

# A Quantitative Analysis of Automatic Text Summarization (ATS)

Akshat Gupta, Dr. Ashutosh Singh, Dr. A. K Shankhwar  
Department of Electronics Engineering  
HBTU Kanpur-208002 India

**Abstract:-** In the last ten years, Automatic Text Summarization (ATS), which is extremely widely known, has been made possible by the billions of sentences and words on the Internet and in literally thousands of records, comprising literary genres, peer-reviewed research, important documents, and other data. When willing to engage with increased numbers of pieces of literature, word-based summarization is time-consuming, rare, and unsustainable. We prefer using textual data snippets for this justification: it helps save time, we gain actual information rapidly, and computer scientists have been researching ATS counterparts since the 1950s. In this research work, we'll carefully study numerous procedures and provide the best summary based on the score technique used. A wide number of approaches and scoring algorithms will also be scientifically justified. Hybrid text summarization-scoring methods are created by merging extractive and abstractive summarization (ATS) with resource processing ATS. We especially compared the techniques adopting term-based techniques, latent semantic analysis (LSA), lex rank and clustering-based algorithms, and kl sum, in addition to comparing the execution of the relevant information of those in the summary in accordance, including its virtualization. The outcomes of a research study entitled "Smart Textual Summarization in Trying to Score Strategic Plans" demonstrated the best prediction performance and storage systems. Out of the various grading systems, cultural significance, appropriateness, and predictability hint to the methodological framework. Multi-document summarization collections are special, and upgrading mastering technologies is tricky. We provide such a special approach to concatenating several files. We actually research data mining algorithms and offer a technique for multi-document summarizing based on the model. based on word score systems in terms of both words and characters.

**INDEX TERMS:** Introduction to ATS, approaches to ATS, construction of the NLTK model, SUMY, evaluation of automatic text summarization

## 1. INTRODUCTION

Thoroughly retrieving a huge collection of information necessitates some effort and a typo. Furthermore, the objectives of such summarization could be capable of providing extra in-depth information about one particular piece. The recent advancement of documentation and activities has boosted the relevance of automatic summarization handling. The prime objective of accurate word summarization is to deliver the primary sources in a compressed version that retains meaning. Word embedding summary is a technology for splitting textual information. The important characteristics of the material should indeed be summarized into a concise, comprehensible overview. Machine learning with concise overview is a technique for data gathering in the field of natural language processing. It is currently recognizable how vital the knowledge gained in the development procedure is, playing a vital role both within the updated and formal domains, especially given the rapid developments in technology. In the technological environment, machine learning that can optimize term construction and significantly shorten sentences is vital because of the massive volume of data that is gathered every millisecond. Incorporating textual summaries helps to accelerate research. framework, much more than anything, significantly lowers calculation time while simultaneously enhancing the generation of outstanding, lengthy collective knowledge. Farm work is critical; artificial language modelling has already been shown to be ineffective at highlighting texts. An ATS system's overall goal is to deliver a review that compresses and negates redundancy while presenting the main facts of the given document. In accordance with the ATS method [1], viewers may quickly and without difficulty truly understand the significant things in the first piece of content. The readers will earn from the showcase, which is already programmatically displayed, and save a significant amount of time and effort. An absolutely phenomenal explanation should "feel compelled to figure out the most important intelligence collected from a source (or resources) to construct an upgraded version of specific actual text" for a specific individual (or individuals) and activity (or tasks) (May Bury). The summary proceeded, "A conclusion might well be roughly characterized as a piece of writing."

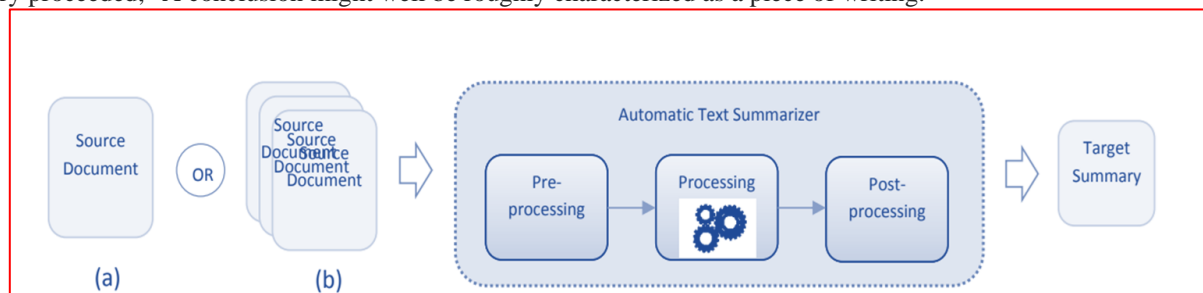


Figure no.1 single or multiple file

ATS systems become either single-document or multi-document systems. The latter provides an overview from a variety of research articles, whereas the earlier generates it from a single document. Section 2: The following tasks are incorporated into an ATS system's overall architecture, as illustrated in Fig. 1. [3]

