A

ARTIFICIAL INTELLIGENCE PROJECT REPORT

ON

NewsGlance – News Summarisation

Submitted by:

Rohit (220657)

Radhika Bhati (220650)

Deepanshu Aggarwal (220660)

Under mentorship of

Dr. Atul Mishra

(Assistant Professor)

BML MUNJAL UNIVERSITY™

FROM HERE TO THE WORLD

Department of Computer Science Engineering

School of Engineering and Technology

BML MUNJAL UNIVERSITY, GURUGRAM (INDIA)

December 2024

TABLE OF CONTENT

CANDIDATE'S DECLARATION

We hereby certify that the work on the project entitled, "NewsGlance – News Summarization Project", in partial fulfilment of requirements for the award of Degree of **Bachelor of Technology** in School of Engineering and Technology at BML Munjal University, having University Roll No. 220660, 226650, 220657 is an authentic record of our own work carried out during a period from August 2024 to December 2024 under the supervision of DR. ATUL MISHRA.

Deepanshu Aggarwal

Radhika Bhati

Rohit

SUPERVISOR'S DECLARATION

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Faculty Supervisor Name: Dr. Atul Mishra

Signature:

ABSTRACT

In the fast-paced, ever-evolving world of today, many individuals with busy schedules are faced with a daunting challenge of staying updated on current developments. Reading brief summaries of articles on news can serve as a quick alternative to reading entire articles. This project shall develop an automatic text summarization system that would provide brief, yet comprehensive summaries of news stories. The purpose is to provide approximately 100 word summaries for each article, thus permitting the user to quickly get the gist and investigate further into which stories are of interest. The system utilizes both extractive and abstractive methods to summarize the text. Extractive summarization picks out the most significant sentences from the original text, whereas abstractive summarization generates new sentences based on the content. The CNN/DailyMail dataset, which comprises over 300,000 articles plus their respective human-written summaries, serves as the backbone dataset for this research effort. The dataset is preprocessed and analyzed for the length distribution of both articles and summaries, with keyword and pattern identification by visualization to the extent of using word clouds. As such, it uses four methodologies-the two extractive techniques: frequency-based ranking and TF-IDF, together with two abstractive methods: RNN-based Seq2Seq model and T5 transformer. The presented models have been tested, fine-tuned, and evaluated using standard metrics of the industry, ROUGE-1, ROUGE-2, ROUGE-L, METEOR, and BERTScore. Experimental results show effectiveness in these methodologies for generating accurate and concise news summaries. This project reveals a potential method through which people can obtain effective ways of staying informed while tracking progress in natural language processing regarding text summarization activities.

ACKNOWLEDGEMENT

We are highly grateful to **DR. ATUL MISHRA, ASSISTANT PROFESSOR**, BML Munjal University, Gurugram, for providing supervision to carry out this project from August-December 2024.

DR. ATUL MISHRA has provided great help in carrying out our work and is acknowledged with reverential thanks. Without wise counsel and able guidance, it would have been impossible to complete the training in this manner.

We would like to express thanks profusely to thank **DR. ATUL MISHRA**, for stimulating us from time to time. We would also like to thank the entire team at BML Munjal University. We would also thank our teammates who devoted their valuable time and helped in making this project successful.

LIST OF FIGURES

LIST OF TABELS

LIST OF ABBREVIATIONS

1. **AI** - Artificial Intelligence
2. **BART** - Bidirectional and Auto-Regressive Transformers
3. **BBC** - British Broadcasting Corporation
4. **BLEU** - Bilingual Evaluation Understudy
5. **BERT** - Bidirectional Encoder Representations from Transformers
6. **CNN** - Convolutional Neural Networks
7. **DUC** - Document Understanding Conference
8. **GRU** - Gated Recurrent Unit
9. **NLP** - Natural Language Processing
10. **PEGASUS** - Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence
11. **ROUGE** - Recall-Oriented Understudy for Gisting Evaluation
12. **RNN** - Recurrent Neural Network
13. **SummaRuNNer** - Summarization RNN Extractive Model
14. **T5** - Text-To-Text Transfer Transformer
15. **TF-IDF**: Term Frequency-Inverse Document Frequency
16. **Seq2Seq**: Sequence-to-Sequence
17. **GPU**: Graphics Processing Unit
18. **IQR**: Interquartile Range
19. **CSP** - Constraint Satisfaction Problem
20. **API** - Application Programming Interface
21. **NLTK -** Natural Language Toolkit

Chapter 1

Introduction

1.1. Problem Statement

In such a fast-paced world, the amount of information overwhelms people such that it becomes difficult to keep up with the news. News stories are informative and require a lot of time to read through. People leading very busy lifestyles do not have time for updates, thus a growing need for quicker ways of getting information.

Engaging with full-length news stories in order to extract only necessary information can be a rather useless exercise, especially where readers are mainly interested in the main headlines. The problem is compounded by the sheer number of tedious details that appear in many of the articles and are of little particular interest to the reader. Instead, brief abstracts that summarize the information in an article best serve the purpose of staying current.

It is also not practical and resource-consuming to manually compose such summaries for many articles. Traditional summarization methods, whether they are performed manually or with preset policies, fail to accommodate the growing demand for more timely and accurate information dissemination.

Therefore, there is an immense need for an automated mechanism that can analyse significant amounts of news and provide summaries that are clear, concise, and to the point. The mechanism would effectively address the problems presented by time constraints and information overload while maintaining the relevance and quality of the summarized content produced.

1.2. Objective

The main objective of this project is to design an automated system able to summarize news articles into brief, informative, and coherent texts. Since the necessity for quick access to news highlights has increased, this system will provide summaries of about 100 words that summarize the original article's content without transferring redundant information to the reader.

This project focuses on exploring and applying two fundamental techniques for text summarization:

Extractive Summary: This method distinguishes and selects the most important sentences from the source article, considering factors such as word frequency, ranking of the sentences, or statistical tools such as TF-IDF. It is a direct copy of content from the article, ensuring that the core information is left intact in its pure form.

Abstractive Summary: In this, summaries are generated using highly sophisticated natural language processing methods through reformulating and reordering the information. It comes up with new sentences that summarize essential things regarding the article in question in a fashion similar to that of a human condensing content.

The system is developed and tested based on the CNN/DailyMail dataset that consists of over 300,000 news articles alongside human summaries. This dataset ensures that the system can address various topics and types of articles.

By trying to save time for its readers, this project facilitates more efficient consumption of news, and help in developments in natural language processing and deep learning-based summarization methods, the project automates the summarization process.

1.3. Motivation

In the modern digital era, there is much information generated and distributed daily, especially in journalism. Due to the spread of digital channels, news consumers are exposed to many stories on a wide variety of topics. However, the current reader lacks the time to critically read long news pieces, hence creating a gap between the availability of information and effective uptake.

For those who have a chaotic schedule, access to the most relevant information needs to be as quick as possible. It is impossible to read entire articles on each piece of news so a summary that captures a story's essence if created and provides readers with the ability to stay informed on matters without a large time commitment for long articles.

Although manual summarization is somewhat efficient, it takes a lot of time and resources when scaled up. The recent breakthroughs in natural language processing and machine learning technologies provide great opportunities for automating this process. The practice of text summarization saves time for its audience and also provides a scalable approach to handle vast amounts of textual information, making it particularly useful in fields such as journalism, education, and content aggregation.

This project is inspired by the need to bridge the gap between the high availability of information and the less amount of time that people have, using automated summarization techniques. The initiative explores both extractive and abstractive techniques in order to cover the capabilities of modern algorithms to generate higher quality summaries. Automating the summarization process addresses real-world challenges that readers face and pushes the growing field of natural language processing, thereby making information more accessible and manageable for a larger audience.

1.4. Significance

The automated summarization of texts has increased the importance in the society, characterized by an expanding need for crisp and readily comprehensible information. News consumers face an inflexible flood of articles on a variety of topics, making it impossible to stay up-to-date of the issues without giving significant time to reading. A good summarization system addresses this problem by giving users brief yet informative overviews that enable them to quickly understand the gist of an article.

