International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.06, April 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

Comparative analysis on Clue Word Summarization and Latent Dirchlet Allocation Algorithm

Kuldeep Kaur[1], Anantdeep Kaur

[1.,2]*Department of Computer Engineering, Punjabi University, Patiala, India*

ABSTRACT

Summarizing email conversations are challenging due to the characteristics of emails, especially the conversational nature. Most of the existing methods dealing with email conversations use the email thread to represent the email conversation structure, which is not accurate in many cases. We are presenting an approach which will help to remove flaws of Clue Word Summarization (CWS) algorithm. In CWS the emails are marked with only one category whereas according to LDA one email may not be categorized into one category only. The email may have more than one categorization. We have designed system for categorization of emails using NetBeans IDE. The accuracy of the algorithm is calculated by testing both the algorithm with no. of content inputs. The Accuracy of CWS is 60% whereas the accuracy of LDA is 90%. The sensitivity of the algorithm is calculated by testing both the algorithm with no. of content inputs. The sensitivity of CWS is 100% whereas the sensitivity of LDA is 100%. The specificity of the algorithm is calculated by testing both the algorithm with no. of content inputs. The specificity of CWS is 0% whereas the specificity of LDA is 0%. The Precision of the algorithm is calculated by testing both the algorithm with no. of content inputs. The Precision of CWS is 60% whereas the Precision of LDA is 90%. The processing time of CWS is 1800 milliseconds whereas the processing time of LDA is 1500 milliseconds.

***Keyword:*EmailsSummarization, Clue Word Summarizer, Latent Dirichlet Allocation, and Natural Language Processing.**

I.INTRODUCTION

Before the invention of the Internet and the creation of the Web, the vast majority of human conversations were in spoken form, with the only notable, but extremely limited, exception being epistolary exchanges. Some important spoken conversations, such as criminal trials and political debates (e.g., Hansard, the transcripts of parliamentary debates), have been transcribed for centuries, but the rest of what humans have been saying to each other, throughout their history, to solve problems, make decisions and more generally to interact socially, has been lost.

This situation has dramatically changed in the last two decades. At an accelerating pace, people are having conversations by writing in a growing number of social media, including emails, blogs, chats and texting on mobile phones. At the same time, the recent, rapid progress in speech recognition technology is enabling the development of computer systems that can automatically transcribe any spoken conversation.

The net result of this ongoing revolution is that an ever-increasing portion of human conversations can be stored as text in computer memory and processed by applying Natural Language Processing (NLP) techniques (originally developed for written monologues - e.g., newspapers, books). This ability opens up a large space of extremely useful applications, in which critical information can be mined from conversations, and summaries of those conversations can be effectively generated.

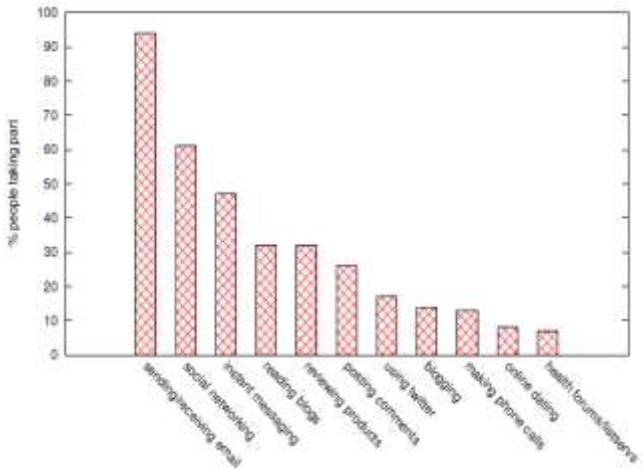

Fig.1: Popularity of various online conversational activities.

This is true for both organizations and individuals. For instance, managers can find the information exchanged in conversations within a company to be extremely valuable for decision auditing. If a decision turns out to be ill-advised, mining and summarizing the relevant conversations may help in determining responsibility and accountability. Similarly, conversations that led to favorable decisions could be mined and summarized to identify effective communication patterns and sources within the company. On a more personal level, an informative summary of a conversation could play at least two critical roles. On the one hand, the summary could greatly support a new participant to get up to speed and join an already existing, possibly long, conversation (e.g., blog comments). On the other hand, a summary could help someone to quickly prepare for a follow-up discussion of a conversation she was already part of, but which occurred too long ago for her to remember the details. Furthermore, the ability to summarize conversations will also be crucial in our increasingly mobile world, as a long incoming message or extensive ongoing conversations could be much more easily inspected on a small screen in a concise, summarized form.

Summarization

Automated summarization is the method of lowering a textual content file with PC software with the intention to create a summary that keeps the most vital points of the authentic document. Technologies that may make a coherent summary keep in mind variables inclusive of the period, writing style and syntax. Automated statistics summarization is part of machine mastering and records mining.

The primary concept of summarization is to discover a consultant subset of the facts, which includes the facts of the entire set. Summarization technology is utilized in a big quantity of sectors in the industry these days. An instance of the use of summarization generation is search engines like Google and Yahoo such as Google. Different examples consist of report summarization, photograph collection summarization, and video summarization. Document summarization attempts to routinely create a consultant précis or abstract of the whole document, with the aid of finding the maximum informative sentences. Similarly, in photograph summarization, the device unearths the maximum representative and crucial (or salient) photos. Similarly, in patron motion pictures one could want to get rid of the uninteresting or repetitive scenes and extract out miles shorter and concise model of the video. That is additionally critical, say for surveillance films, wherein one might need to extract most effective essential occasions inside the recorded video, due to the fact that maximum part of the video can be uninteresting with not anything happening. As the trouble of statistics overload grows, and as the quantity of information will increase, the interest in automatic summarization is also increasing.

Usually, there are processes to computerized summarization: extraction and abstraction. Extractive methods work through choosing a subset of current phrases, terms, or sentences inside the authentic text to shape the summary. In contrast, abstractive methods construct an internal semantic illustration after which uses herbal language generation techniques to create a summary that is what a human might generate. Such a summary might comprise phrases not explicitly gift in the original. Studies into abstractive techniques are an increasingly essential and active research region, however, because of complexity constraints, studies up to now have targeted usually on extractive techniques. In a few utility domain names, extractive summarization makes more experience. Examples of these consist of photo collection summarization and video summarization.

Extraction-based summarization

On this summarization assignment, the automated gadget extracts items from the entire series, without enhancing the objects themselves. Examples of this include key phrase extraction, where the aim is to pick out character phrases or terms to "tag" a report, and document summarization, in which the intention is to choose whole sentences (without editing them) to create a short paragraph précis. Similarly, in image collection summarization, the machine extracts photos from the collection without enhancing the photographs themselves.Extraction based summarization is process of subset the document data.

Extractive summaries present a number of advantages, such as:

- An extractive process is more lightweight than an intelligent procedure of summary composition. The extractive technique translates into a reduced computation time or processing time.
- By using entire parts of the text included in the original email, it is impossible to compose new phrases with incorrect synonyms. Even if the flow between parts might result shaky, the internal meaning of every single part remains the same.
- When users read some text in the summary, they can easily link it back to the original email if needed. On the contrary, tracing back a topic from an abstractive summary to the original email requires more time.
- Extractive summarization process is easy because there is no automatic generation of summary.