- I. **Pre-processing:** system is currently establishing the underlying document's information in position using different grammatical techniques, including sentence feature extraction, word smart contracts, eliminating deletion, part-of-speech tagging, separating, and so on.
- II. **processing:** transforming the original document(s) into an overview using one or perhaps several text summarizing algorithms. In Section 3, the vastly separate ATS techniques are outlined. In Section 4, multiple treatments and internal frameworks for implementing an ATS system are reviewed.
- III. **Post-processing:** which entails restructuring the preferred texts before constructing the final summary, represents one of the most challenging tasks in natural language processing (NLP) and machine learning (AI) in general. Luhan's work, which electronically creates highlights from feature articles and is based on ATS, had already been published in 1958. The ATS has a stronger stance on pressing scientific issues, such as the summarization of multi-documents (Hahn & Mani, 2000).

review of the automatically generated summary without comparison to the life form. overview and conclusion are being evaluated. creation of a configuration overview that is comparable to one created by a human (Hahn & Mani, 2000) Researchers are still looking for a comprehensive ATS system that can provide a review that: 1) takes into account all of the significant topics in the text document, 2) does not in itself cover backup or redundant data, and 3) is comprehensible and pertinent to users. In order for computer-generated highlights to compare with life-form highlights, research studies have continued to be conducted and are, at the time of this writing, attempting to establish techniques and procedures for creating overviews. ATS research started in the 1950s.

### 1.1 Need for Text Summarization

There is indeed a considerable amount of text-based documentation in the online world. News stories, novels, movies, speeches, and other types of text may all be Summarized. The amount of material on the Web and in other publications calls for a more accessible and flexible text summary. An automated text summary is a tool for extracting a brief yet insightful summary of text from a document. Using a text summary also reduces reading time, expedites information retrieval, and increases the quantity of details that may be conveyed. Machine learning algorithms are necessary to create machine learning algorithms that can automatically condense lengthy texts.

## 2. APPROACHES OF AUTOMATIC TEXT SUMMARIZATION

Automatic text summarization systems exist in two types: single-document and multi-document. The development of automatic text summarization systems uses one of these techniques.

- I. Abstraction-based Summarization
- II. Extraction-based Summarization
- III. Hybrid-based summarization, third

By employing these techniques, the summarization challenge is converted into a supervised sentence-level assessment opportunity.

### 2.1 Abstraction-based Summarization

The summary applies NLP to characterize the fascinating aspects of green technologies. Instead of only taking the words from the original text to generate the summary, this formula is applied to the capabilities to formulate new phrases. These complements extracting a summary, which supports using a clear meaning to produce a summary. In order to establish a range of times for phrases in the summary, automatic text summarization focuses on the most key things in the data collected from sentences. It's possible that the actual definition used to have a word added on.

#### 2.1.1 Advantages of Abstraction-based Summarization

Applying more favorable terms based on reviewing, analyzing, and compressing results in improved summaries with additional terminology not included in the input version. It will use current deep learning technologies to its advantage. Because of techniques, this one must be treated as a cascade interpretation dedication in order to incorporate the literary material into it.

#### 2.1.2 Disadvantages of Abstraction-based Summarization

In practice, generating a high-quality abstractive summary is very difficult.

### 2.2 Extraction-based Summarization

In this procedure, we begin by focusing on extracting the most important data from the input phrases in order to create a summary. The most remarkable sentences from the input document are chosen and used to develop the summary; no new sentences are constructed; they are absolutely the same as those that were present in the classic variety of input sentences.

#### 2.2.1 Advantage of Extraction-based Summarization

The extraction technique outperforms the abstraction strategy in speed and simplicity, and it succeeds in giving users complete control since it rapidly extracts phrases, which allows them to comprehend the summary using the actual terminology employed by the original.

#### 2.2.2 Disadvantage of Extraction-based Summarization

The extractive approach is far from the method that human experts write summaries.

#### 2.2.3 limitation of the produced extracted summary

- I. Some summary phrases repeat information.
- II. Poor vocabulary and integrity in links between co-referenced data consistent.

### 3. MATERIALS AND TECHNIQUES

**3.1 Collection of Data:** The general methodology and process by which the experimental study has been performed is shown Fig. 2.