This project is important because it might change the way people read the news. Summarization saves time and effort; therefore, it is really valuable for busy professionals, students, and anyone else who needs efficient access to information. Besides, it helps people identify articles of interest, thus reading in a more targeted and meaningful way.

Technologically speaking, this project brings great innovations to NLP and deep learning fields. From extracting methods to abstractive summaries, the project implements all the most advanced algorithms at its core to resolve challenges in summarizing a less structured text. The use of

abstractive techniques which are, techniques used when summarizing texts with qualities similar to those of human, further pushes use of machine-based knowledge and language generation.

Moreover, this work has broader applications beyond news summarization. Automated summarization can benefit industries like education, healthcare, and legal services by processing lengthy documents, research papers, or case files into concise summaries. In journalism, it can streamline content aggregation and personalized news delivery.

Chapter 2

Literature Review

Text summarization, one of the major branches under Natural Language Processing, has been explored significantly to handle the growing problem of information overload. This chapter highlights seminal works in both extractive and abstractive summarization schemes with respect to their approach, performance, and limitations. This literature review would serve as a justification to further develop our proposed system.

2.1. Overview of the Papers

Text summarization has been categorized into extractive and abstractive approaches. Extractive techniques rely on identifying and concatenating key phrases or sentences from a source text. Abstractive methods interpret the text, rephrase, and reformulate it, similar to human summarization.

Ramesh Nallapati et al. developed SummaRuNNer-an extractive model based on sequence classification by using GRU. The model achieved an extremely high degree of interpretability since it separated out abstract features such as salience and novelty while learning from human-generated summaries, which came from the CNN/DailyMail dataset. Although effective, the model has limitations in terms of domain adaptation when tested over datasets such as DUC 2002[1].

Transformer models changed the game for NLP tasks, especially abstractive summarization. In comparison of transformer-based models of T5, PEGASUS, and BART, Anushka Gupta et al. observed that T5 was a most effective model with the best balance for ROUGE scores, 0.47 for ROUGE-1. The BART model presented promising fluency and alignment to reference summaries, and PEGASUS produced outputs that were almost too concise[3].

The T5 transformer model presented by Ratan Ravichandran and colleagues used powerful contextual associations to perform abstractive summarization. The paper featured better ROUGE

scores for the 25 epochs of training but highlighted the need for further improvements to make summary coherence better [2].

Extractive summarization picks sentences or phrases directly from the text. One of the prominent studies, Verma and Verma (2020) highlights the benefits of the extractive methods, for instance, simplicity and adhesion to the source content but also points out challenges such as redundancy and a loss of contextual flow [6]. Abstractive summarization creates new sentences by rewriting content. The study brings to light how abstractive methods might yield a more coherent summary but are computation-intensive and prone to hallucination [4].

Varab and Xu (2023) introduced GenX, a comprehensive framework that utilizes generative modeling for a range of extractive and abstractive tasks. Their research illustrates that pre-trained abstractive models, when adequately adapted, are capable of achieving state-of-the-art performance in extractive summarization. The results of their experiments conducted on the CNN/DailyMail dataset indicated enhanced robustness and zero-shot performance when juxtaposed with conventional extractive systems[4].

Similarly, Hermansson and Boddien (2020) studied pre-trained language models such as BERT and RoBERTa for extractive summarization. Their evaluation on academic texts showed that robust pre-training significantly improves generalization to domains outside the training data. However, human evaluations reported readability and coherence as needing improvement, pointing out the inadequacies of current metrics like ROUGE[5].

2.2. Challenges on Recent Trends

- Data Preprocessing: Correct tokenization, stopwords removal, and stemming are very important preprocessing steps that help improve the summarization accuracy as suggested by Gupta et al.'s [3].

- Assessment Metrics: ROUGE is considered the gold standard of metrics for evaluating generated summaries compared to human-written references. However, other metrics, including BLEU and semantic similarity, also deserve further exploration[1][3].

- Hybrid methods are an increasingly popular area of research. The combination method proposed by SummaRuNNer is a balance of simplicity and comprehensiveness[1].
- Verma et al. (2020) presented the following recurring problems in text summarization:
    - Redundancy: Including repetitive content reduces informativeness
    - Irrelevance: Poor choice of features may lead to summarization with irrelevant information.
    - Coverage Gaps: Failure to encompass all relevant topics undermines the utility of summary.
    - Cohesion and Readability: Most summaries by machines lack smooth transitions between ideas[6].

Pattern Development Recent works advocate hybrid methods combining extractive and abstractive approaches for optimized performance. For example, hierarchical ranking models in GenX integrate extractive precision with abstractive flexibility, achieving balanced summaries[4].

2.3. Comparison Table

The comparison table below outlines the performance of different prominent summarization models based on ROUGE scores:

Table 1. Comparison Table of approach used in Literature Review

Model	**Methodology**	**Dataset**	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**Key Observations**
SummaRuNNer	Extractive (RNN)	CNN/DailyMail	26.2	10.8	14.4	High interpretability; domain adaptation challenges.

T5	Abstractive (T5)	BBC News	47.0	33.0	42.0	High fluency and coherence.
BART	Abstractive	BBC News	40.0	28.0	40.0	Fluent summaries; better handling of noisy inputs.
PEGASUS	Abstractive	CNN/DailyMail	42.0	29.0	40.0	Effective, but generated summaries are too concise.
T5 (Small)	Abstractive	CNN/DailyMail	38.0	28.0	38.0	General improvement in efficiency over 10 epochs
BERTSumExt	Extractive (BERT-based)	CNN/DailyMail	42.73	20.13	39.20	Strong baseline; struggles with long document contexts.
GenX (Search)	Generative (Extractive)	CNN/DailyMail	43.57	20.54	40.01	Combines generative and extractive approaches; excels in zero-shot settings.

Chapter 3

Exploratory Data Analysis

3.1. Dataset

In this project, the utilized dataset is the CNN/DailyMail News Dataset, including over 300,000 news articles (exactly 311977 in total), which include summaries developed by humans. Initially built as part of question-answering research, it eventually became a very popular benchmarking test for text summarization work. The dates of CNN news articles range from April 2007 to April 2015, while DailyMail dates range from June 2010 to April 2015. A version of the same dataset is publicly available via Papers with Code. The dataset is divided into three subsets: training, validation, and test sets. The training set has about 287113 articles, the validation set has 13368 articles, and the test set has 11490 articles. The average length of the articles is about 766 words and 29 sentences. The corresponding summaries, known as "highlights," are much shorter, averaging 56 words and 3.7 sentences.

All entries of the database consist of three prime attributes:

- Id (Categorical): an unique identifier for each article-summary pair.
- Article (Text): techniques such as TF-IDF or word embeddings allow the entire unstructured textual content of the news article to be transformed into numerical features, enabling modelling objectives.
- Highlights(Text):The human-composed summary, presented in an unstructured text format, serves as the intended output for models that are built for summarization.

Articles vary greatly in lengths and summary summaries, whereby some articles were beyond 1600 words long or there were highly large summaries to the word limits set. So, in spite of variability, it presents a valid dataset to be used by extractive or abstractive deep models trained for sophisticated text summarizations.

3.2. Exploratory Data Analysis and Visualisations

3.2.1. Missing Data:

There are no missing values in the columns (id, article, and highlights). All entries have non-null values:

- id: 287,113 non-null
- article: 287,113 non-null
- highlights: 287,113 non-null

3.2.2. Outliers:

We identified outliers by calculating the interquartile range (IQR) for article_len and highlights_len, and visualized them using a scatter plot. This revealed some articles with unusually long highlights relative to their length. To resolve this, we applied a threshold and removed rows where the highlights length exceeded 60% of the article length, reducing the dataset from 287,113 to 286,950 rows.