Abstraction-based summarization

Extraction techniques merely copy the records deemed most critical by means of the device to the précis (for instance, key clauses, sentences or paragraphs), at the same time as abstraction includes paraphrasing sections of the source file. In preferred, abstraction can condense a text more strongly than extraction, however, the applications that may do that are tougher to develop as they require the use of natural language technology era, which itself is a growing discipline.

Whilst a few work has been achieved in abstractive summarization (developing an abstract synopsis like that of a human), most of the people of summarization systems are extractive (selecting a subset of sentences to the region in a précis) [9].

Applications of Summarization

There are widely two sorts of extractive summarization duties depending on what the summarization application specializes in. The first is common summarization, which focuses on acquiring a frequent summary or summary of the collection (whether or not documents, or units of pix, or videos, news testimonies etc.). The second is question applicable summarization, now and again known as question-based summarization, which summarizes objects precise to a query. Summarizations structures are capable of creating each question applicable textual content summaries and common machine-generated summaries depending on what the user desires.

An instance of a summarization hassle is reported summarization, which tries to automatically produce an abstract from a given record. Once in a while, one might be inquisitive about generating a summary from a single supply record, even as others can use a couple of supply files (as an instance, a cluster of articles at the same subject matter). This trouble is referred to as multi-document summarization. The related software is summarizing news articles. Believe a system, which robotically pulls collectively news articles on a given topic (from the web), and concisely represents the latest information as a summary.

Photograph collection summarization is some other application instance of computerized summarization. It is composed in deciding on a consultant set of photos from a bigger set of photos [10]. A précis on this context is beneficial to expose the most consultant pics of consequences in a photo collection exploration gadget. Video summarization is an associated area, wherein the device automatically creates a trailer of a protracted video. This also has applications in customer or personal movies, in which one would possibly need to, skip the dull or repetitive actions. In addition, in surveillance films, one would need to extract essential and suspicious hobby, even as ignoring all the uninteresting and redundant frames captured.

Clue word

A clue word in node (fragment) F is a word which also appears in a semantically similar form in a parent or a child node of F in the fragment quotation graph. Applying stemming to the identification of clue words, using Porter's stemming algorithm to compute the stem of each word, and use the stems to judge the reoccurrence.

Fragments (a) and (b) are two adjacent nodes with (b) as the parent node of (a).

Here observing 3 major kinds of reoccurrence : the same root (stem) with different forms, e.g., "settle" vs. "settlement" and "discuss" vs. "discussed" as in the example above. Synonyms/antonyms or words with similar/contrary meaning, e.g., "talk" vs. "discuss" and "peace" vs. "war". Words that have a looser semantic link, e.g., "deadline" with "Friday morning".

Algorithm CWS : Algorithm Clue Word Summarizer (CWS) uses clue words as the main feature for summarization. The assumption is that if those words reoccur between parent and child nodes, they are more likely to be relevant and important to the conversation.

The system consists of four part, each part with its own working. These parts are as follow:

1. Main Function
2. Tokenizer
3. Summarizer
4. Print Summary

Latent DirichletAlllocation (LDA)

Latent DirichletAlllocation (LDA) [12] is an algorithm that specifically aims to find these short descriptionsfor members in a data collection. Originally proposed in the context of text document modeling, LDA positsthat one way of summarizing the content of a document quickly is to look at the set of words it uses. Becausewords carry very strong semantic information, documents that contain similar content will most likely usea similar set of words. As such, mining an entire corpus of text documents can expose sets of words thatfrequently co-occur within documents. These sets of words may be intuitively interpreted as topics and actas the building blocks of the short descriptions.

More formally, LDA is a probabilistic, generative model for discovering latent semantic topics in largecollections of text data. Each discovered topic is characterized by its own particular distribution over words.Each document is then characterized as a random mixture of topics indicating the proportion of time thedocument spends on each topic. This random mixture of topics is essentially our "short description": Itnot only expresses the semantic content of a document in a concise manner, but also gives us a principledapproach for describing documents quantitatively. We can now compare how similar one document is toanother by looking at how similar the corresponding topic mixtures are.

Though originally proposed for text documents, LDA exists as a very general topic discovering framework.In recent years, the model has been extended to numerous applications in other domains, including: objectrecognition [13, 14, 15, 16, 17], natural language processing [18, 19], video analysis [20, 21], collaborative filtering[22], spam filtering [23], web-mining [24], authorship disambiguation [25], and dialogue segmentation [26].

II.LITERATURE REVIEW

RupalBhargava et.al in [1] proposed a technique utilizing which one can break down various dialects to discover assumptions in them and perform notion examination. The strategy use diverse systems of machine figuring out how to examine the content. Machine interpretation is utilized as a part of the framework to give the component of managing diverse dialects. After the machine interpretation, the content is prepared for finding the assumptions in the content. With the coming of websites, gatherings, and online audits there is significant content present on the web that can be utilized to investigate the assessment about a specific subject or a question. Thus to diminish the preparing it is advantageous to separate the imperative content present in it. So the framework proposed utilizes content outline procedure to remove vital parts of the content and after that utilizations it to examine the slants about the specific subject and its angles.

ShaganSah et.al in [2] proposed novel systems for abridging and clarifying long recordings. Existing video synopsis methods concentrate solely on distinguishing key edges and sub shots, however assessing these condensed recordings is a testing assignment. Their work proposes strategies to create visual rundowns of long recordings, and furthermore, proposes systems to comment on and produce literary synopses of the recordings utilizing intermittent systems. Fascinating fragments of the long video are extricated in light of picture quality and also cinematographic and purchaser inclination. Key casings from the most impactful fragments are changed over to literary explanations utilizing successive encoding and disentangling profound learning models. Their synopsis procedure is benchmarked on the Video Set dataset and assessed by people for instructive and etymological substance. They trusted this to be the primary completely programmed strategy equipped for synchronous visual and literary synopsis of long buyer recordings.

Dan Cao et.al in [3] reviewed on every one of the components that utilization measurements and idea of the perplexing system for scoring sentences. The analysis comes about on single component and mixes of different elements they proposed are talked about. Quantitative and subjective perspectives were considered in their appraisal performing on the DUC 2002 informational collections.

SumyaAkter et.al in [4] presented a strategy for content synopsis which separates essential sentences from a solitary or numerous Bengali records. The information document ought to be pre-handled by tokenization, stemming operation and so on. At that point, word score is figured by Term-Frequency/Inverse Document Frequency and sentence score is dictated by summing up its constituent words' scores with its position. Signal and skeleton words have additionally been considered to ascertain the sentence score. For single or numerous records, K-implies grouping calculation has been connected to create the last synopsis. The outcome demonstrates agreeable yields in contrast with the current methodology having straight run time quality. The result shows outputs in comparison to the approaches possessing linear run time complexity.

Taeho Jo et.al in [5] proposed a specific adaptation of KNN (K Nearest Neighbor) where the likeness between highlight vectors is figured considering the closeness among properties or elements and also one among values.