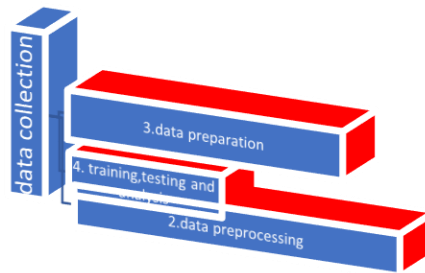


Figure 2: Major Steps of the Proposed Method

In this study, we are using a data-set [1] which has been gathered from individually for the for various topics and it is provided with human generated and algorithm generated summaries for the training and testing of our models. The data-set contains of text files, 2 for each topic one with human generated summary and the other with algorithm generated summary.

**3.2 Description of Data:** we have chosen the .text file in N numbers of randomly files and performed the algorithm's. we prepared the NLTK model .and compared the best accurate summary by using algorithm.

#### 3.3 Common model of Extraction based summary

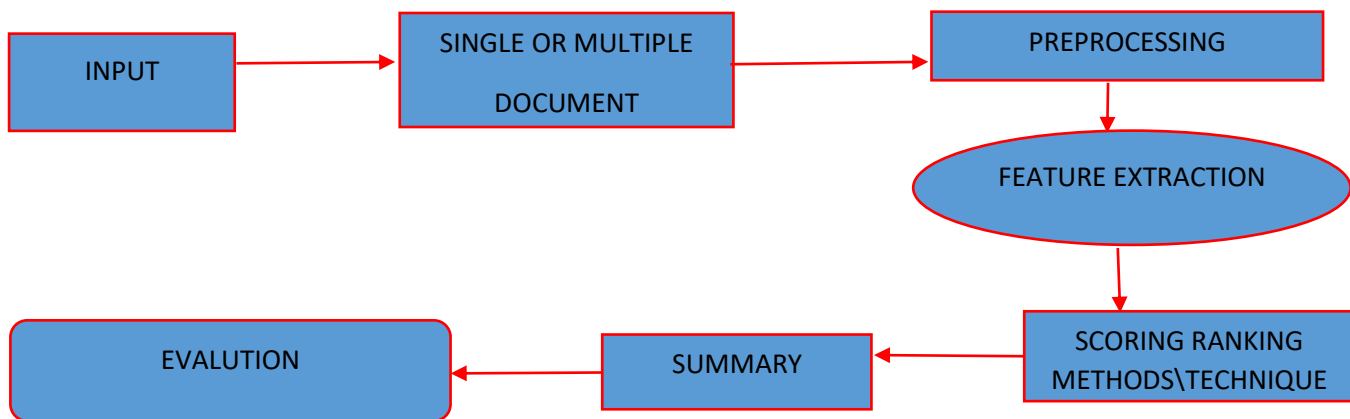


Figure 3: common NLTK model [11]

##### 3.3.1 Algorithm

**INPUT:** Data input is received in text format.

**OUTPUT:** A suitable output text that has been condensed and is shorter than the source text is produced. The extracted, summarized output is presented here.

Step 1 It is to take a.txt file that contains N documents.

Step 2: Apply an extraction technique to tokenize the text file's words and characters.

Step 3: Carrying out the Possession Procedures

Step 4: A feature extraction strategy combining alternative scoring techniques

Step 5: The rankings of each phrase are obtained.

Step 6: Create the NLTK modal (desired summary) using alternative scoring techniques.

Measure their correctness in terms of words and time as well as characters and time in step 7.

Step 8: We'll determine which scoring techniques produce the desired results.

**Prepossessing techniques in ATS:** Frequent prepossessing is carried out in order to eliminate the harsh, unprocessed text. Messages and interactions including errors in the content, including terminology or discarded terminology, are recognized as "additive noise" and "uncompressed text." The following list of prospective gains seems to include a few of the most frequently used benefit techniques:

- I. **Parts of speech (pos) tagging:** Speech tagging is a technique for arranging text words into categories based on speech categories like parts of speech.
- II. **Stop word filtering:** Stop words are dropped either prior to or following textual criticism, depending on the circumstances. Stop words that may be spotted and omitted from plain text provide a, an, and by.
- III. **Stemming:** It eliminates speech patterns and derivative forms from a group of words known as primary or root forms. Text stemming transforms words to examine additional work forms by applying language's methods, such as acknowledgment.
- IV. **Named entity recognition (NER):** Words in the input text are recognized as names of items.

- V. **Tokenization:** It is a text processing method that links text streams into tokens, which can be words, phrases, symbols, or other crucial factors. The technique's purpose is to examine specific words in a document.
- VI. **Noise reduction:** Special characters and punctuation make up the vast majority of textual content's numerous other characters. Key punctuation and special characters are also used. While special characters and significant punctuation are essential for human perception of publications, they can indicate a problem with segmentation systems.
- VII. Others are like slang and abbreviations and capitalization, etc.