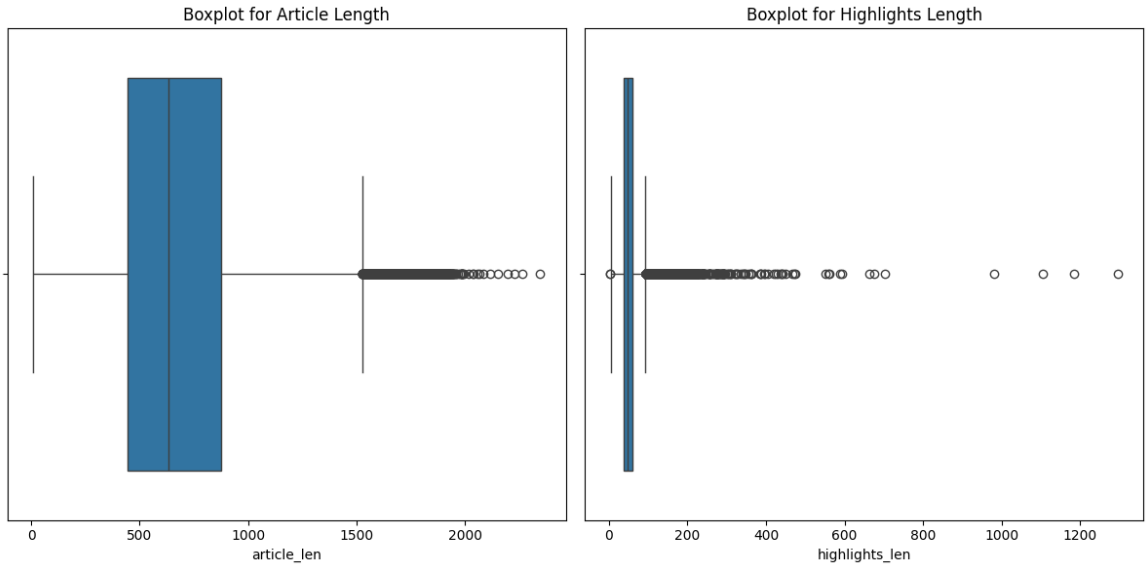


Figure 1. Outlier Detection Using Boxplot

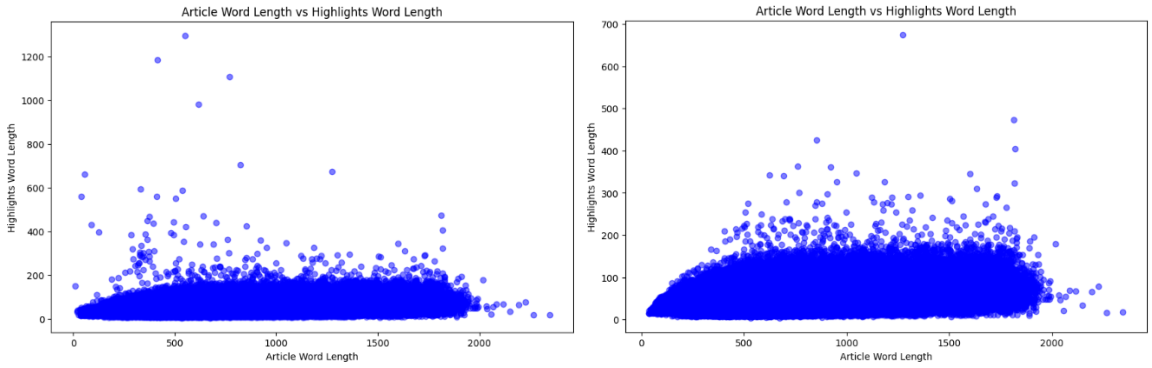


Figure 2. Scatter Plot

3.2.3. Word Cloud

Generating word clouds proved to further support exploratory data analysis by indicating the most common and prominent words in articles and highlights, which aids in grasping common themes and content trends. Together, these visualizations contribute to detecting data quality issues, understanding feature relationships, and making intelligent decisions regarding preprocessing of data.

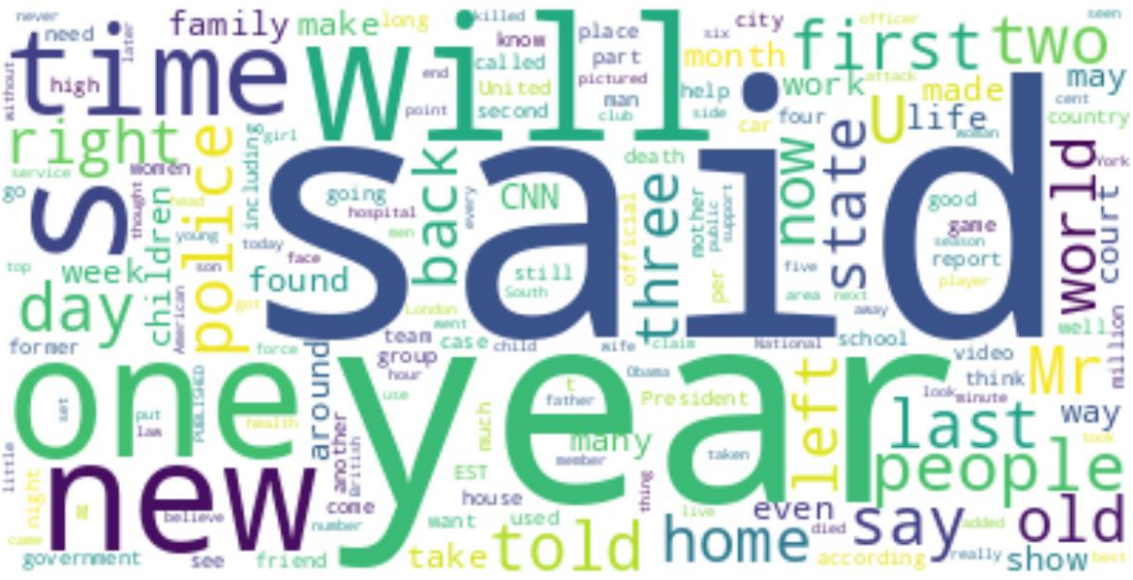


Figure 3. Visualisation through Word cloud

3.2.4. Skewness Analysis

Some of the features in our dataset are skewed. Article_len and highlights_len have a positive skew distribution, so most articles and highlights are smaller, but a few of them

tends to be significantly much longer. Therefore, it creates a long tail in the data distribution.

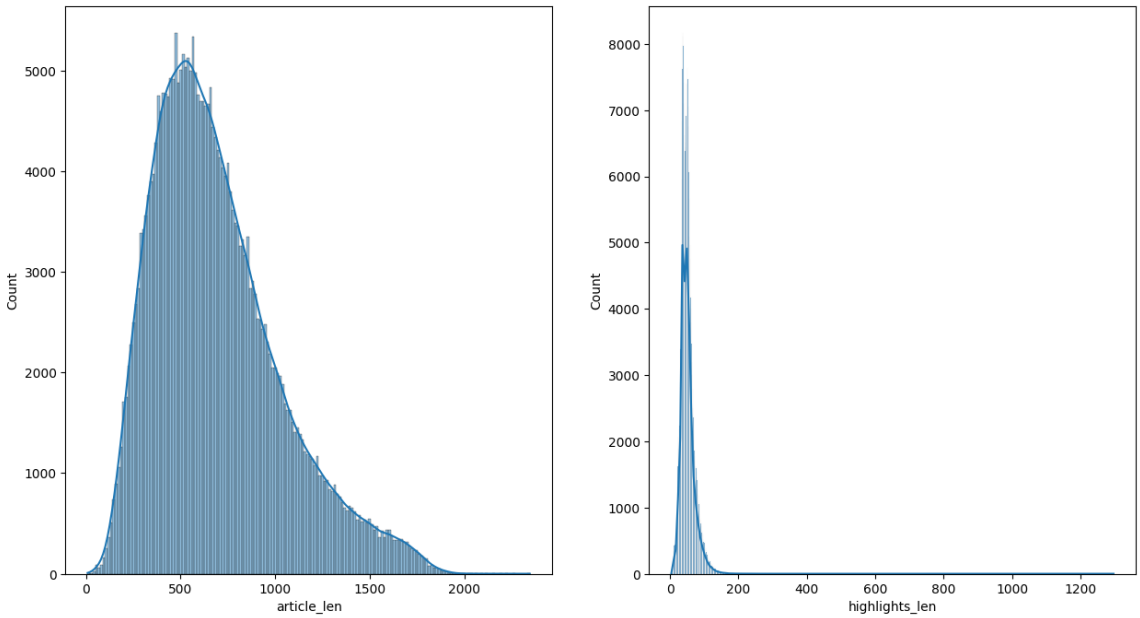


Figure 4. Distribution of Article and Highlight word Lengths

But, we do not need data transformation for our dataset. Since our model is processing only the actual text of articles and not the numeral features such as article_len or highlights_len that are represented by highlighting instead, transforming these length features would not directly improve the performance of the model. Our model's focus is on summarizing text, and it is raw textual data that matters when it comes to training. Length-based features are used mainly in exploratory analysis and outlier detection but do not play a major role in the basic functions of the model. So, no transformation is required for our project goals.