The errand of content outline is seen as the parallel arrangement undertaking where each passage or sentence is ordered into the pith or non-quintessence, and in past works, enhanced outcomes are acquired by the proposed form in the content characterization and bunching. In this examination, they characterized the similitude which considers the two properties and qualities esteems, adjusts the KNN into the form in view of the comparability, and utilizations the altered form as the way to deal with the content synopsis errand. As the advantages from this exploration, they may expect the more minimized portrayal of information things and the better execution. Along these lines, the objective of this examination is to actualize the content synopsis calculation which speaks to information things all the more minimalists and gives the greater unwavering quality.

P Krishnaprasadet.al in [6] proposed strategy uses the sentence extraction in a solitary record and creates a nonexclusive rundown for a given Malayalam archive (Extractive synopsis). Sentences in the archive are positioned in view of the word score of each word exhibit in it. Top N positioned sentences are removed and orchestrate them in their sequential request for a rundown era, where N speaks to the measure of the outline as for the level of unique report estimate. The standard metric ROUGE is utilized for execution assessment. ROUGE computes the n-gram cover between a created synopsis and reference outlines. Reference outlines were built physically. Examinations demonstrate that the outcomes are promising.

Md. MajharulHaqueet.al in [7] presented an approach of programmed Bangla content rundown by improving a current key expression based strategy. The improvement is refined with three stages as tails: (i) adjusting the key expressions determination process, (ii) incorporating the principal sentence in outline in the event that it contains any title word and (iii) tallying numerical figure which is introduced in digits and words for sentence scoring. Well-ordered execution investigation of their proposed approach is depicted for two datasets. Execution is measured with ROUGE (Recall Oriented Understudy for Gisting Evaluation) programmed assessment bundle.

JianweiNiuet.al in [8] proposed OnSeS, a novel short content rundown technique which makes full utilization of word2vec to speak to a word and uses neural system model to produce each expression of the synopsis. OnSeS comprises of three expressions: 1) grouping short messages utilizing the K-implies calculation; 2) positioning substance of each bunch by building a chart based positioning model utilizing BM25; 3) producing the principle purposes of each bunch with the assistance of neural machine interpretation display on the best positioned sentence. The exploratory outcomes uncover that they're proposed completely information driven approach beats best in class technique.

III.PROBLEM FORMULATION

With the ever increasing popularity of emails, it is very common nowadays that people discuss specific issues, events among a group of people by emails. The conversations via emails are valuable for the user as a personal information repository. But accessing of the increasing no. of emails is major problem. In the base paper, they adopted three cohesion metrics, clue words, semantic similarity and cosine similarity, to measure the weight of the edges. Moreover, the study how to include subjective opinions to help identify important sentences for

summarization.By the use of CWS accuracy of the system is not so good which lies between 40 to 60%. The need of algorithm is felt in which the accuracy could be maximum from the previous algorithm, therefore, we use another algorithm using latent dirichlet allocation for Summarizing Emails.

IV.PROPOSED RESEARCH METHODOLOGY

Flowchart of CWS algorithm:

Step 1: Select the content for summarizer. The contect is collect from email, paragraph, document and articles. CWS is work on large text or data for the summarization purpose.

Step 2: Decide the number of words in the content related to the topic in CWS. Words that are related to the keywords that are comes in the particular category are consider for the topic.

Step 3: Choose the Topic distribution is based on the keywords of the particular category. For example if sentence is related to the animal category and its keywords are Lion,Tiger etc.

Step 4: The email content that is related to the particular category e.g. Fruits, is entered, then it start searching the keywords i.e apple, mango, orange that match with the Fruits category.

Step 5: CWS algorithm check a matching of individual keywords with topic. If it finds means keyword is similar to some part of topic.

Step 6: Now, CWS algorithm find the matching topic of particular category with different no. of keywords. For example in the fruit category the keywords are Apple, Orange, Mango etc.

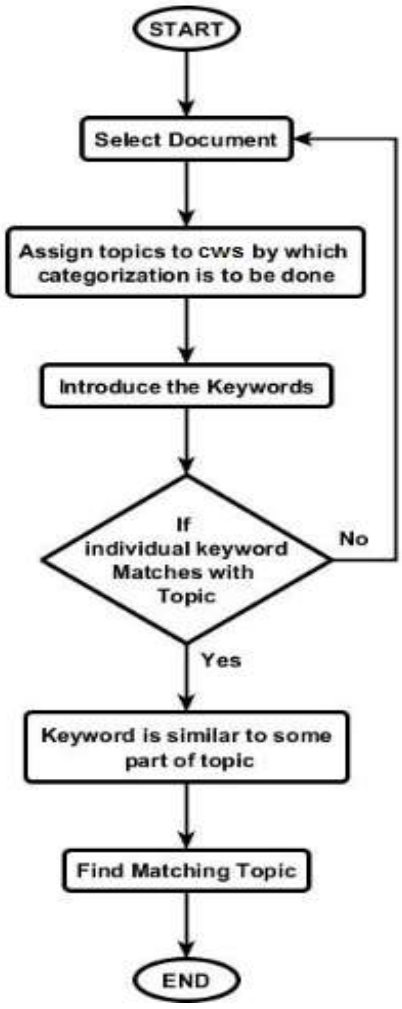

Fig. 2: Flow Chart of CWS Algorithm

Flowchart of LDA algorithm:

Step 1: Select the content for summarizer. The contect is collect from email, paragraph, document and articles. In LDA we consider small text or content of data for the summarization purpose.

Step 2: Decide the number of words in the content related to the topic. Words that are related to the keywords that are comes in the particular category are consider for the topic.

Step 3: Choose the Topic distribution is based on the keywords of the particular category. For example if sentence is related to the animal category and its keywords are Lion,Tiger etc.

Step 4: If email content that is related to the one category e.g. Fruits, is entered, then it start searching the keywords i.e apple, mango, orange that match with the Fruits category.

Step 5: LDA algorithm check a matching of keywords set with topic. If it finds means keyword is similar to exact phrase in the text.

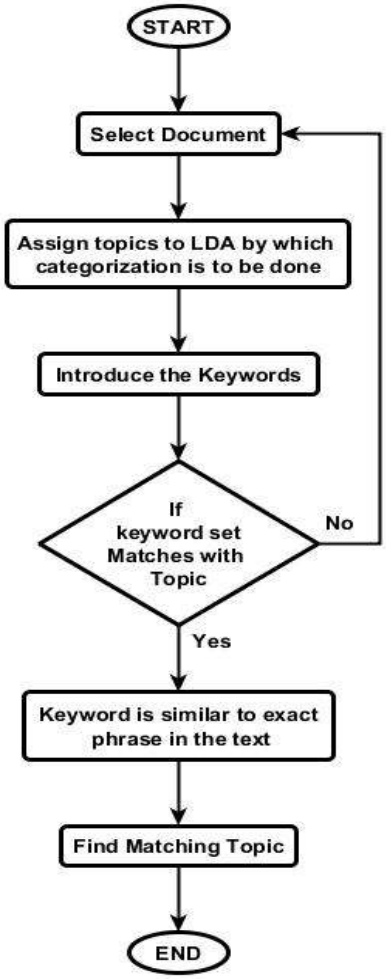

Fig. 3: Flowchart of LDA Algorithm

Step 6: LDA algorithm find the matching topic of different category with keywords set. These keywords set may be belong to more than one category. For example, keyword set contain apple and lion keywords, then it belong to two categories fruits and animals.