### 3.3.2 FEATURE EXACATION: scoring technique in Automatic Text Summarization

- I. Term-Based Technique
- II. Cluster- Based Technique; LEX Rank-Based Technique
- III. LSA - Based Technique
- IV. Statistical Technique; KL-sum

#### I. Term-based (TF-IDF) scoring technique

Phrase-based techniques always adopt the bag-of-words (BOW) model to determine the frequency of a term. This model has various modifications, including the TF-ISF (term frequency-inverse sentence frequency) model and the TF-IDF model. There are 18 one-sentence team-based summarizing techniques, both for single- and multi-document summarizing. Term-based techniques typically use the backpack model to determine the frequency of a phrase that contains multiple variations. The sentence grading techniques should be aware of the words and phrases, length, and relevance, an essential noun, open communication, knowledge of machine learning, nouns, and thinking skills, regulatory agencies by name, memory requirements for words and sentences, cue words, intended frequency, and the undergrowth's shortest path. The efficiency of experimental sentence-scoring techniques is reviewed regularly through global undergraduate research. Sentence scoring methods should take into account a variety of influences, such as the order, length, and centrality of the sentence. These scoring techniques were used as key points in many machine-learning algorithms [10]. Prior to determining the final grade, the proposed multiple-sequence technique helps identify comparable sentences. Once the machine has identified its statistically significant results, the performances are compared using the reader summary [9]. By employing a rapacious technique, this successfully avoids the multiple performance complexity associated with opportunities that require directional order. This technique is utilized in comprehensive text summarizing machines because it essentially simplifies the ability to quickly renew phrases in long and complicated summaries.

#### Cluster based scoring technique

The aspects are dynamically dense peak clustering (DPCS), according to the most recent Z. Hung et al. study. and determined the modifications in the phrases and broad transformation of the data in the discovery ratings. Evaluation components are used to identify the requirement: directional buildup (leading sentences in files are given a grade of 1, and the score tends to drop with a ratio of 1 to N), value of centroid (the average cosine similarity between sentences and the entire area of the sentences in the files), and subsequently, first prison terms intersect (the closest contacts' similarity of a sentence with the first sentence in the same document). Lex rank is used as an example of unsupervised learning in machine learning for a clustering-based word extraction technique. The map performance appraisal data for every significant link that showcases an absolutely vital area will be produced and updated in the database. Each page's contents, which include both frequently used terms and information that is similar to them, are recovered to develop and maintain the summary. Sentence filtering is undertaken for each text using keywords that are conceptually related and commonly used at the end of the process. Every millisecond, phrases that are similar are exploited. Each additional piece of information is recorded, and the collection of papers is then authenticated. A technical short review provides more detailed details about events, occurrences, or anything else. The viewer initially views the data as a specific cause for concern because it's not completely credible. The MDS platform places a significant amount of emphasis on process awareness, although achieving this goal is enormously costly and complicated to implement. There are two processes involved in the summary coding stage. Establishing groups and clusters comes first. Hierarchical clustering is used to design and create the separation in accordance with time. The collected information file summaries are then ordered chronologically. The number of clusters that have been time-tagged before the gathering is demonstrated by a learning algorithm using the clustering technique. The sentences are evaluated using the Stanford Parser. The issue is communicated to the viewers via the application rating.

#### II. Techniques for Lex Rank Score:

The words have a Lex rank in the file that matches all words with the same point of view. As an outcome, every lexical section indicates A, which appears to have been briefly defined in the file. Instead of constructing the noun to produce the conception of the term, a lexical chain is first generated and used with the noun from the file. The best lexical chains are identified and ranked from lowest to highest after the establishment of a lex rank [2]. Since each word in a lexical chain represents the same idea, we decide on the most evident term from every lexical sequence to operate as the lexical rankings reflective. Finally, we petition on summary phrases that make utilize the standard vocabulary.

The Lex rank technique for A top search character that also uses a graph-based methodology for sorting challenges is referred to as "Lex rank," and it is essentially incredibly similar to text rank. By employing the technique of Eigen vector centrality in an internet backbone structure of sentences, Lex rank is utilized to assess the impact of a phrase.