Chapter 4

Methodology

4.1. Problem Statement

4.1.1. Define the Problem

There are challenges with handling the immense amount of news in a fast-paced environment because time is usually less. Despite the useful information that appears in news, their length sometimes renders it challenging to consume the essence of all details in the shortest time possible for readers. There is therefore a high demand for concise, simple summaries that summarize the underlying ideas behind articles. The process of manually summarizing thousands of articles is a labour-intensive and resource-consumption process, thereby requiring the development of an automated approach.

The challenge involves producing a system that can summarize vast news articles into concise and very accurate summaries of about 100 words. This system ensures that the resulting summaries contain important information and easily readable.

4.1.2. Relevance to AI and Real-World Applications

This is a relevant problem in the field of Artificial Intelligence, especially in Natural Language Processing. Automated text summarization is a process that uses deep learning and transformer architectures to analyse large amounts of textual information and extract meaningful insights. This task is challenging in many ways, including understanding context, identifying key elements, and generating summaries that are close to human writing.

In practical cases, self-driving summarization systems do have significant implications. In the journalism world, they aid in the swift transmission of news by providing

summaries of huge articles. In academic settings, summarization tools help the students and researchers to shorten lengthy materials or research papers.

Addressing this problem can significantly enhance the accessibility of information, reduce cognitive load, and improve decision-making across domains with AI.

4.2. State Space Search

4.2.1. State-Space Definition

The state-space can be defined as the different stages involved in the process of converting a complete news article into text forms of brief summaries. Each step involved in the summarization process and options for personalization may well be treated as states in this space.

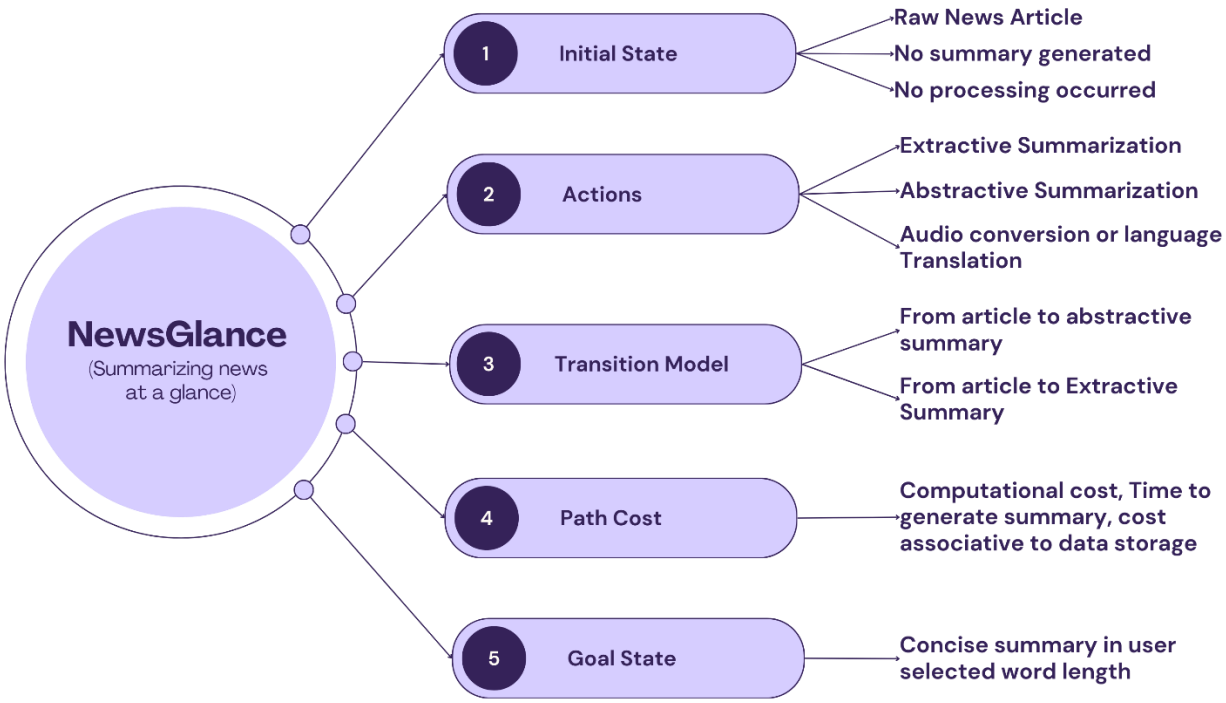


Figure 5. State space representation diagram showcasing

- **State:**

  A state is an instance of a news article in any form, such as original, extracted sentences or abstractive paraphrasing. The state also takes the user preferences like summary length.

- **Initial State:**

  In the initial state, the system is fed with a complete news article for which no summarization has been carried out so far.

- **Goal State:**

  The goal state should be a coherent, readable, and accurate news summary, presented in the format desired-text with options for both extractive and abstractive summaries.

- **Actions:**

  Extract news article sentences based on importance; this is the process of extractive summarization. Obtain summary sentences through natural language generation techniques. This is abstractive summarization. The system presents the user with a summarized content in a format they prefer.

- **Transition Model:**

  Any action results in a change of some form of state. For example, The system moves into a new state from an initial state when it identifies main sentences of an article. It transitions based on the generation of the abstractive summary.

- **Path Cost:**

  The cost of the path can be derived from the resources and time consumed in generating the summary while maintaining accuracy and coherence in the summarized content. The lower the path cost is, the faster and more resource-efficient the summarization process will be.

- **Objective:**

  News digests should be presented to the users as soon as possible and just the way they each want it-in their desired format, and length. The system shall minimize users' time and effort in maintaining informativeness while offering choice.

4.2.2. Search Strategy

- **Description of Chosen Algorithm**

  In order to do extractive summarization **Greedy Search** works by repeatedly picking up sentences by their scores, for example, word frequencies or TF-IDF scores, until some constraint on total length is met.

  For abstractive summarization, **Beam Search** strategy is used in decoding. Beam Search expands the most promising nodes in the state space, keeping a fixed number of hypotheses at each step to optimize the coherence and relevance of the generated summary.

- **Justification and Implementation**

  Greedy Search: This algorithm is effective and suitable for extractive summarization since it promotes sentences with high scores but not exhaustively. It's used by ranking the sentences on the basis of their significance and selecting the best one.

  Beam Search: It is used in abstractive methods for fluency and logical structure because it's a balance between exploration and exploitation. Probabilities of tokens are assigned by the model during generation, but Beam Search identifies the sequence with the highest overall probability.

4.3. Knowledge Representation

4.3.1. Representation Technique and Implementation

It represents both structured and unstructured formats of knowledge for text summarization. This dataset contains text articles, and human generated highlights. Different kinds of techniques of representation used in both the methods process and analyse data:

1. **Extractive Summarization Techniques**
   - **Word Frequency Method**

Text is processed to identify how frequently the terms occur. Those frequencies help in determining the importance of sentences in the article. Important content is extracted by removing stopwords and normalizing term frequencies.