Research Methodology:

Step 1: Select the content for summarizer.

Step 2: To Implement the Clue Word Summarizer (CWS) Algorithm in NetBeans IDE.

Step 3: Calculate the result of the CWS Algorithm with Accuracy, Sensitivity, Specificity and Precision.

Step 4: Implement the Latent Dirichlet Allocation (LDA) algorithm in NetBeans IDE.

Step 5: Calculate the result of the LDA Algorithm with Accuracy, Sensitivity, Specificity and Precision.

Step 6: Comparision of LDA Accuracy with CWS Accuracy.

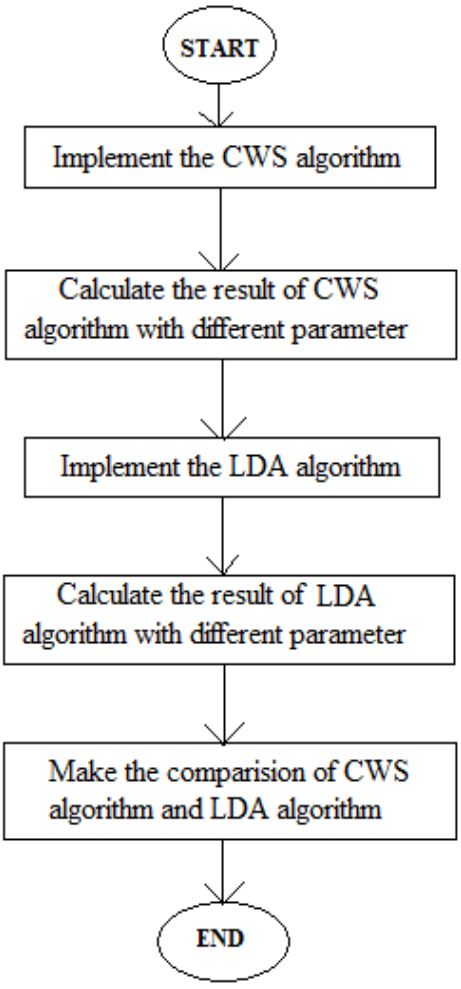

Fig. 4: Research Methodology

V.EXPERIMENTAL PROCEDURES

We have proposed Latent Dirichlet allocation (LDA) for email summarization, which will help to remove flaws of Clue Word Summarization (CWS) algorithm. In CWS the emails are marked with only one category whereas according to LDA one email may not be categorized into one category only. The email may have more than one categorization.

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.06, April 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

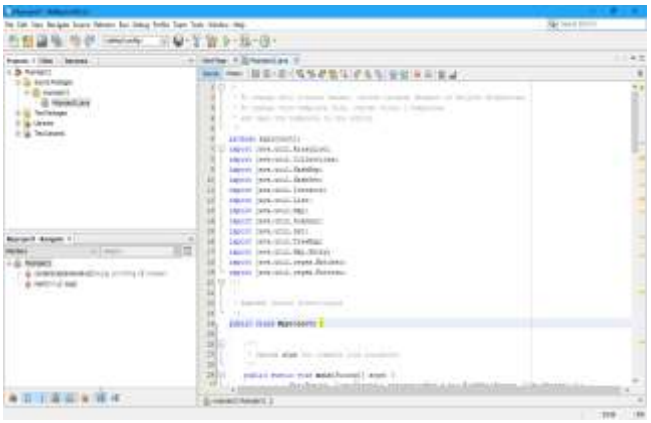

Fig. 5: CWS Code developed using NetBeans IDE

The basic interface of NetBeans and Code developed is shown in figure 5. Here we code for detection of two categories of emails named as Animals and Fruits which are having quantities like Lion, Tiger, Elephant, Mango, Apple, and Orange. All these quantities are case sensitive.

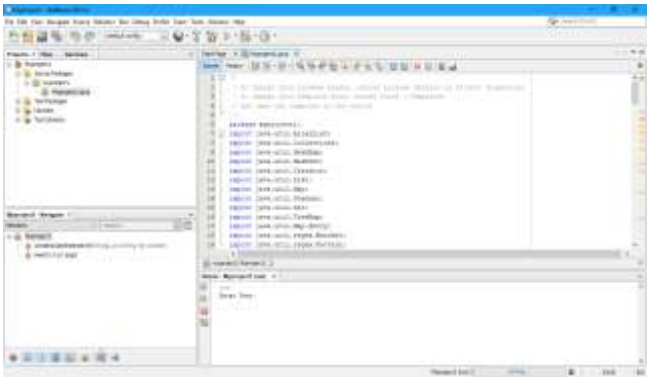

Fig. 6: Output Windowfor CWS

The output window pops below when we run the code is shown in figure 6. The output window shows message "Enter Text:" Where we have to enter text for categorization of Email Contents.

Fig. 7: Output Window for Fruits Category

The output window for fruits category is shown in figure 7. When we run the code and output window will be appear on the screen and user enter the text i.e Enters Text: “In most case the common fruits are Apple and Orange”. The Result shows that “Paragraph is of following Category: Fruits”. At the end of output results, it also shows the time taken for providing output results. These Results are for CWS algorithm and same results will come for LDA algorithm.

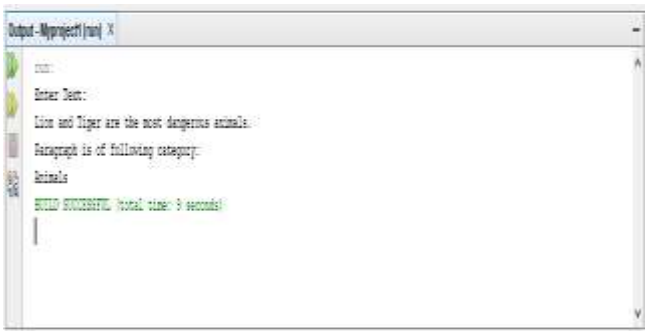

Fig. 8: Output Window for Animals Category

The output window for animal category is shown in figure 8. When we run the code then this output window will be shown on the screen at the below of NetBeans Screen. After that Enter Text option will be appear in the output window. Its user enter the text manually likewise Enters Text: “Lion and Tiger are the most dangerous animals”. The Result shows that “Paragraph is of following Category: Animals”. These Results are for CWS algorithm and same results will come for LDA algorithm.