The Lex Rank technique clearly refuses to acknowledge unsupervised texts. The backbone for a detailed overview is a central Eigen vector. Sentences are placed at the intersections of the graph and classified based on how similar they are to one another. Using a cosine similarity metric, endpoints are given greater importance. The key structures are that to the viewer, sentences "recommend" other, absolutely similar phrases. If a conclusion is quite similar to numerous others, it's nearly always a phrase with considerable value. Using lex rank approaches, lex rank for SUMY implementation in NLP



#### Advantage of this techniques

- I. Maintains redundancy
- II. Improves coherency

**Disadvantage of this techniques:** It cannot deal with dangling anaphora problem.

#### III. Latent Semantic Analysis Scoring Techniques (LSA)

using latent semantic analysis, a technique for sustaining summary significance in a document collection (LSA). Essentially, it generates a word and sentence matrix; a single or multiple phrase's weighted term frequency vector from the keyword search is depicted in the column. Only the latent semantic structure is reassembled using the mathematical model of singular value decomposition (SVD), which reveals the relationships between words and phrases on the input matrix. The document collection is assessed to identify a broad range of themes, while during the summary, the sentences with the greatest accuracy weights among all fields of study are accepted. Ferreira et al. continued on their efforts to make sentences richer in their literature reviews. Formation based on grammatical structures and phrase processing capabilities. They believe that the following essential characteristics have previously been underestimated by the scientific field: the dilemma of relevance [4]. The representative's likeness to other candidates in the summary and significance to the number of targets serve as the two major components of the screening and acquisition score in MMR. These reviews account for the entire conclusion of the machine learning, and the operation is concluded when all requirements have been satisfied. The MMR approach has worked well [5] because lengthy documents usually include a lot of big words. It is extensively used to extract words and other metadata from journals associated with a particular thread. A comprehensive multi-document summary has become less and less predictable as an outcome.

#### IV. Statistical Technique

**Sum Basic:** This is typically employed to create multi-document summaries. It employs the fundamental idea of probabilities by putting into consideration that the higher-frequency phrases in the word2vec model of the text have a higher tendency of emerging in the document's summary. The frequency increases as synthesizing phrases are performed. The supervised machine learning kullback-liebler (kl) sum technique will still be employed by myself. The Kullback-Liebler (KL) sum technique, in which the summary size is predetermined, will be discussed (L words). This method excitedly attempts to increase the number of words contained in a summary even if the deviation drops.

#### Overview of KL-SUM

Utilizing ranges lesser than L and Uni-gram appearance, we create a group of paragraphs that are as close to the original text as is practical. An "n-gram" or "Uni-gram" is a continuous stream of n items from a piece of visual or text, as used in the research of natural language processing (NLP). Rating of this approach: In Bayesian inference, the Kullback-Liebler divergence (relative entropy) evaluates how distinct one probability distribution is from another. There is no significant difference between the summary and the document, providing for more effective meaning to be interpreted.  $Kl(p||q)$  describes the Kullback-Liebler (kl) divergence.

$$Q = \log p(w) q(w) \dots \dots \dots (1)$$

**Algorithm:** It uses a greedy optimization approach;

- I. Set  $s = \{\}$  and  $d = 0$ .
- II. While  $||s||=L$  do:
- III. For  $I$  in  $[1..N]$ ,  $d_i = KL(p_s | p_d)$ .
- IV. Set  $s += s_i$  to the smallest  $d_i$  and  $d = d_i$  to the smallest  $d_i$ .
- V. Stop if there is no "I" such that  $d_i$

The possible limitation is to help organize the selected phrases in the major influence on the way by the value of  $p_i$ . The technique used is to compute a position for each selected phrase  $s$  from document  $D$  by following the order in the source documents. The index  $p_i$  (in  $[0..1]$ ) represents the location of  $s_i$  under  $D$ . The words or phrases in the implementing the best are arranged according to the order of  $p_i$ .

#### Features of KL sum

- I. Frequently, the Kullback-Liebler divergence is not a big deal.
  - II. The kl maintains well-defined and invariant statistical distributions, potentially due to variation.
  - III. The two probity mass functions ( $p_1, q_2$ ) and ( $p_2, p_2$ ) are convex if the kl  $D(p^*q)$  in each of the preceding probity mass values ( $p_1, q_1$ ) and ( $p_2, p_2$ ) is identical.
  - IV. Like with the Shannon entropy, the Kullback-Liebler divergence is neutral for separate distributions.
- KL occasionally arises in machine learning, so it is quite beneficial to possess a complete understanding of exactly what the KL-divergence reflects. I recommend reading publications on statistical inference if you're interested in knowing more about KL divergence software solutions in statistics. The KL-divergence basic principle has a long and rich history in computer science.

#### 4.Methodology

It is a pretty easy process where we launch the online application and choose the input file that we want to summaries. The summarization tool will then evaluate this input and stimulate you to choose the most suitable tool for that master plan based on pie graphs that show the accuracy of different algorithms in terms of predicting the speed and execution time within sentences of words and characters.

You can then choose to use a different algorithm or try using one of your choice. The algorithms applied throughout this case are extractive in scope.

This summarization technique will produce an output, and the software will be used to verify its reliability. This part aims to describe the extractive summarization technique that is performed in the text. In this summary duty, the programmed framework excludes goods from the entire selection while affecting the special items. Instances of this include key word extraction, where the objective is to find specified words or expressions to "tag" a record, and document review, where the main objective is to extract complete sentences (without altering them) to generate a concise sequence overview.

