas, where training datasets were constructed automatically. Further unified model is trained for the same. This model is used for multi-language purposes. This system uses Wikipedia for constructing datasets automatically containing Chinese and English texts for expert academic classifiers. Support Vector machines are used for training the classifier on Chinese and English datasets. According to the research the SVM classifier achieved the best results and also it outperforms the bayes and k-nearest neighbor methods. Limited training data is retrieved and selected from Wikipedia, whereas the method requires larger dataset. For the proposed system limited data is available, thus using SVM will result in low performance of the model. [10]

E. T. Lau with other researchers worked for modelling prediction and classification of academic achievement of students using the Artificial Neural Network (ANN). ANN is an effective but difficult mechanism for implementing nonlinear behaviors that frequently define the actual problems in the world. ANN performs similar to the human brain for completing a task with enhanced performance through training, repeated improvement and learning. There is a three layered neuron structure in ANN with input, hidden and output layers. The input is obtained in the form of numerical information along with activation values and feature sets. These values are further passed through the neuron network to the hidden layer. Further, the hidden layer computes the weighted sum of each input. These computed weights are summed at the output layer to obtain results with the help of activation function. If the threshold of activation function is surpassed then the summed neurons will convert mathematically. The researchers found 84.8 % accuracy after using ANN for classification. But it was found that ANN performs poorly in classification of students based on their gender due to more False Negative values obtained. This result was found because of imbalance between the ratio of different classes of training data used. So ANN performs poor when there is imbalance between datasets of different classes and requires larger dataset [11].

Simplest of all classification models is Naïve Bayesian classifier, where it assumes all features to be independent of each other. Two commonly used generative models of Naïve Bayes classifiers are the Multivariate Bernoulli model and the Multinomial model.

Multivariate Bernoulli model depends on binary data that is (0, 1). Every dimension in the feature vector is either 0 or 1. There are n dimensions in the feature vector where n is the total number of words in the documents.

$$P(x \mid \omega_j) = \prod_{i=1}^{m} P(x_i \mid \omega_j)^b \cdot (1 - P(x_i \mid \omega_j))^{(1-b)}$$

where, $(b \in 0,1)$

x_i – n dimensional feature vector and

ω_j – class label

In the Multinomial Naïve Bayes theorem, instead of binary values term frequency is used. Term Frequency is the number of times a word occurs in a particular document. For characterizing text documents, weighted term frequency is used rather than term frequency. As it is useful if stop words are not removed from the text. The weighted term is term frequency- inverse document frequency (tf-idf). This technique gives importance to a word in inversely proportional ratio of how frequent a word appears in all the documents [9].

$$\text{Tf-idf} = tf_n(t,d) \cdot idf(t)$$

$$idf(t) = \log(n_d / n_d(t))$$

where,

n_d – total number of documents and

$n_d(t)$ – number of documents containing the term t.

If the size of the bag of words is relatively large, observed comparisons prove that the multinomial model performs better than the multivariate social multi-Bernoulli model [12].

III DESIGNED SYSTEM

As per the discussion of techniques depicted in section 2, Graph-based approach for extractive text summarization is found to be suitable for the designed system since it provides specific topic or query-based summaries and requires no training being an unsupervised learning algorithm. The system comprises three modules as shown in Fig. 1 extraction, classification and report generation.

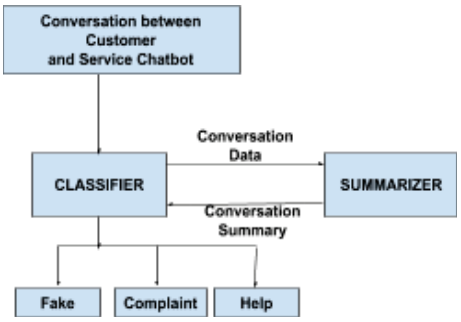

Figure 1: Block Diagram of Designed System

Module A: Extraction

The Extraction module produces a summary of data involved in the conversation between the service chatbot and the customer. The input to this module is the conversation between the service chatbot and customer.Graph based extractive algorithm - TextRank, is used since it is an unsupervised learning algorithm and no overhead of training the module is required. Also, the importance of a vertex, that is, a sentence is computed based on global information derived from the graph rather than from only the specific information of the vertex.

Module B: Classification

Input to this module is the extracted summary of conversation between service chatbot and customer. To classify the conversation into one of the three categories-Help, Compliant and Fake Customer Naive Bayes algorithm is used. The categories along with their corresponding probabilities are obtained using this algorithm.

This is achieved by the Countvectorizer which is used to tokenize the summary and generate vocabulary of words. While converting text to numbers, we only consider a maximum 1500 number of features which are most occurring words in the conversation data. There are two conditions for the words to be in the generated vocabulary, that are the word should be present in minimum 2 documents and should occur in maximum 70% of all documents. As words occurring in almost every document do not deliver special knowledge about the document and so these words are not useful for classification. Further the Tf-Idf is calculated on the generated vocabulary. Tf-Idf gives the relative importance of words in conversations. Finally the classification is done using a multinomial model of Bayes theorem.