- **TF-IDF Vectorization**

  TF-IDF process in finding the important words converts the sentences into numerical tokens, which will represent the document frequency for the words. This approach ensures correct rankings and selection of significant sentences.

2. **Abstractive Summarization Techniques:**
   - **Seq2Seq RNN Architecture:**

     Articles are tokenized and passed through to contextual embeddings. It uses an encoder-decoder model, GRU in our case, which processes the input and produces summaries in novel language constructs.

   - **T5 Transformer Model:**

     The text is tokenized and represented as embeddings through a pretrained transformer model. The model generates summaries for the input text in text-to-text format, assisted by task-specific prefixes.

4.3.2. Appropriateness and Justification

- Extractive Techniques: Are most appropriate in applications where sentences require minimal transformation from their original form.
- Abstractive Techniques: Enable context-aware and human-like summary generation ideal for advanced NLP applications requiring language transformation.

This multi-method approach ensures the representation is flexible and effective for a wide range of summarization needs.

4.4. Intelligent System Design

4.4.1. Extractive Approach 1: Frequency- based Method

System Architecture:

- Preprocessing Module: Remove Stop words, Punctuations, and Case Folding the text.
- Tokenization Module: Split the text into sentences and words.
- Word Frequency Calculation Module: Calculate the normalized frequency of each word in the text.
- Sentence Scoring and Ranking Module: Score the sentences by normalized frequency of words and ranks them.
- Summary Generation Module: Choose the sentences by ranking them to get summary.

Components and functionalities:

- Text Pre-processing:
  - Eliminates noising words like punctuation, stopwords
  - Converts all input to lower case
- Word frequency Calculation:
  - Calculates the occurrence of each word and scale its values using the document maximum.
- Sentence Ranking
  - Rank each sentence by computing the sum of the occurrences of words it contains
- Output
  - It produces summary in top k ranked sentences.

Innovations

- Light and efficient computationally. Used in low-level summarization tasks for texts.
- Clearly interpretable scoring mechanism that guarantees sentence selection clarity.

4.4.2. Extractive Approach 2: TF-IDF-Based Approach

System Architecture:

- Dataset Module: Manages the loading and preprocessing of datasets (train, validation, test).
- TF-IDF Vectorization Module: It transforms sentences into a weighted vector representation based on term frequency (TF) and inverse document frequency (IDF).
- Sentence Scoring Module: Assigns a score to each sentence based on its TF-IDF weights.
- Sentence Selection Module: Extracts the top k sentences based on scores.

Components and Functionalities:

- Data Preprocessing: Cleans text (removes special characters and noise) and tokenizes it into sentences.
- TF-IDF Vectorization: Computes the weight of each word relative to the document and its corpus.
- Sentence Scoring: Adds the TF-IDF weights of words that are present in each sentence and uses the result as a score of the sentence
- Output : Produces an abstract composed of the most informative sentences, which have the highest TF-IDF scores

Innovations

- TF-IDF enhances accuracy through the emphasis on those terms, which are specifically vital for a given document.
- Adaptable to multi-document summarization by calculating cross-document TF-IDF scores.

4.4.3. Abstractive Approach 1 (RNN with Attention)

System Architecture:

- Encoder Module: Apply a bidirectional GRU to encode the input sequence
- Attention Mechanism Module: Dynamically focus on parts of the input in encoding during decoding
- Decoder Module: Produce the summary sequence token by token
- Training Module: Implement the forward and backward propagation for optimization

Components and Functionalities:

- Encoder
    - Embedding Layer: Convert tokens to their dense representations using pre-trained GloVe embeddings.
    - Bidirectional GRU: The model encodes in both forward and backward directions.
- Attention Mechanism:
    - Calculates encoder output weights where the model will attend to the most relevant information while decoding.
- Decoder:
    - The GRU processes the input at hand and integrates attention for context.
    - Projection Layer: Outputs the probability of tokens over the vocabulary using a softmax layer.
- Training Pipeline:
    - Uses teacher forcing that helps in improving the decoding while training
    - Gradient clipping to handle exploding gradients.

Innovations

- Attention mechanism enables dynamic contextual focus that enhances summary

coherence.

- Pre-trained GloVe embeddings enhance semantic understanding of input text.
- Custom loss function treats variable-length sequences for effective training.

4.4.4. Abstractive Approach 2 (T5 Transformer)

System Architecture:

- Tokenizer Module: Pre-trained T5 tokenizer converts the text into token IDs
- Transformer Encoder Module: It encodes input sequences into contextual representations
- Transformer Decoder Module: Decodes these representations to create a summary
- Fine-Tuning Module: Adapts a pre-trained T5 model to specific summarization tasks

Components and Functionalities:

- Tokenizer:
  - It encodes the text into subword tokens using the SentencePiece tokenizer.
  - It handles padding and truncation for batch processing.
- Encoder:
  - It uses a multi-head self-attention mechanism and feed-forward layers to capture the context of the input sequence.
- Decoder:
  - It uses cross-attention with the encoder outputs to generate the summary.
  - It generates tokens sequentially, conditioned on previous tokens.
- Fine-Tuning:
  - It adapts the pre-trained T5 model on task-specific data using Hugging Face's Trainer API.
  - Evaluation metrics like ROUGE scores assess summarization quality.

Innovations:

- Pre-trained model exploits extensive training on diverse datasets, leading to state-of-the-art performance.
- The T5 model can deal with a variety of NLP tasks due to multi-task learning.
- Scalability allows for handling large datasets with longer input sequences.

4.4.5. Comparative Innovations Across All Models:

Extractive Models (Approaches 1 & 2):

- Faster and less resource-intensive since the architectures are simpler.
- Mechanisms that explain which sentences are chosen.

Abstractive Models (Approaches 3 & 4):

- Produce human-like summaries by generating novel phrases.
- Complex semantics must, therefore, be represented using more advanced architectures, including attention and transformers.

These designs reflect both computational efficiency and advanced modelling innovations tailored for summarization tasks.

4.5. Constraint Satisfaction Problem

Our project aims to generate summaries of news articles using extractive and abstractive methods. To model this as a CSP, we defined variables, domains, and constraints that represent the summarization process and ensure that the system produces accurate and relevant summaries within the required constraints.

4.5.1. List of variables, domains and constraints.

i) List of key variables involved in our problem

- Article Length (A): The length of the news article to be summarized.
- Summary Length (S): The desired length of the summary (in terms of the number of sentences or words).

- Sentence Importance (I): A score indicating the importance of each sentence in the article. This score helps in identifying how relevant a sentence is to the overall article.
- Extractive Summary Sentences (E): The set of sentences selected for extractive summarization.
- Abstractive Summary Sentences (B): The set of newly generated sentences for abstractive summarization.

ii) Identifying the possible values each variable can take

- Article Length (A): Any integer value representing the total number of sentences in the article (e.g., 5 to 100 sentences).
- Summary Length (S): Values between 60 to 100 words (based on the target length of the summary).
- Sentence Importance (I): The score for each sentence is a continuous value, typically between 0 and 1, where 1 indicates the highest importance.
- Extractive Summary Sentences (E): A subset of the article's sentences that have high importance scores.
- Abstractive Summary Sentences (B): New sentences paraphrased from the most important content of the article.

iii) Identifying the constraints that need to be satisfied

- Summary Length Constraint:
  The length of the extractive or abstractive summary (S) must be between 10 and 100 words. $S_{min} \leq S \leq S_{max}$, where $S_{min} = 10$ words, and $S_{max} = 100$ words.
- Sentence Selection Constraint (for Extractive Summarization): Sentences that are longer than the specified threshold are evaluated for inclusion, and the highest-scoring sentences are chosen.
- Content Coverage Constraint (for Abstractive Summarization): The abstractive summary (B) should cover all key points in the article while staying concise and coherent.

iv) Identifying if the constraints are binary, unary or higher-order

- Summary Length Constraint:
  Unary constraint: This constraint applies to the summary's word count, limiting the summary length.
- Sentence Selection Constraint:
  Binary constraint: The importance of each sentence must meet a threshold before it can be selected for extractive summarization.
- Content Coverage Constraint:
  Higher-order constraint: The abstractive summary must paraphrase content that covers all key points while maintaining coherence.