Extractive rundowns are created by extracting essential content parts (sentence fragments or entries) from the material by using verifiable examination of individual or paired basic background focuses, such as word/state recurrence, area, or indicate words, to detect the sentences to be extricated.

The "most successive" or "most well located" stuff is recognized as the "most significant" stuff. Thus, such a process provides a comfortable distance from just about any attempt at in-depth knowledge absorption. They are cleverly simple and easy to implement.

## 5.SOFTWARE USED

- I. CMD
- II. Python (3.10.0) & python flask
- III. Pip installer
- IV. Libraries: NLTK (Natural Language Toolkit) and SUMY
- V. WORDNET LEMMAIZER
- VI. Math
- VII. MATPLOTLIB

**I. Command Prompt (CMD):** Command Prompt is a command-line interpreter for Windows operating systems. It is used to carry out instructions given to it, and can lead to difficulties in administrative duties. Sometimes known as Command Shell, CMD Prompt, or even by its filename, cmd.exe.

**II. Python (3.10.0):** Python is a deciphered, elevated-level, universally useful programming language. It's extra-ordinary as a first language since it is brief and simple to peruse. Python utilizes white space, instead of wavy sections or watchwords, to delimit squares.

**III. Pip installer:** The best and largest III. PIP framework is used to implement and monitor Python-created compute packets. The Python Language Benchmark has a large number of combinations and is the prescribed destination for packs and their requirements (Py-PI). PIP is often pre-activated in Python distributions. Pip3 needs to stand for "Programming Language 3" and belongs to Python 3.10.0.pip.

### LIBRAIES:

- I. **NLTK** (Natural Language Toolkit) which contains packages for text processing libraries such as Tokenization, parsing, classification, stemming, tagging, character and word counts, and semantic reasoning.
- II. **SUMY:** We have investigated systematically the idea and use of the Lex rank technique for text accumulation using the observations stated below. Using lex rank approaches, lex rank for SUMY implementation in NLP.

**IV. Word Net Lemmatization:** It is a module in the NLTK stem. It performs stemming using different classes. It is a processing interface for removing morphological affixes, for example, grammatical role, tense, and derivation morphology, from words and learning only the word stem.

**IV. Math:** This module provides access to the mathematical function summarization that mathematical concepts like linear algebra are required for pertaining to vector spaces, matrices, etc. It is essential while calculating the frequency score.

**VI. Mat plot lib:** its modules the plots the graph and structures of required the correct graph for particular topic in python .in the make the graph by this software.

## 6.Result and Discussion

In the work, which has 20 sentences, the system with 10 papers is evaluated. The summarizer delivers the sentences as an output with just a rank greater than 8. Extractive summarization has already been constructed using Python 3.10.0 and NLTK. By using data mining and processing a summary of the results, we were capable of selecting 10 folders from the a.txt file that each comprised 20 sentences. We then used a range of techniques, including the TF-IDF term-based technique, the cluster-based Lex ranking technique, the LSA latent semantic analysis method, and, finally, the KL-sum technique. ultimately decide the shortest extracted summary, then. We rated the summaries based on how quickly the words and characters appeared, and we ultimately created a graph showing the relationship between time and the words and characters. The csv file was created in Word first. We examined how much time was spent on each summary, as well as which character from which file merged into the word in the shortest and longest time. The entire unit is outlined in Table 1.

		mini.time	max. time					
kl-sum		0.02	6.45					
	word	85	76					
	character	496	462					
tf-idf		0.186	0.874					
	word	48	346					
	character	258	1951					
lex		0.05	6.31					
	word	85	44					
	character	496	212					
lsa		0.054	1.47					
	word	45	170					
	character	496	948					

Table 1

The most efficient way to enhance our extractive summary is to leverage the strategies and algorithms we established in the table to reduce the computation and maximal timeframe, then evaluate how many characters are concatenated into words within the maximum and minimal periods of time using all relevant techniques. According to the summary, we identified the Lex rank and LSA upgrading technique. We compared the content and figured that the Lex rank is preferable to the LSA upgrading computer programmer in terms of how quickly the letters can be combined to form words. The KL-SUM strategy was subsequently used, which is a fairly long process compared to all other sorts of techniques. It based on figure 7 and 8 following. Finally we figure out the relationship between time, word vs character in minimum and maximum time. It based on figure 5 and 6.

