**TEXT DOCUMENT SUMMARIZTION**

**Mathipriya S \*1, Sowmiya S \*2, Harish V \*3, Robero \*4, Dr.G.Kousalya\*5**

\*1,2,3,4 Department of Computer Science, Coimbatore Institute of Technology, Coimbatore-14.

\*5 Professor, Coimbatore Institute of Technology, Coimbatore-14.

**ABSTRACT**

In this new era, where a large amount of information is available on the Internet, it becomes really important to provide an improved mechanism to extract the information quickly and most efficiently. The World Wide Web has brought us a vast amount of on-line information. Every time someone searches something on the Internet, the response obtained is lots of different Web pages with many information, which is impossible for a person to read completely. There is a rapid increase in generation of textual data on a daily basis, be it in healthcare sector, academia, government offices or the corporate sector, the task of maintaining the enormous amount of textual data and processing the data to gain correct and valuable information has become more time and resource intensive. It is very difficult for human beings to manually extract the summary of a large documents of text. So, there is a problem of searching for relevant documents from the number of documents available, and absorbing relevant information from it. In order to solve the above two problems, the automatic text summarization is very much necessary. The basic idea behind summarization is finding the subset of the data which contains the information of all the set. There is a great need to reduce unnecessary data. It is very difficult to summarize the document manually so there is the great need of automatic methods. Our goal with this project is to come up with deep learning model (based on extractive summarization). The most daunting task is to come up with an efficient scoring algorithm which would produce a better output for even a wide range of text.

**Keywords:** Summarization, data, deep learning

1. **INTRODUCTION**

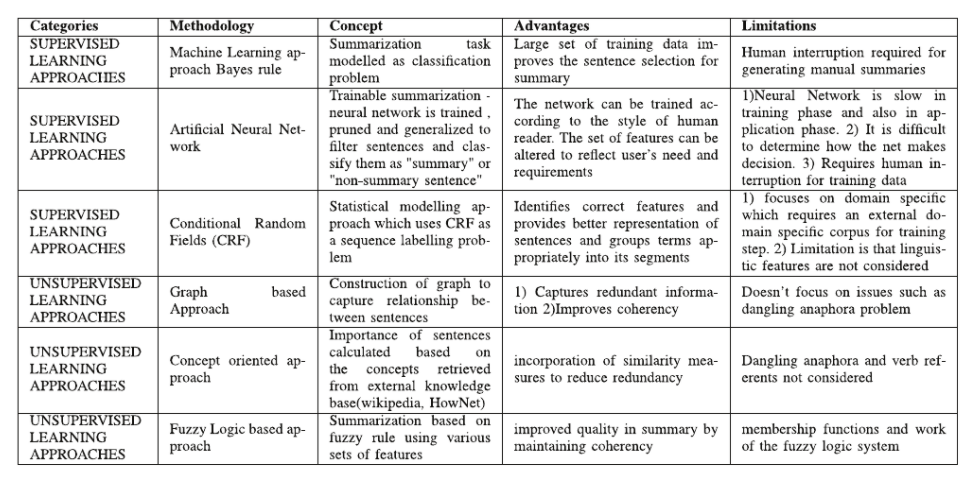
The best solution to reduce the effort and time taken in reading a long document is summarization, which can be done by deep learning. Deep learning is a subset of machine learning, which itself falls within the field of artificial intelligence. Deep learning is a machine learning technique that teaches computers to do what comes naturally to humans: learn by example. With deep learning, computer model learns to perform classification tasks directly from images, text, or sound and can achieve state-of-the-art accuracy, sometimes exceeding human-level performance. Models are trained by using a large set of labelled data and neural network architectures that contain many layers.

Traditionally recurrent neural networks and their variants have been used extensively for Natural Language Processing problems. In recent years, transformers have outperformed most RNN models. The Transformer architecture excels at handling text data which is inherently sequential. They take a text sequence as input and produce another text sequence as output.

BERT (Bidirectional Encoder Representations from Transformers) is a Natural Language Processing Model proposed by researchers at Google Research in 2018. One of the main reasons for the good performance of BERT on different NLP tasks was the use of **Semi-Supervised Learning**. This means the model is trained for a specific task that enables it to understand the patterns of the language. After training the model (BERT) has language processing capabilities that can be used to empower other models that we build and train using supervised learning.