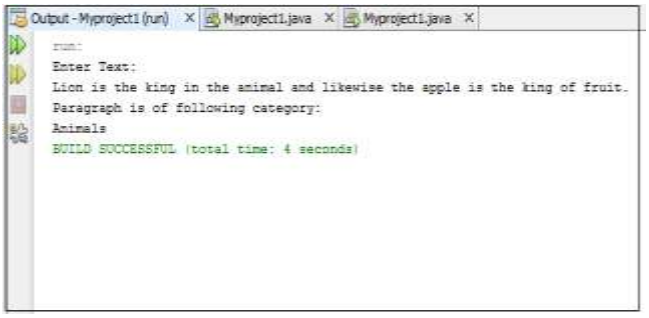

Fig. 9: Output Window for animals and Fruits Category for CWS

The output window for animals and fruits category is shown in figure 9. When we run the code and Enters Text: “Lion is the king in the animals and likewise the Apple is the king of fruits”. The Result shows that “Paragraph is of following Category: Animals”. Here CWS algorithm is detect only one category i.e. Animals but 2nd category i.e. Fruits is not detect by the CWS algorithm. At the end of output results, it also shows the time taken for providing output results.

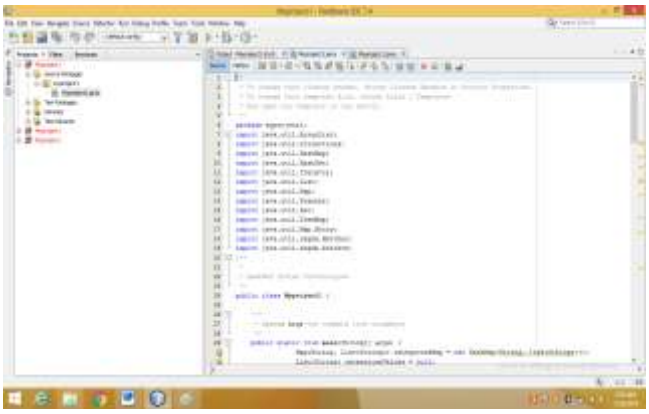

Fig. 10: LDA Code run using NetBeans IDE

The LDA Code using NetBeans IDE shown in figure 10. Here we code for detection of two categories of emails named as Animals and Fruits which are having quantities like Lion, Tiger, Elephant, Mango, Apple, and Orange. All these quantities are case sensitive.

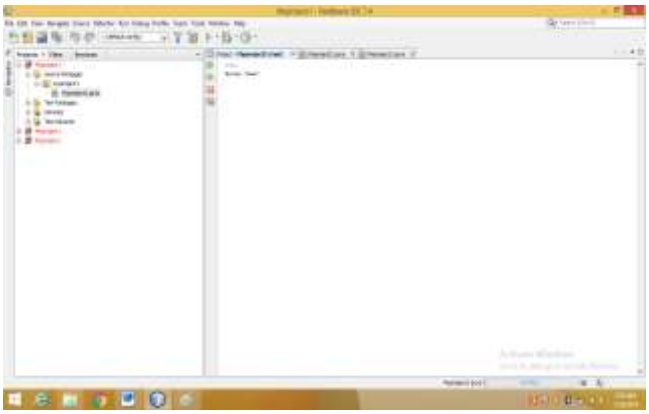

Fig. 11: Output Window for LDA

The output window pops below when we run the code is shown in figure 11. The output window shows message "Enter Text:" Here we enter text for categorization of Email Contents.

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.06, April 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

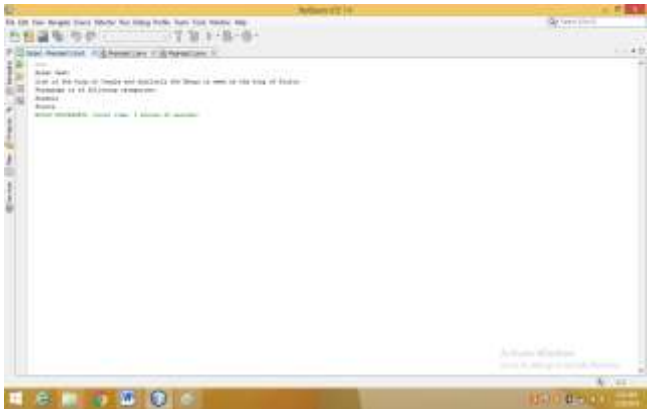

Fig. 12: Output Window for animals and Fruits Category for LDA

The output window for animals and fruits category is shown in figure 12. When we run the code and Enters Text: "Lion is the king in the animals and likewise the Apple is the king of fruits". The Result shows that "Paragraph is of following Category: Animals Fruits". Here LDA algorithm is detecting both categories i.e. Animals and Fruits at a one time. At the end of output results, it also shows the time taken for providing output results.

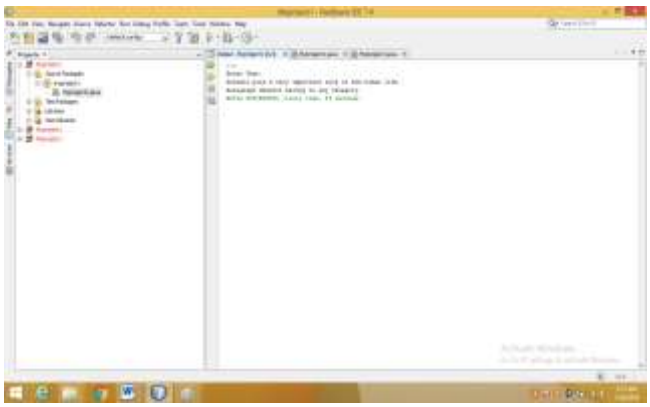

Fig. 13: Output window for No Category

The output window for no category is shown in figure 13. When we run the code and Enters Text: "Animals play a very important role in the human life". The Result shows that "Paragraph does not belong to any Category". At the end of output results, it also shows the time taken for providing output results. There are no clue words which are meant for categorization, therefore, this content does not belong to any Category.

Calculations with different Primary Function

Email summarization is depending on different types of primary function. In this analysis, email summarization is done with both algorithm CWS and LDA with the help of 4 primary functions.

- **number of positive samples (P)**
- **number of negative samples (N)**
- **number of true positive (TP)**
- **number of true negative (TN)**
- **number of false positive (FP)**
- **number of false negative (FN)**

TABLE I. Accurate Calculation table for CWS

Primary Function	No. of Sentences			
	10	20	30	40
TP	6	12	17	23
TN	0	0	0	0
FP	4	8	13	17
FN	0	0	0	0

TABLE II. Accurate Calculation table for LDA

Primary Function	No. of Sentences			
	10	20	30	40
TP	9	18	27	36
TN	0	0	0	0
FP	1	2	3	4
FN	0	0	0	0

Parameter for Comparison

1. Accuracy
2. Sensitivity
3. Specificity
4. Precision

1. Accuracy (ACC):

$$ACC = \frac{(TP + TN)}{(TP + FP + TN + FN)}$$

For CWS:

Upto 10 sentences (CWS)

$$\text{ACC} = \frac{6+0}{(6+4+0+0)} = \frac{6}{(10)} = 0.6\% = 60\%$$

Upto 20 sentences (CWS)

$$\text{ACC} = \frac{12+0}{(12+8+0+0)} = \frac{12}{(20)} = 0.6\% = 60\%$$

Upto 30 sentences (CWS)

$$\text{ACC} = \frac{17+0}{(17+13+0+0)} = \frac{17}{(30)} = 0.56\% = 56\%$$

Upto 40 sentences (CWS)

$$\text{ACC} = \frac{23+0}{(23+17+0+0)} = \frac{23}{(40)} = 0.57\% = 57\%$$

For LDA:

Upto 10 sentences (LDA)

$$\text{ACC} = \frac{9+0}{(9+1+0+0)} = \frac{9}{(10)} = 0.9\% = 90\%$$

Upto 20 sentences (LDA)

$$\text{ACC} = \frac{18+0}{(18+2+0+0)} = \frac{18}{(20)} = 0.9\% = 90\%$$

Upto 30 sentences (LDA)

$$\text{ACC} = \frac{27+0}{(27+3+0+0)} = \frac{27}{(30)} = 0.9\% = 90\%$$

Upto 40 sentences (LDA)

$$\text{ACC} = \frac{36+0}{(36+4+0+0)} = \frac{36}{(40)} = 0.9\% = 90\%$$

TABLE III. Comparison of Accuracy for CWS and LDA algorithm

Accuracy	Upto 10 Sentences	Upto 20 Sentences	Upto 30 Sentences	Upto 40 Sentences
CWS Algorithm	60	60	56	57
LDA Algorithm	90	90	90	90

2.Sensitivity or true positive rate (TPR)

$$TPR = \frac{TP}{P} = \frac{TP}{(TP + FN)}$$

For CWS:

Upto 10 sentences (CWS)

$$\textbf{TPR} = \frac{6}{(6+0)} = \frac{6}{(6)} = 100\%$$

Upto 20 sentences (CWS)

$$\textbf{TPR} = \frac{12}{(12+0)} = \frac{12}{(12)} = 100\%$$

Upto 30 sentences (CWS)

$$\textbf{TPR} = \frac{17}{(17+0)} = \frac{17}{(17)} = 100\%$$

Upto 40 sentences (CWS)

$$\textbf{TPR} = \frac{23}{(23+0)} = \frac{23}{(23)} = 100\%$$

For LDA:

Upto 10 sentences (LDA)

$$\textbf{TPR} = \frac{9}{(9+0)} = \frac{9}{(9)} = 100\%$$

Upto 20 sentences (LDA)

$$\textbf{TPR} = \frac{18}{(18+0)} = \frac{18}{(18)} = 100\%$$

Upto 30 sentences (LDA)

$$\textbf{TPR} = \frac{27}{(27+0)} = \frac{27}{(27)} = 100\%$$

Upto 40 sentences (LDA)

$$\textbf{TPR} = \frac{36}{(36+0)} = \frac{36}{(36)} = 100\%$$

TABLE IV. Comparison of Sensitivity for CWS and LDA algorithm

Sensitivity	Upto 10 Sentences	Upto 20 Sentences	Upto 30 Sentences	Upto 40 Sentences
CWS Algorithm	100	100	100	100
LDA Algorithm	100	100	100	100

3. Specificity (SPC) or true negative rate

$$SPC = \frac{TN}{N} = \frac{TN}{(TN + FP)}$$

For CWS:

Upto 10 sentences (CWS)

$$\text{SPC} = \frac{0}{(0+4)} = \frac{0}{(4)} = 0\%$$

Upto 20 sentences (CWS)

$$\text{SPC} = \frac{0}{(0+8)} = \frac{0}{(8)} = 0\%$$

Upto 30 sentences (CWS)

$$\text{SPC} = \frac{0}{(0+13)} = \frac{0}{(13)} = 0\%$$

Upto 40 sentences (CWS)

$$\text{SPC} = \frac{0}{(0+17)} = \frac{0}{(17)} = 0\%$$

For LDA:

Upto 10 sentences (LDA)

$$\text{SPC} = \frac{0}{(0+1)} = \frac{0}{(1)} = 0\%$$

Upto 20 sentences (LDA)

$$\text{SPC} = \frac{0}{(0+2)} = \frac{0}{(2)} = 0\%$$

Upto 30 sentences (LDA)

$$\textbf{SPC} = \frac{0}{(0+3)} = \frac{0}{(3)} = 0\%$$

Upto 40 sentences (LDA)

$$\textbf{SPC} = \frac{0}{(0+4)} = \frac{0}{(4)} = 0\%$$

TABLE V.Comparison of Specificity for CWS and LDA algorithm

Specificity	Upto 10 Sentences	Upto 20 Sentences	Upto 30 Sentences	Upto 40 Sentences
CWS Algorithm	0	0	0	0
LDA Algorithm	0	0	0	0

4. Precision or positive predictive value (PPV)

$$PPV = \frac{TP}{(TP + FP)}$$

For CWS:

Upto 10 sentences (CWS)

$$\textbf{PPV} = \frac{6}{(6+4)} = \frac{6}{(10)} = 0.6\% \text{ or } 60\%$$

Upto 20 sentences (CWS)

$$\textbf{PPV} = \frac{12}{(12+8)} = \frac{12}{(20)} = 0.6\% \text{ or } 60\%$$

Upto 30 sentences (CWS)

$$\textbf{PPV} = \frac{17}{(17+13)} = \frac{17}{(30)} = 0.56\% \text{ or } 56\%$$

Upto 40 sentences (CWS)

$$\textbf{PPV} = \frac{23}{(23+17)} = \frac{23}{(40)} = 0.57\% \text{ or } 57\%$$

For LDA:

Upto 10 sentences (LDA)

$$\textbf{PPV} = \frac{9}{(9+1)} = \frac{9}{(10)} = 0.9\% \text{ or } 90\%$$

Upto 20 sentences (LDA)

$$\textbf{PPV} = \frac{18}{(18+2)} = \frac{18}{(20)} = 0.9\% \text{ or } 90\%$$

Upto 30 sentences (LDA)

$$\textbf{PPV} = \frac{27}{(27+3)} = \frac{27}{(30)} = 0.9\% \text{ or } 90\%$$

Upto 40 sentences (LDA)

$$\textbf{PPV} = \frac{36}{(36+4)} = \frac{36}{(40)} = 0.9\% \text{ or } 90\%$$

TABLE VI. Comparison of Precision for CWS and LDA algorithm

Precision	Upto 10 Sentences	Upto 20 Sentences	Upto 30 Sentences	Upto 40 Sentences
CWS Algorithm	60	60	56	57
LDA Algorithm	90	90	90	90

VI.RESULTS AND DISCUSSIONS

Graph in the term of Accuracy

Accuracy refers to the closeness of the measured value to a standard. It is defines as correctness of the measured values. The different range of accuracy is defined in the Table III. Here accuracy is checked with different no. of the sentences.