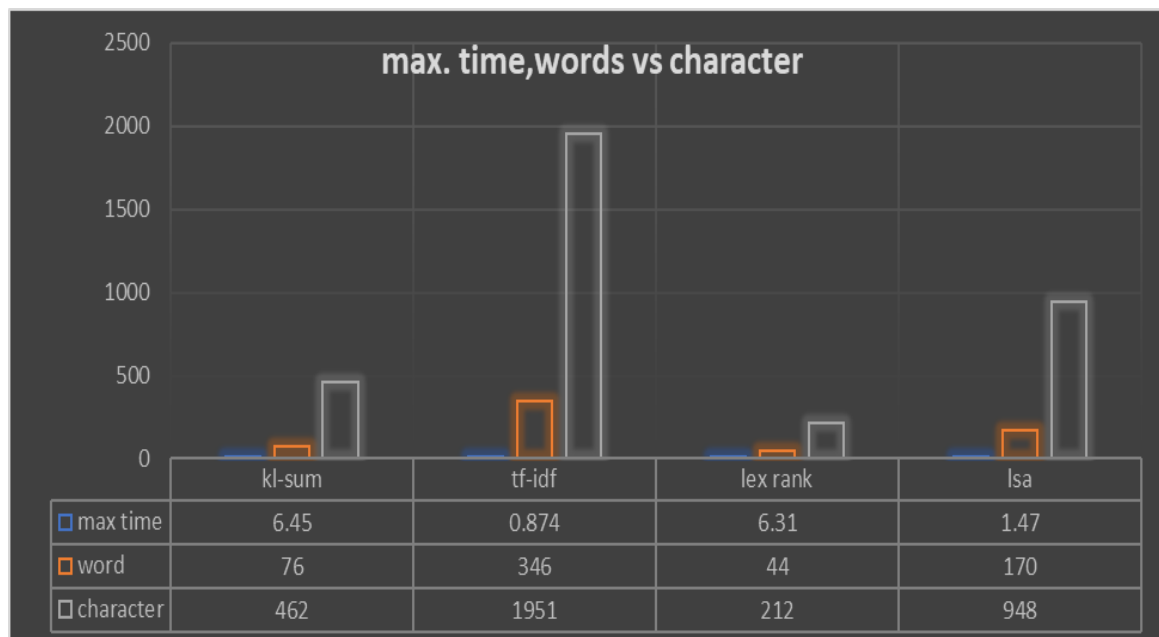


Figure:5

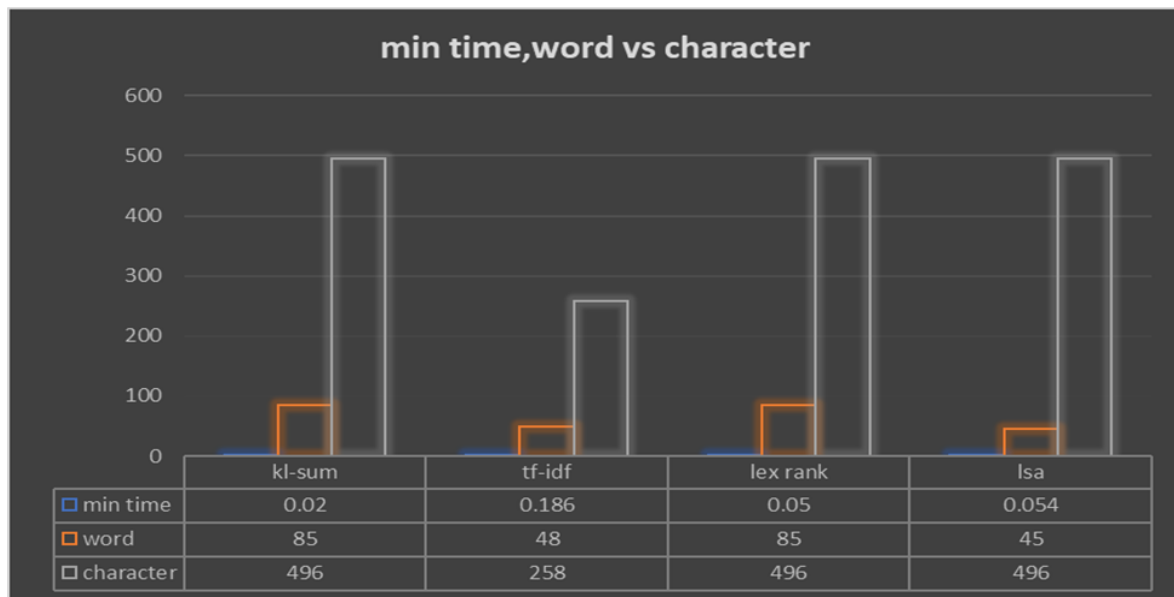


Figure:6

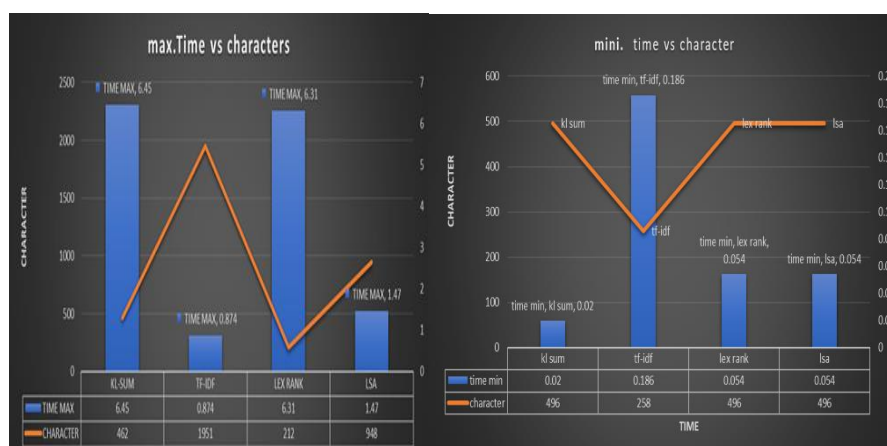


Figure:7

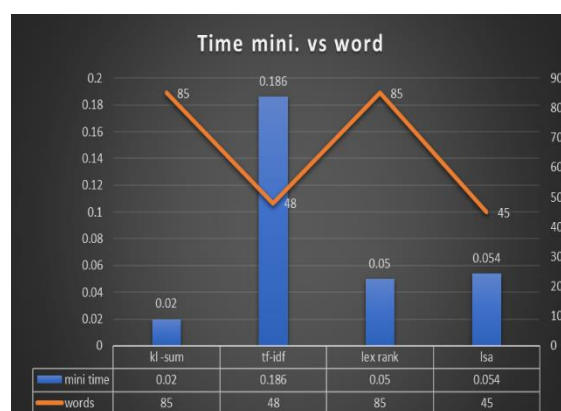


Figure:8



## 7.CONCLUSION

Automatic text summarization is a complicated process with many sub-tasks. Using a statistically specific methodology based on the ranking of the sentences to choose the words or phrases for the summarizer, we suggested extractive-based text summarizing in this study. A text summary is constructed and uses the sentences that were extracted. Comparing the suggested model to the standard approach, accuracy is improved. A user may find it extremely difficult to keep up with all the text that may be of interest when the amount of textual material available electronically increases fairly rapidly. The quality of the resulting summary may be evaluated across different summarizing algorithms in addition to their efficacy in terms of speed and reliability.

## REFERENCE:

- [1] AL Zuhair, A, and AL.Handheld , M., "An approach for combining multiple weighting schemes and ranking methods in graph based multi -document summarization", IEEE Access, vol 7 ,pp.120375-120386,2019.2.
- [2] Cohen A., and Goharian, "Scientific document summarization", IEEE International journal on digital libraries, vol 19(2), pp.87-303,2019.
- [3] Ashraf, M. Zaman and M. Ahmed, "To ameliorate classification accuracy using ensemble vote approach and base classifiers", emerging technologies in data mining and information security, pp.321-334, springer ,2019.