1. **LITERATURE SURVEY**

Best results were obtained with Naive Bayes classifier for both compression rates, using the same features. But with the C4.5 as classifier, the obtained results were poor: the results are similar to the First-Sentences and Word Summarizer baselines. Another approach which uses a seq2seq (encoder-decoder architecture) model with a simple dot product attention. Another model is based on a neural network which consists of one input layer, one hidden layer, and one output layer. The document is fed to the input layer, computations are carried in the hidden layer and an output is generated at the final layer. A comparison of various approaches used is given bellow.



**Figure 1:** Comparison of different approaches

1. **METHODOLOGY**

**Prework:**

Extraction of text from PDF is done using PDFminer package in python. Unlike other PDF-related tools, it focuses entirely on getting and analysing text data. PDFMiner allows one to obtain the exact location of text in a page, as well as other information such as fonts or lines. It includes a PDF converter that can transform PDF files into other text formats (such as HTML). Once the text has been extracted, the following preprocessing steps were done. The entire text is separated as topic and paragraph and stored as key value pairs in python dictionary. Filtering is done to eliminate author name, references, page no from the text, punctuations, any specific marks like IEEE.

**Vamsi/T5 Paraphrase model:**

The model that has been used here is the Vamsi/T5\_Paraphrase\_Paws. It is implemented using TensorFlow, torch, seq2seq, transformers(t5) using transformers attention mechanism and masking function wherein decoder we hide a word and try to predict it.  A Paraphrase-Generator built using transformers which takes an English sentence as an input and produces a set of paraphrased sentences. This is an NLP task of conditional text-generation. The model used here is the T5forConditionalGeneration from huggingface transformer library. This model is trained on the Google’s PAWS dataset and the model is saved in the transformer model hub of hugging face library under the name Vamsi/T5\_Paraphrase\_Paws. Let d denote a document containing several sentences [sent1, sent2, · · ·, sentm], where senti is the i-th sentence in the document. Extractive summarization can be defined as the task of assigning a label yi ∈ {0, 1} to each senti , indicating whether the sentence should be included in the summary. It is assumed that summary sentences represent the most important content of the document.

1. **MODELING AND ANALYSIS**

**MODEL ARCHITECTURE:**

The proposed model uses BERT transformer. BERT is essentially just made up of stacked up encoder layers. We give inputs to BERT using the above structure. The input consists of a pair of sentences, called sequences, and two special tokens: [CLS] and [SEP]. We then get the Token embeddings by indexing a Matrix of size 30000x768(H). Here, 30000 is the Vocab length after word piece tokenization. The weights of this matrix would be learned while training. Final the input given to BERT is **Token Embeddings + Segment Embeddings + Position Embeddings.** We try to predict each word of the input sequence using our training data with Cross-Entropy loss. We mask 15% random words in each training input sequence and just predict output for those words. we will replace any word in 20% of those masked tokens by some random word.