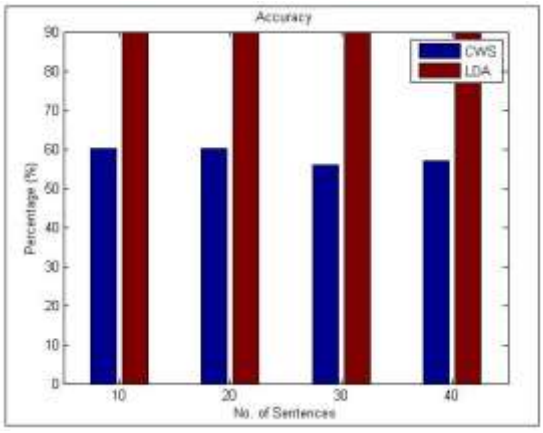

Fig. 14: Comparison of Accuracy for CWS and LDA algorithm

Blue colour depicts the Accuracy of CWS algorithm in the Graph and Red colour depicts the Accuracy of LDA algorithm. At the horizontal axis no. of inputs is shown and at the vertical axis percentage is shown in figure 14. The accuracy is calculated by testing both the algorithm with no. of content inputs. The Accuracy of CWS is 60% whereas the accuracy of LDA is 90%.

Graph in the term of Sensitivity

Sensitivity is the statistical measure of the performance. Sensitivity is also called True Positive Rate or probability of detection e.g. Percentage of sick people who are correctly identified as having the condition. The different range of sensitivity is defined in the Table IV. Here sensitivity is checked with different no. of the sentences.

Blue colour depicts the Sensitivity of CWS algorithm in the Graph and Red colour depicts the Sensitivity of LDA algorithm. At the horizontal axis no. of inputs is shown and at the vertical axis percentage is shown in figure 15. The Sensitivity is calculated by testing both the algorithm with no. of content inputs. The Sensitivity of CWS is 100% whereas the accuracy of LDA is 100%.

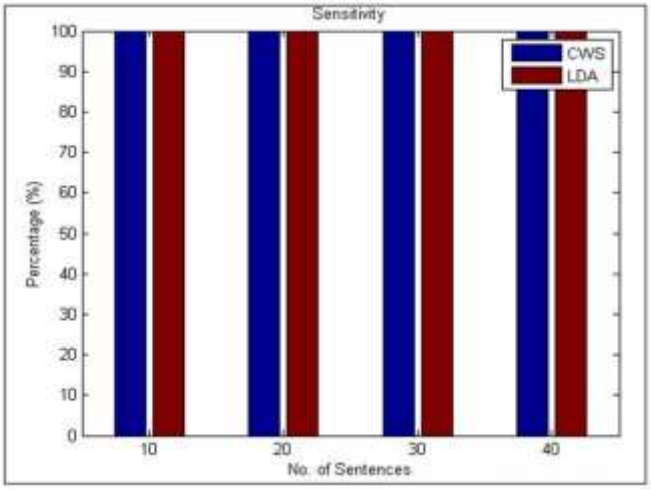

Fig. 15: Comparison of Sensitivity for CWS and LDA algorithm

Graph in the term of Specificity

Specificity is defined as a True Negative Rate. Specificity measures the proportion of negative that are correctly identified as such e.g. Percentage of healthy people who are correctly identified as not having the condition. Specificity is also be defined as the quality or state of being specific. The different range of Specificity is defined in the Table V. Here Specificity is checked with different no. of the sentences.

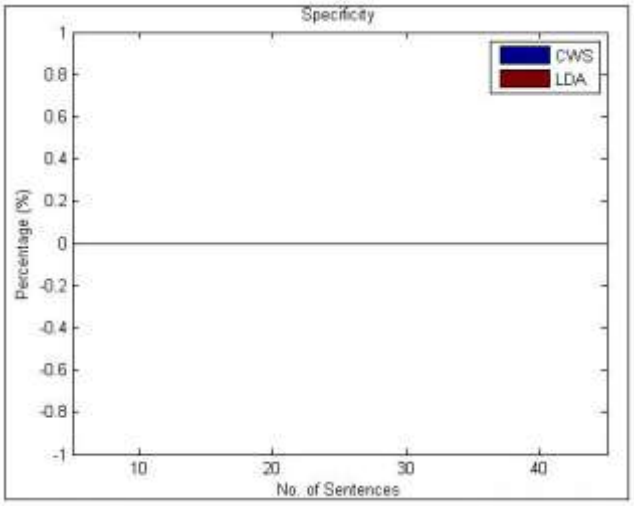

Fig. 16: Comparison of Specificity for CWS and LDA algorithm

Blue colour depicts the Specificity of CWS algorithm in the Graph and Red colour depicts the Specificity of LDA algorithm. At the horizontal axis no. of inputs is shown and at the vertical axis percentage is shown in figure 16. The Specificity is calculated by testing both the algorithm with no. of content inputs. The Accuracy of CWS is 0% whereas the accuracy of LDA is 0%.

Graph in the term of Precision

Precision is fraction of the documents retrieved that are relevant to the user's information need. Precision is also called Positive Predictive values. It takes all retrieved documents into account. It can also be evaluated at a given cut-off-rank, considering only the topmost results returned by the system. The measure is called precision at N (P@N). The different range of Precision is defined in the Table VI. Here Precision is checked with different no. of the sentences or different categories.

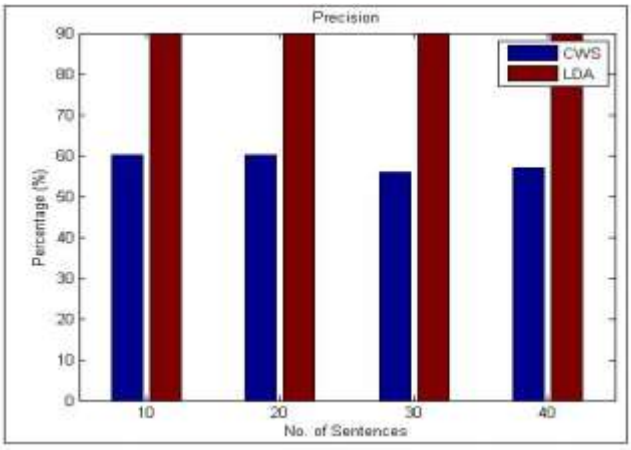

Fig. 17: Comparison of Precision for CWS and LDA algorithm

Blue colour depicts the Precision of CWS algorithm in the Graph and Red colour depicts the Precision of LDA algorithm. At the horizontal axis no. of inputs is shown and at the vertical axis percentage is shown in figure 17. The Precision is calculated by testing both the algorithm with no. of content inputs. The Precision of CWS is 60% whereas the Precision of LDA is 90%.

Graph in the term of Time

Time is calculated by actually time taken for completing the process. The processing time of the algorithm is calculated by testing both the CWS algorithm and LDA algorithm with no. of content inputs.

At the horizontal axis both CWS and LDA algorithm is shown and at the vertical axis processing time in millisecond is shown in figure 18. The processing time of CWS algorithm is 1800 milliseconds whereas the processing time of LDA algorithm is 1500 milliseconds.