- [4] Sabahi, K. Zhang, Z.P. Nadher, "A Hierarchical Structured Self-Attentive Model for Extractive Document Summarization (HSSAS)", IEEE Access, vol. 6, pp. 205–242, 2018.
- [5] Chen, S.H. lieu and H.M., "An information distillation framework for extractive summarization", IEEE\ACM trans audio speech lang-process., vol26, pp.161-170,2018.
- [6] Naik, S.S., and Ga Onkar, M.N., " IEEE international conference on recent trends in electronics information and communication technology (RTEICT), vol. 9, pp. 156–178, 2017
- [7] Fang, s., Mud., Deng z., and Wu, z., " word sentences co-ranking for automatic extractive text summarization", In expert systems with applications, vol 72, pp.189-195,2017.
- [8] Wang, S., Zhao, X., Li, B., Ge, B., Tang, D.Q., "Integrating extractive and abstractive models for long text summarization," IEEE international congress on big data big edge data congress, pp. 305–312, 2017.
- [9] M. p. Agus., and D. Suharto no, " Summarization using Term Frequency Inverse Document Frequency (TF-IDF)" by Christen, Com. Tech., vol. 7, pp. 285-294, 2017.
- [10] Liu., C. tseng., Machan, "Incest's: towards real time incremental short text summarization on comment streams from social network service", IEEE trans, Knowle., .data Eng., vol 60, pp114,2015.
- [11] KIM, H. Moon and H. Han," A weight adjusted voting algorithm for ensembles of classifiers", Journal of the Korean statistic society, vol 40, pp.437-449, march,2011.