


Research Article

An Extractive Text News Summarization: A Hybrid Optimization with Ensemble Learning Approach

Estabraq Abdulreda Kadhim¹ 
ComInSyS Lab, Department of
Computer Engineering
University of Tabriz
Tabriz, Iran
estabraq.ai.1989@gmail.com

Mohammad-Reza Feizi-Derakhshi² 
ComInSyS Lab, Department of
Computer Engineering
University of Tabriz
Tabriz, Iran
mfeizi@tabrizu.ac.ir

ARTICLE INFO

Article History

Received: 18/12/2023

Accepted: 29/1/2024

Published: 1/5/2024

This is an open-access
article under the CC BY
4.0 license:

<http://creativecommons.org/licenses/by/4.0/>

**ABSTRACT**

Automatic Text Summarization is a crucial feature for managing the ever-increasing volume of textual data. However, existing methods often struggle with feature identification for sentence importance, which leads to a lack of maintained narrative coherence and accuracy. In this proposed approach, the summarization process leverages the Chi-square Binary Cuckoo Search (Chi-BCS) method for feature selection, this optimizes text features enhance the summary content and utilizes insights from classification to ensure summaries are contextually relevant and concise. Feature selection aims to improve the performance of machine learning models by reducing the dimensionality of the input data and removing irrelevant or redundant features. Classification, on the other hand, contributes to better summarization by distilling lengthy or redundant content into key points, thereby enhancing both efficiency and accuracy. The proposed approach implements a model that leverages advanced Natural Language Processing and machine learning techniques for effective extractive summarization on both BBC and CNN/DailyMail datasets. Key features extracted from the text include Named Entity Recognition, Cue phrases, TF-IDF, Sentence position, sentiment analysis, etc. Various algorithms are employed to improve classification performance, such as Decision Trees, Support Vector Classifier, Gradient Boosting, Random Forest, K-Nearest Neighbors, and Logistic Regression. Among all the methods evaluated, the Random Forest and Ensemble Hard Voting approach achieved the highest F-score of 96.26 and 0.9322 respectively on the BBC and CNN/DailyMail dataset. In the text summary evaluation, the ensemble method also delivered exceptional results, with ROUGE-2 and ROUGE-L F1 scores reaching 0.799 and 0.818, respectively on BBC. While our ensemble model achieved to high score on ROUGE1 and ROUGE 2 reaching 0.275, 0.5017, respectively on CNN/DailyMail when compared with state of art highlighting the model's strong performance. These findings demonstrate that the proposed model is highly effective for both the classification and summarization of large-scale textual data.

Keywords: List three to five key words here. With joiner symbol (;)

1. INTRODUCTION

This growth in textual data from news items, novels, journals, scientific publications, databases, and medical records is a big breakthrough [1]. However, this rapid increase has created a new challenge: when people search for information, they are often confronted with an overwhelming volume of data, making it difficult to identify relevant results. To address this, information needs to be condensed so that users can quickly access what they need. Yet, this solution introduces another issue [2,3], the complex and time-consuming process of manually summarizing such large amounts of data [4,5].

Natural Language Processing techniques have become increasingly important. NLP techniques have significantly improved the efficiency and accuracy of various tasks, such as information retrieval, sentiment analysis, and language translation. These advancements have transformed how humans interact with machines [6,7], leading to more intuitive and effective communication [8]. These techniques enable the extraction not only of explicit and implicit knowledge from literature texts, but also offer tools for more advanced analysis oriented towards improving diagnosis, prevention, and treatment strategies. NLP-based methods enable researchers to conduct not only in-depth studies into specific systems but also facilitate the development of essential real-world software applications [9,10]. Among these applications [11], automatic text summarization has become a pivotal player in NLP. Over the past decade, considerable attention to text summarization has led to notable advances, enabling users to easily extract the main ideas from large volumes of information [12]. The primary objective of automatic text summarization is to provide a condensed version of the original text [13,14], highlighting all essential points while avoiding redundancy [15,16]. This article introduces an integrated model for text summarization and classification, combining advanced NLP techniques with machine learning algorithms. The model first preprocesses large textual datasets and then extracts features.