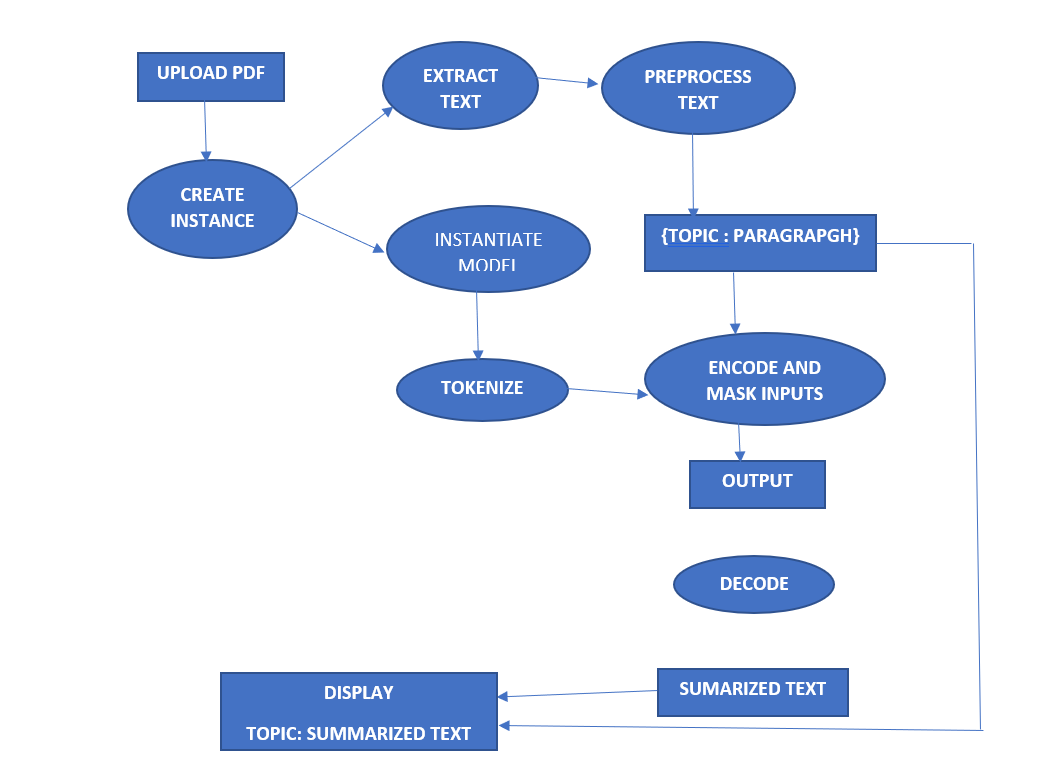
**Diagram

Description automatically generated**

**Figure 2:** Model architecture

**OVERALL SYSTEM DESIGN:**

The proposed system takes input text from the user as a pdf and gives the generated summary as output. The proposed system processes the input using BERT, Attention, VamsiT5 Paraphrase model. Web page is designed using HTML, CSS, python flask. The overall system design is given below.

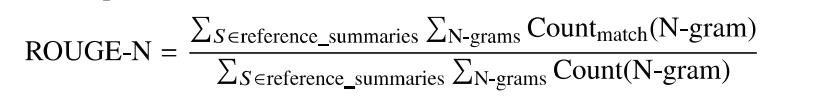
****

**Figure 3:** Flow diagram of the system

1. **EVALUATION METRICES**

**ROUGE:**

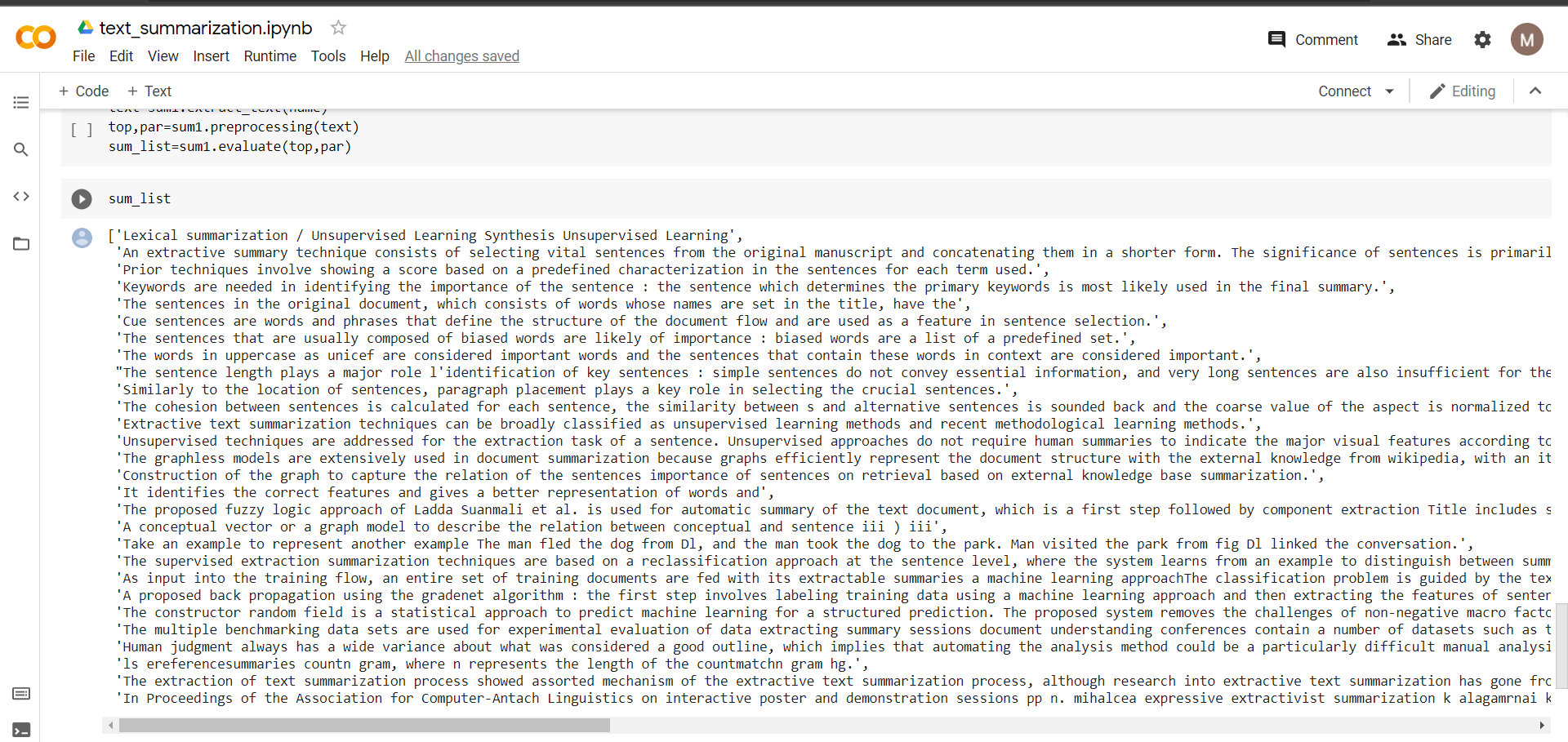
Rouge Formally, ROUGE-N is an n-gram recall between a candidate summary and a set of reference summaries. ROUGE-N is computed as follows:

****

where, n stands for the length of the n-gram Count match (N- gram) is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries. Count(N-gram) is the number of N-grams in the set of reference summaries.

1. **RESULTS AND DISCUSSION**

The main aim of our proposed work is to generate an efficient summary. And we can generate handwritten script for the given input text using our proposed model. The Rouge score of proposed model is 50. A sample of the output is given below.

****

**Figure 4:** Sample of Generated Summary

1. **CONCLUSION AND FUTURE WORK**

Extractive summarization process is highly coherent, less redundant and cohesive (summary and information rich). In this paper, we showcased how pretrained BERT can be usefully applied in extractive text summarization. We introduced a novel document-level encoder and proposed a general framework for extractive summarization. Experimental results across three datasets show that our model achieves state-of-the-art results across the board under automatic and human-based evaluation protocols. In future, the system can be implemented much faster. The system can be advanced to take more pdf as input at the same time, summarize and even find some similarities between those papers. As our model works faster with GPU, we have planned to run this cloud in the presence of GPU, so that the computation will be faster.

1. **REFERENCES**
2. M. G. Ozsoy, F. N. Alpaslan, and 1. Cicekli, "Text summarization using latent semantic analysis," Journal of Information Science, vol. 37, no. 4, pp. 405-417, 2011. Tom S.F. Haines,Oisin Mac Aodha,and Gabriel J. Brostow. “My Text in Your Handwriting” ,2016
3. 1. Mashechkin, M. Petrovskiy, D. Popov, and D. V. Tsarev, "Automatic text summarization using latent semantic analysis," Programming and Computer Software, vol. 37, no. 6, pp. 299-305, 2011. Yousel Elarian, Rand Abdel-Aal, Mohammad Tanvir Parvez. “Handwriting Synthesis: Classification and Techniques”, Analysis and Recognition, 2014
4. V. Gupta and G. S. Lehal, "A survey of text summarization extractive techniques," Journal of emerging technologies in web intelligence, vol. 2, no. 3, pp. 258-268, 2010.
5. J. L. Neto, A. A. Freitas, and C. A. Kaestner, "Automatic text summarization using a machine learning approach," in Advances in Artificial Intelligence. Springer, 2002, pp. 205-215. Tom S.F. Haines, Osin Mac Aodha and Gaberiel J. Brostow University College London. “My Text in your Handwriting”,2019
6. K. Kaikhah, "Automatic text summarization with neural networks," 2004.
7. L. SUanmali, M. S. Binwahlan, and N. Salim, "Sentence features fusion for text summarization using fuzzy logic," in Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on, vol. 1. IEEE, 2009, pp. 142-146.