Module C: Report Generation

Report is a bar graph with most occurring words in all documents on the x-axis and number of documents in which the corresponding word is present on the y-axis. This report is generated separately for each category. This report will be helpful for improving the business, which tells about most conversed words in all documents. Companies can further improve their strategies accordingly, save money, time and customer satisfaction.

IV CONCLUSION

In this paper, various techniques have been illustrated for extractive text summarization and classification. However, taking into consideration the requirements of the designed system, it makes use of Graph-based approach for extraction and Naive Bayes for classification. The advantage of this approach is that it gives better results since it focuses on the key sentences to be extracted rather than natural language generation by understanding the appropriate meaning of the data. In the designed system, a Graph based approach is used to extract the summary using Text Rank algorithm. However, no significant research has been done on the value of n, number of sentences to be chosen from the document as an extractive text summary. In the above designed system, for choosing the appropriate value of n, analysis was done by selecting values of n ranging from 1 to 2/3rd of the length of the original data. By putting n as 2/3rd of the length of the document, most promising results were achieved. Finally a report will be provided to the user, which displays the most occurred words by the documents in which the word has occurred.

The system provides dominance by saving efforts and time of reading and analyzing long documents since the users only have to work on relevant information. This is accomplished by system taking long conversations as input data and reducing them into concise and relevant sentences. These relevant sentences are denoted as a summary of the long document. Extractive text summarization also provides an edge to the system by providing better performance in terms of fluency and grammar. In particular, Graph-based approach of extraction can be adapted for computing query-based summaries and no training of the module is required.

In future, the system can be enhanced to incorporate the following features: According to individual's perception, the meaning of the conversation may vary and this results in ambiguity in the generated

output. This is referred to as bias that occurs due to different interpretations of the same conversation.

The results from the proposed system showed that there are fluctuations in accuracy of determining the conversation as Fake due to misinterpretation of natural language which gives rise to bias. Future work can be done on this feature to enhance the system. Extractive text summarization lacks redundancy in sentences and coherence across sentences. Further, the accuracy of extraction depends on the chosen affinity function. This accuracy can be improved in future with the use of different affinity functions.

V REFERENCES

[1] Tulasi Prasad Sariki, G. Bharadwaja Kumar, Utkarsh Shukla, Ayush Mishra,2019," An Adroit Approach for Extractive Text Summarization ",International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958, Volume-8 Issue-5.

[2] Yogan Jaya Kumar, Ong Sing Goh, Halizah Basiron, Ngo Hea Choon and Puspalata C Suppiah, 2016, "A Review on Automatic Text Summarization Approaches", Journal of Computer Science.

[3] Moratanch, N. & Gopalan, Chitrakala.,2017, "A survey on extractive text summarization",1-6.10.1109/ICCCSP.2017.7944061.

[4] Wen Xiao, Giuseppe Carenini, 2019, "Extractive Summarization on Long Documents by Combining Global and Local Context", Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing,Association for Computational Linguistics.

[5] Jiacheng Xu,Greg Durrett,2019,"Neural Extractive Text Summarization with Syntactic Compression", Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing,Association for Computational Linguistics.

[6] Neelima G, Veeramanickam M.R.M, Sergey Gorbachev, Sandip A. Kale, 2019, "Extractive Text Summarization using Deep Natural Language Fuzzy Processing", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue- 6S4.

[7] El-Refaey, Ahmed & Abas, A.R. & Elhenawy, Ibrahim, 2018, "Review of recent techniques for extractive text summarization", Journal of Theoretical and Applied Information Technology. 96. 7739-7759.

[8] Omar Alqaryouti,Hassan Khwileh,Tarek Farouk,Ahmed Nabhan, Khaled Shaalan, 2018, "Graph-Based Keyword Extraction", Springer International Publishing AG 2018.

[9] Naresh E, Vijaya Kumar B P, Pruthvi V S, Anusha K, Akshatha V , 2019, "Survey on Classification and Summarization of Documents", International Journal of Research in Advent Technology, Vol.7, No.6S,E-ISSN: 2321-9637.

[10] Jilei Li, GuangQuan Cheng, QiPeng Yin, JinCai Huang, WenChen Chen, JinMing Du, 2019, "A practical method for the expert academic personas classification based on text classifier", Springer-Verlag GmbH Germany, part of Springer Nature 2020.

[11] E. T. Lau, L. Sun, Q. Yang, 2019, " Modelling, prediction and classification of student academic performance using artificial neural networks", Research Article , SN Applied Sciences - A Springer Nature Journal.

[12] Gurinder Singh, Akriti Tyagi, Bhawna Kumar, Loveleen Gaur, 2019, "Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification", International Conference on Automation, Computational and Technology Management (ICACTM) Amity University.