4.5.2. Approach used by the Intelligent Agent to solve the CSP

i) Algorithms used to solve the CSP.

For solving the problem of news summarization, we will employ a hybrid approach combining both:

- Extractive Summarization (using heuristic search algorithms like Greedy Search). The sentences are ranked based on their importance, and the top sentences are selected to form the summary.
- Abstractive Summarization (utilizing Seq2Seq models with Attention and Transformers, T5).

ii) Why we chosen this specific algorithm.

- Extractive Summarization with Greedy Search: These algorithms are computationally efficient, particularly suited for selecting important sentences based on the relevance score (Sentence Importance variable). They help maximize the selection of important sentences in a way that fits the summary length constraint.
- Abstractive Summarization with Seq2Seq Transformers: Models like T5 can paraphrase key information while generating coherent and human-like summaries. These models excel in handling large text inputs and generate semantically meaningful summaries.

iii) List of tools, languages, and libraries used

- Language: Python
- Libraries/Frameworks:
  - Hugging Face Transformers: For implementing transformer model T5.
  - Pandas: For data handling and manipulation.
  - NLTK: For sentence tokenization and processing in extractive summarization.
  - PyTorch/TensorFlow: For training and fine-tuning abstractive summarization models.

iv) Algorithm Implementation

- **Extractive Summarization Implementation**:
  - It uses text processing techniques to rank the importance of each sentence.
  - The top-ranked sentences are selected based on user-defined limits (e.g., the number of sentences).
  - The extractive summary is displayed after processing.

- **Abstractive Summarization Implementation**:
  - The article is passed to a **pre-trained transformer model T5**.
  - The model generates an abstractive summary by considering the content and rephrasing key points in a new way.
  - The generated summary is checked for length to ensure it falls within the predefined constraint (10–100 words).

v) Step by step process how the CSP is solved by the agent.

- **Input Parsing**: The agent takes a news article and user preferences.
- **Preprocessing**: The article is tokenized into sentences, and importance scores are computed for each sentence.

- **Extractive Summarization**: Sentences are ranked by importance, and the highest-scoring sentences are selected while ensuring the summary length constraint is satisfied.
- **Abstractive Summarization**: The article is passed through the transformer model to generate a summary in a concise form, ensuring all important information is retained.
- **Final Output**: The agent outputs the text summary

4.5.3. How the algorithm solves this problem

i) **Input Parsing**: The user provides a news article along with preferences, such as whether they want an audio summary or a specific language for the summary.

ii) **Preprocessing**: The article is split into sentences, and each sentence is scored based on its importance using methods like TF-IDF or embeddings (for extractive summarization).

iii) **Extractive Summarization**: The algorithm selects the top-ranked sentences (based on importance scores) while ensuring the summary stays within the desired length (10-100 words).

iv) **Abstractive Summarization**: If the abstractive option is chosen, the article is fed into a pre-trained transformer model **T5**. The model generates a new summary by rephrasing key information in a concise and coherent way, also adhering to the word count constraint.

v) **Output**: The algorithm finally delivers the summarized text (in extractive or abstractive form) ensuring the user receives a concise, accessible news summary.

4.5.4. Comparative analysis

i) **Comparison with Other Summarization Tools**:

- Compared to other available summarization tools like BERTSUM and Gensim, the intelligent agent offers flexibility with user-defined parameters for summary length, allowing for a more customized experience.

- While BERTSUM excels in extractive summarization with high accuracy, the agent's use of T5 for abstractive summarization allows it to generate more human-like summaries.

ii) **Comparison of Extractive vs. Abstractive Summarization**:

- **Extractive Summarization**: Generally faster and maintains original sentence structure, making it easier to understand but may lack coherence in the final summary.
- **Abstractive Summarization**: Takes longer but often provides a more coherent and contextually relevant summary, albeit with a higher chance of losing fidelity to the original content.

Chapter 5

Results

The extractive and abstractive summarization approaches were compared using multiple performance metrics. The following insights can be derived from the results, but it should also be noted that extractive and abstractive approaches are inherently different and thus cannot be directly compared.

5.1. Extractive Approach

The two extractive summarization methods adopted are as follows: Extractive 1, or Text Ranking Based, and Extractive 2, or TF-IDF Based. Results across various performance metrics show the clear supremacy of the TF-IDF-based method.

Table 2. Metric results for Extractive Approach

Metric	Extractive 1 (text ranking based)	Extractive 2 (TF-IDF based)	Best Approach
ROUGE-1	15.97	35.08	TF-IDF
ROUGE-2	8.3	16.7	TF-IDF
ROUGE-L	11.87	23.16	TF-IDF
METEOR	33.38	37.53	TF-IDF
BERTScore	81.09	86.69	TF-IDF

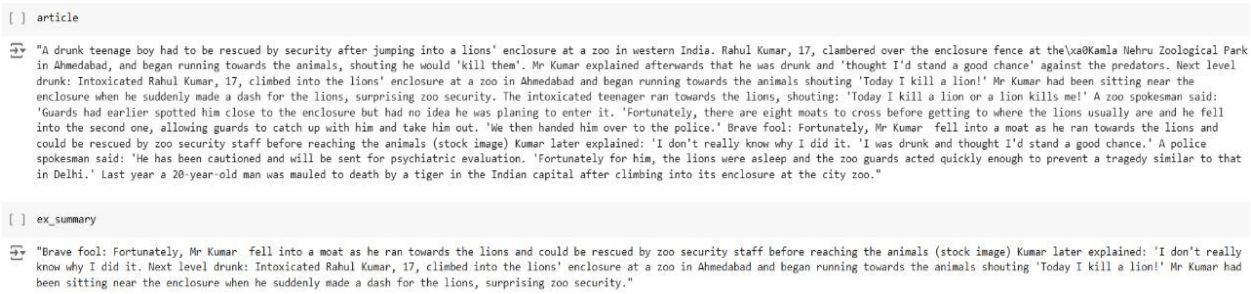
5.1.1. Observation

i) **Performance of TF-IDF:**

- TF-IDF scores better for all metrics such as ROUGE-1 at 35.08, ROUGE-2 at 16.7, and BERTScore at 86.69.
- That means that TF-IDF will be more precise in the retrieval of salient sentences and the context that relates to it as well.

ii) Text Ranking: Limitations

- The text ranking-based technique lags considerably in the ROUGE-2 and ROUGE-L metrics which count overlap in multiword sequences and the longest matching sequence respectively.
- This implies a very poor capability to pick the most contextually relevant sentences.



```
[ ] article

"A drunk teenage boy had to be rescued by security after jumping into a lions' enclosure at a zoo in western India. Rahul Kumar, 17, clambered over the enclosure fence at the\xa0Kamla Nehru Zoological Park
in Ahmedabad, and began running towards the animals, shouting he would 'kill them'. Mr Kumar explained afterwards that he was drunk and 'thought I'd stand a good chance' against the predators. Next level
drunk: Intoxicated Rahul Kumar, 17, climbed into the lions' enclosure at a zoo in Ahmedabad and began running towards the animals shouting 'Today I kill a lion!' Mr Kumar had been sitting near the
enclosure when he suddenly made a dash for the lions, surprising zoo security. The intoxicated teenager ran towards the lions, shouting: 'Today I kill a lion or a lion kills me!' A zoo spokesman said:
'Guards had earlier spotted him close to the enclosure but had no idea he was planing to enter it. 'Fortunately, there are eight moats to cross before getting to where the lions usually are and he fell
into the second one, allowing guards to catch up with him and take him out. 'We then handed him over to the police.' Brave fool: Fortunately, Mr Kumar  fell into a moat as he ran towards the lions and
could be rescued by zoo security staff before reaching the animals (stock image) Kumar later explained: 'I don't really know why I did it. 'I was drunk and thought I'd stand a good chance.' A police
spokesman said: 'He has been cautioned and will be sent for psychiatric evaluation. 'Fortunately for him, the lions were asleep and the zoo guards acted quickly enough to prevent a tragedy similar to that
in Delhi.' Last year a 20-year-old man was mauled to death by a tiger in the Indian capital after climbing into its enclosure at the city zoo."

[ ] ex_summary

"Brave fool: Fortunately, Mr Kumar  fell into a moat as he ran towards the lions and could be rescued by zoo security staff before reaching the animals (stock image) Kumar later explained: 'I don't really
know why I did it. Next level drunk: Intoxicated Rahul Kumar, 17, climbed into the lions' enclosure at a zoo in Ahmedabad and began running towards the animals shouting 'Today I kill a lion!' Mr Kumar had
been sitting near the enclosure when he suddenly made a dash for the lions, surprising zoo security."
```