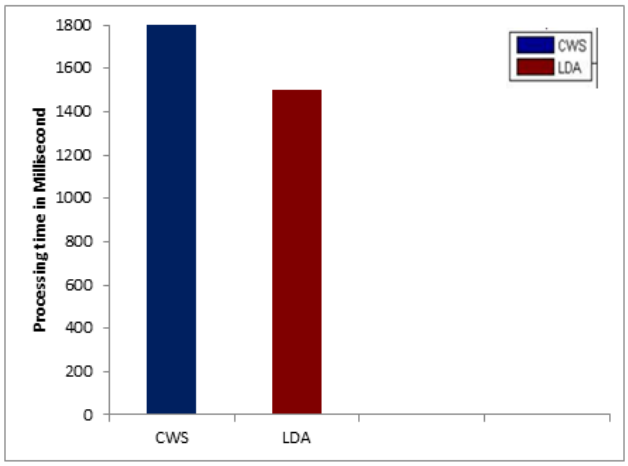

Fig. 18: Comparison of Processing time for CWS and LDA Algorithm

VII.CONCLUSION AND FUTURE SCOPE

Summarizing email conversations is challenging due to the characteristics of emails, especially the conversational nature. Most of the existing methods dealing with email conversations use the email thread to represent the email conversation structure, which is not accurate in many cases [11]. We are presenting an approach which will help to remove flaws of Clue Word Summarization (CWS) algorithm. In CWS the emails are marked with only one category whereas according to LDA one email may not be categorized into one category only. The email may have more than one categorization. We have designed system for categorization of emails using NetBeans IDE. The accuracy of the algorithm is calculated by testing both the algorithm with no. of content inputs. The Accuracy of CWS is 60% whereas the accuracy of LDA is 90%.The sensitivity of the algorithm is calculated by testing both the algorithm with no. of content inputs. The sensitivity of CWS is 100%

whereas the sensitivity of LDA is 100%.The specificity of the algorithm is calculated by testing both the algorithm with no. of content inputs. The specificity of CWS is 0% whereas the specificity of LDA is 0%.The Precision of the algorithm is calculated by testing both the algorithm with no. of content inputs. The Precision of CWS is 60% whereas the Precision of LDA is 90%.The processing timeof CWS is 1800 milliseconds whereas the processing time of LDA is 1500 milliseconds.

The future scope of the proposed algorithm is to mark the emails with specific category tags. In future, we also try to remove the limitation of thresholding in proposed algorithm.In future scope of LDA algorithm is improvement of sensitivity and specificity.

REFERENCES

[1] R. Bhargava and Y. Sharma, "MSATS: Multilingual sentiment analysis via text summarization," 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence, Noida, 2017, pp. 71-76.

[2] S. Sah, S. Kulhare, A. Gray, S. Venugopalan, E. Prud'Hommeaux and R. Ptucha, "Semantic Text Summarization of Long Videos," 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 2017, pp. 989-997.

[3] Dan Cao and LiutongXu, "Analysis of complex network methods for extractive automatic text summarization," 2016 2nd IEEE International Conference on Computer and Communications (ICCC), Chengdu, 2016, pp. 2749-2756.

[4] S. Akter, A. S. Asa, M. P. Uddin, M. D. Hossain, S. K. Roy and M. I. Afjal, "An extractive text summarization technique for Bengali document(s) using K-means clustering algorithm," 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Dhaka, 2017, pp. 1-6.

[5] T. Jo, "K nearest neighbor for text summarization using feature similarity," 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), Khartoum, 2017, pp. 1-5.

[6] P. Krishnaprasad, A. Sooryanarayanan and A. Ramanujan, "Malayalam text summarization: An extractive approach," 2016 International Conference on Next Generation Intelligent Systems (ICNGIS), Kottayam, 2016, pp. 1-4.

[7] M. M. Haque, S. Pervin and Z. Begum, "Enhancement of keyphrase-based approach of automatic Bangla text summarization," 2016 IEEE Region 10 Conference (TENCON), Singapore, 2016, pp. 42-46.

[8] J. Niu, Q. Zhao, L. Wang, H. Chen, M. Atiquzzaman and F. Peng, "OnSeS: A Novel Online Short Text Summarization Based on BM25 and Neural Network," 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, 2016, pp. 1-6.

[9] En.wikipedia.org. (2017). Automatic summarization. [online] Available at: https://en.wikipedia.org/wiki/Automatic_summarization [Accessed 13 Jun. 2017].

[10] Jorge E. Camargo and Fabio A. González. A Multi-class Kernel Alignment Method for Image Collection Summarization. In Proceedings of the 14th Iberoamerican Conference on Pattern Recognition: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications (CIARP '09), Eduardo Bayro-Corrochano and Jan-OlofEklundh (Eds.). Springer-Verlag, Berlin, Heidelberg, 545-552. doi:10.1007/978-3-642-10268-4_64.

[11] H. Aaron, Y. Jen-Yuan, "Email thread reassembly using similarity matching". In Proceedings of the Third Conference on Email and Anti- Spam (CEAS), 2006.

[12] Blei, D.M., Ng, A.Y. and Jordan, M.I., 2003. Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), pp.993-1022.

[13] Fei-Fei, L. and Perona, P., 2005, June. A bayesian hierarchical model for learning natural scene categories. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on (Vol. 2, pp. 524-531). IEEE.

[14] Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A. and Freeman, W.T., 2005, October. Discovering objects and their location in images. In Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on (Vol. 1, pp. 370-377). IEEE.

[15] Russell, B.C., Freeman, W.T., Efros, A.A., Sivic, J. and Zisserman, A., 2006. Using multiple segmentations to discover objects and their extent in image collections. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on (Vol. 2, pp. 1605-1614). IEEE.

[16] Cao, L. and Fei-Fei, L., 2007, October. Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on (pp. 1-8). IEEE.

[17] Wang, X. and Grimson, E., 2008. Spatial latent dirichlet allocation. In Advances in neural information processing systems (pp. 1577-1584).

[18] Griffiths, T.L., Steyvers, M., Blei, D.M. and Tenenbaum, J.B., 2005. Integrating topics and syntax. In Advances in neural information processing systems (pp. 537-544).

[19] Boyd-Graber, J.L. and Blei, D.M., 2009. Syntactic topic models. In Advances in neural information processing systems (pp. 185-192).

[20] Niebles, J.C., Wang, H. and Fei-Fei, L., 2008. Unsupervised learning of human action categories using spatial-temporal words. International journal of computer vision, 79(3), pp.299-318.

[21] Wang, X., Ma, X. and Grimson, E., 2007, June. Unsupervised activity perception by hierarchical bayesian models. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on (pp. 1-8). IEEE.

[22] Marlin, B.M., 2004. Modeling user rating profiles for collaborative filtering. In Advances in neural information processing systems (pp. 627-634).

[23] Bíró, I., Szabó, J. and Benczúr, A.A., 2008, April. Latent dirichlet allocation in web spam filtering. In Proceedings of the 4th international workshop on Adversarial information retrieval on the web (pp. 29-32). ACM.

[24] Mei, Q., Liu, C., Su, H. and Zhai, C., 2006, May. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In Proceedings of the 15th international conference on World Wide Web (pp. 533-542). ACM.

[25] Rosen-Zvi, M., Griffiths, T., Steyvers, M. and Smyth, P., 2004, July. The author-topic model for authors and documents. In Proceedings of the 20th conference on Uncertainty in artificial intelligence (pp. 487-494). AUAI Press.

[26] Purver, M., Griffiths, T.L., Körding, K.P. and Tenenbaum, J.B., 2006, July. Unsupervised topic modelling for multi-party spoken discourse. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics (pp. 17-24). Association for Computational Linguistics.