Figure 6. Summary generated by Text Ranking

Article 2: by . alex greig . published . 0939 est 2 january 2014 . . updated . 1300 est 2 january 2014 . a human skull has been found on a paper plate surrounded by beads in california. police responded to a tip of a sighting of human bones along grizzly peak boulevard in the grizzly peak area of berkley. officer johanna watson told cbs that police found the human skull out in the open on wednesday afternoon. scroll down for video . skull found the skull was found on grizzly peak boulevard a high ridge popular with hikers and mountain bikers . strange discovery police arrived to investigate the skull sighting around 540pm wednesday . little is known about the skull including its age but officer said it appeared to be more than several years old. according to the mercury news police do not believe the skull was of native american origin. . the alameda county coroner is examining the skull. grizzly peak is a summit in the berkley hills popular with hikers cyclists and tourists. part of me is not surprised for how many different types of people i see up here all the time local resident dakota defiore told cbs. mystery police say the skull appears to be old but have not released any further information stock image you know its dark and sometimes people drink up here. but part of me is kind of shocked you know never really thought driving up here id see a bunch of cop cars. police are investigating the bizarre discovery. they have not made any further information available but will provide an update on the case thursday morning.

Extractive Summary 2: police responded to a tip of a sighting of human bones along grizzly peak boulevard in the grizzly peak area of berkley. officer johanna watson told cbs that police found the human skull out in the open on wednesday afternoon. part of me is not surprised for how many different types of people i see up here all the time local resident dakota defiore told cbs. mystery police say the skull appears to be old but have not released any further information stock image you know its dark and sometimes people drink up here. but part of me is kind of shocked you know never really thought driving up here id see a bunch of cop cars.

Figure 7. Summary generated by TF-IDF

5.1.2. Important Insight

The TF-IDF is the best extractive summarization method because it can assign meaningful weights to words and can prioritize sentences with more information density.

5.2. Abstractive Approach

Two abstractive summarization methods were tested: RNN-based summarization and T5-based summarization. It is very clear that T5 has outperformed the RNN-based approach.

Table 3. Metric results for Abstractive approach

Metric	RNN	T5	Best Approach
ROUGE-1	14.32	27.07	T5
ROUGE-2	2.21	9.16	T5
ROUGE-L	12.01	19.69	T5
METEOR	9.23	21.31	T5
BERTScore	75.06	86.38	T5

5.2.1. Observations

i) T5 Model Supremacy:

- T5 surpassed the baseline in all the metrics by a significant margin: ROUGE-1: 27.07, ROUGE-2: 9.16, and METEOR: 21.31.
- The transformer architecture and pretraining on large datasets enable T5 to better understand complex patterns in language and generate coherent human-like summaries.

ii) Limitations of RNN Approach

- RNN-based approach did not do well with metrics like ROUGE-2: 2.21 and METEOR: 9.23, which meant it lacked the ability to produce good multi-word summaries and semantically rich sentences.
- RNNs tend to drift away from the context in a longer sequence, which impacts overall performance.

article: The President doled out fist bumps and pats on the shoulder to the delegation stationed along the red carpet at the foot of Air Force One, which the White House said was part of an effort to reduce physical contact amid the rapid spread of a new coronavirus variant.\nBut only minutes later, Biden broke with the new effort -- exchanging a hearty handshake with the former Israeli Prime Minister Benjamin Netanyahu, the current opposition leader. Biden later also clasped hands with a pair of Holocaust survivors at Yad Vashem.\nThe President's half-hearted effort to reduce physical contact on his trip to the Middle East was a jolting switch for Biden, who has been shaking plenty of hands in the days leading up to his trip. And it raised questions about whether the White House was trying to avoid the optics of Biden shaking hands with Saudi Arabia's Crown Prince Mohammad bin Salman, whom Biden will meet for the first time on Friday.\nBiden has been facing questions about that meeting after saying as a candidate he would make Saudi Arabia a "pariah" for the killing of journalist Jamal Khashoggi. The CIA has alleged that Khashoggi was killed in an operation approved by bin Salman.\nPressed by reporters aboard Air Force One, White House press secretary Karine Jean-Pierre denied that was the reason for the attempt at reduced physical contact.\n"We are saying that we're going to try to minimize contact as much as possible. But also, there are precautions that we are taking because this is up to his doctor. BA.4, BA.5 is indeed, as we're seeing, increasing. And we want to make sure that we're taking those precautions to keep him safe and to keep all of us safe," Jean-Pierre said, referring to emerging coronavirus variants.

summary: President's half-hearted effort to reduce physical contact on his trip to Middle East was a jolting switch . Biden has been shaking plenty of hands in the days leading up to his trip . White House press secretary Karine Jean-Pierre denies that was the reason for the effort . CIA has alleged that Jamal Khashoggi was killed in an operation approved by bin Salman .

Figure 8. Summary generated by T5 Model

5.2.2. Important Insights

T5 is an abstractive summarization process that is the best due to its advanced architecture in handling diverse and long text inputs.

5.2.3. Comparison between Extractive and Abstractive Models

Extractive and abstractive models are fundamentally different concepts and cannot be compared straightaway.

- Extractive methods select sentences directly from the original text, ensuring factual accuracy but limiting flexibility.
- Abstractive methods generate entirely new sentences, offering more creative and concise summaries.

5.3. Final Observations

- For extractive summarization, the TF-IDF-based approach is the best, providing high accuracy and relevance in extracted sentences.
- T5 models outperform RNNs significantly for abstractive summarization, providing coherent, semantically rich summaries.

Both approaches have their strength in applications. Comparing between the extractive and abstractive summarization does not stand for that the two are to be put directly for comparison.

5.4. Deployment on Streamlit

Besides the comparison of extractive and abstractive summarization methods, the best-performing models of both approaches were deployed on a Streamlit-based interface. The user-friendly interface is a simple interface that can input news articles or lengthy texts for immediate view of the generated summaries by the TF-IDF-based Extractive model and the T5-based Abstractive model.

Since deploying a completely workable application will fill out the gap between technical implementations and actual use in this world, it makes a model much more accessible in general:

5.4.1. Features of the Streamlit Deployment

- **Dual Summary:**
  - Extractive summary generated based on TF-IDF methodology.
  - Abstractive summaries created using the T5 model.
- **Interactive Inputs:** Users can paste articles or long text as input.
- **Easy Comparison Results**: Both extractive and abstractive summaries are showcased together for an easier comparison.
- **Real-Time Execution:** The application yields instant results, which shows how efficient both these models are.

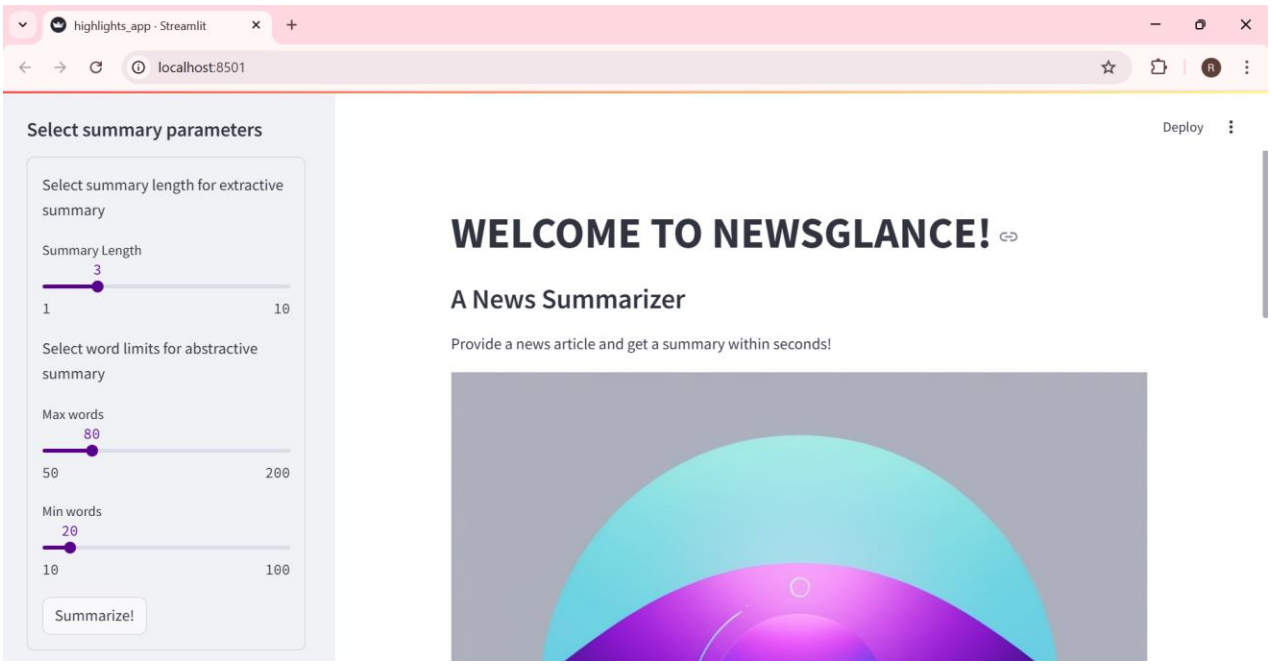


Figure 9. Interface Overview

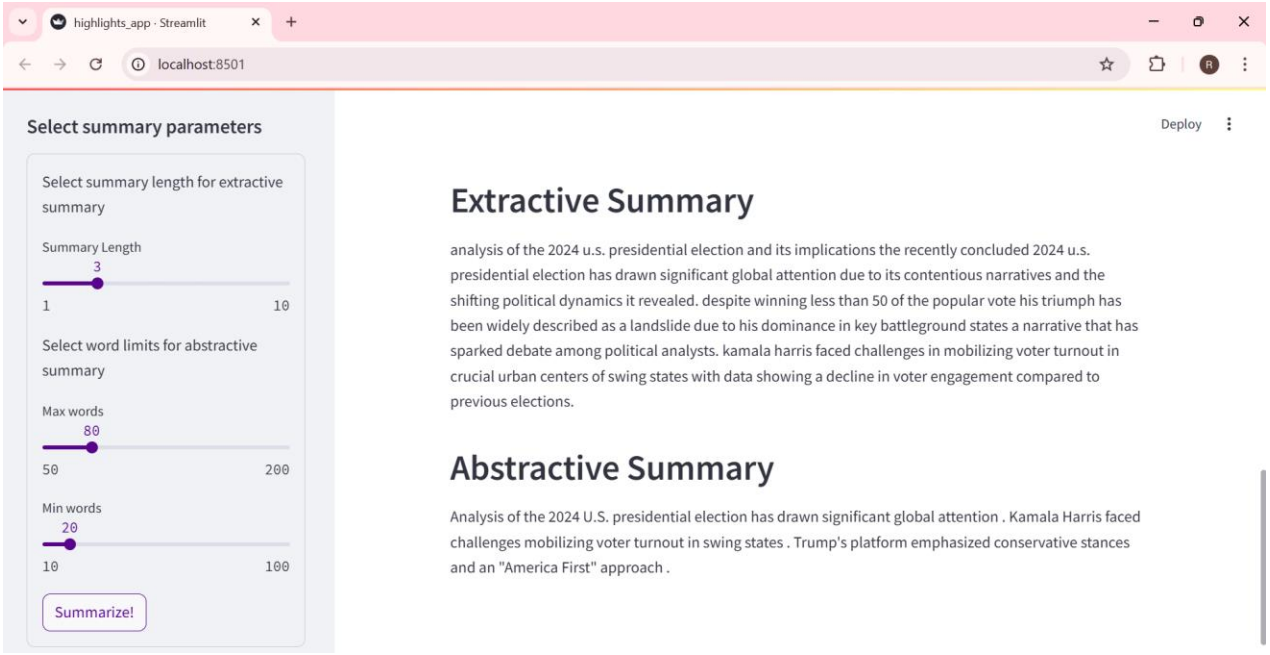


Figure 10. Extractive and Abstractive Summary Output

Chapter 6

Conclusion and Future Scope

6.1. Conclusion

Extractive and abstractive summarization methods can be compatible with different architectures. Important takeaways are given below:

- Extractive Summarization: The method which is based on TF-IDF was most successful as it achieved the much higher ROUGE and BERTScore values than a text-ranking-based method. This points out that weighting as well as prioritizing textual information is critical for the extraction techniques.
- Abstractive summarization: With the complex transformer architecture, T5 was apparently better than a baseline RNN in all of the evaluation metrics. With this, making it the state-of-the-art abstractive summarization model, which generates human-like summaries with much higher coherence and relevance.
- Comparison: Since extractive and abstractive methods are differently fulfilling the needs, it cannot be compared. Extractive methods are based on facts taken through the original text. Whereas in an abstractive method, an abridged summary is generated while ensuring novelty and generally well capturing the main idea of input.

Results from these experiments clearly indicate that the outcome of which approach is good and not so good differs fundamentally - for most problems, depending on the type, such as precision, fluency, or contextual relevance, no single approach is stronger than another.

6.2. Future Scope

According to the current findings, there could be lots of future exploration areas:

i) Hybrid models: When both extractive and abstractive models are combined, strengths would come together to work towards factual correctness, richness in semantics, and ease of reading.

ii) Fine-Tuning on domain-specific corpora: models learnt from fine-tuning on domain-specific datasets would do a better job in those areas where subtle comprehension is very much required.
iii) Better Metrics of Evaluation: ROUGE, METEOR and BERTScore are some useful metrics but there's much more to be talked about new metrics which will better capture the semantics as well as coherence of the summary.
iv) Multilingual Summarization: Models that can handle multiple languages or cross-lingual summarization can make these techniques more applicable in a global scale.
v) Explainability in Summarization Models Developing methods for explaining how models prioritize or generate content can enhance trust and adoption, especially in critical applications.

Addressing these directions can push summarization models forward into much more powerful, versatile tools more applicable within natural language processing.

Bibliography

[1] Nallapati, R., Zhai, F., Zhou, B. (2016). SummaRuNNer: A Recurrent Neural Network-based Sequence Model for Extractive Summarization of Documents

[2] Ravichandran, R., Sharma, S. B., Das, S. (2023). Text Summarization Using the T5 Transformer Model.

[3] Gupta, A., Chugh, D., Katarya, R. (2023). Automated News Summarization Using Transformers.

[4] Vaswani, A., et al. (2017). Attention is All You Need. **ArXiv abs/1706.03762**.

[5] Varab, D., & Xu, Y. (2023). Abstractive Summarizers are Excellent Extractive Summarizers. Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. Retrieved from https://github.com/danielvarab/GenX .

[6] Hermansson, E., & Boddien, C. (2020). Using pre-trained language models for extractive text summarization of academic papers. Master's Thesis, Chalmers University of Technology, Gothenburg, Sweden.

[7] Verma, P., & Verma, A. (2020). A Review on Text Summarization Techniques. Journal of Scientific Research, Volume 64, Issue 1. Banaras Hindu University, Varanasi, IndiaS.