Methods it uses include NER (Named Entity Recognition), sentiment analysis, Cuephrase, and Part-Of-Speech tagging [17]. Such techniques, in particular, contribute to feature extraction using the Chi-BSC optimizing method for feature selection [18] by identifying key entities, assessing emotional tones, and understanding grammatical roles. These are pivotal for properly refining inputs for both classification and summarization processes

The model handles textual data accurately and efficiently, utilizing Decision Trees, SVC Gradient Boosting, Random Forest, K-Nearest Neighbors (KNN), and Logistic Regression. ROUGE scores, accuracy, precision, recall, and F1-score of data-driven decision-making in healthcare and biomedical research measure summarization and classification [19,20]. ROUGE metrics—ROUGE-1, ROUGE-2, ROUGE-L—measure the overlap between generated summaries and reference texts, making them crucial. This study offers several significant novel contributions to evaluating summarization quality quantitatively:

- First, the model leverages the Chi-BCS method for text feature selection, with a set of machine learning models to recognize the most relevant sentences within a document
- The datasets are adapted for supervised learning by adding a new column (e.g., "is_in_summary") that identifies whether a sentence is included in the summary. This facilitates classification and summarization by prioritizing content through categorized features.
- The model incorporates ensemble learning methods, specifically hard voting, to enhance classification accuracy and achieve greater predictability compared to traditional single-model approaches.

Overall, this model significantly improves the ability to maintain sequence continuity and historical accuracy, addressing many of the primary issues identified in previous research.

1. LITERATURE SURVEY

The major goal of Extractive Text Summarization (ETS) is to categorize sentences in a given text and give labels that demonstrate the degree of importance of the sentence for inclusion in a summary. The approach in [1] improved extractive summarization by efficiently combining higher quality with faster summarization. Specifically, a method relies on a weighted feature score scheme. It was tested using the CNN, XSum, and BBC Summarization datasets and evaluated against various metrics, including ROUGE-1, ROUGE-2, ROUGE-L, and BERTScore.

Authors in [21] employed an advanced method by tuning BERT with additional Transformer layers, which were capped with a sigmoid classifier. This concept, BERT-SUMEXT, was recursively used. Nevertheless, in a more distinct sense, the Transformer encoder was not used as a regular way to encode the entire document [22]; instead, a hierarchical Transformer encoder was developed. This encoder is a dual-purpose encoder, supporting both sentence and document types. They assigned the name HIBERT, and after its completion, the team did a detailed work-up of this engine to suit the requirements of the ETS engines.

One of the problems often faced when dealing with long texts is the issue of length. To address this issue, [23] developed a distinct RNN-based model for ExtSum, specifically a title generator called ExtSum-LG. It is a Natural setting with a combination of both worldwide and local contextual knowledge. However, in the case of quick summaries, these often have the drawback of being too repetitive. Taking into account this fact, the [23] organization continues with the modification of the model by the inclusion of mechanisms to diminish the cliches. They developed a comprehensive set of methods, which involved Trigram Blocking, RdLoss, and MMR-Select, and upgraded MMR-

Select+ algorithms [23]. The fact that trigram blocking is a semi-classical method (essentially classical) aimed at minimizing redundancy makes it important to keep this feature of the method in mind. This mechanism operates by rendering a given sentence unconquerable when it overlaps with a trigram that has already been chosen. The last fact that is impressive to consider is the integration of their model with redundancy-reducing techniques, which currently yield state-of-the-art performance (SOTA) on renowned platforms such as PubMed and arXiv [24].

The networks introduce the next level of computational layers, such as CNN [25], RNN [26], or even Transformer layers [27, 28]. These layers not only establish the required document-level representations for the entire sentence but also utilize the linear connections between sentences. However, a central deficiency of these models, which should be emphasized, is their indirect, implicit, and unclear approach to dissecting hierarchical representations. This operates as a kind of black box where interpretability is transformed into an unwitting victim. Our plan is intended to buck the current trend. We believe every sentence representation must be implemented explicitly. This is done by using section titles and integrating additional HiStruct data, particularly the positioning of sentences in their specific hierarchies. The technique not only promises effectiveness, but it also offers extra features such as efficiency and legibility.

In [29], researchers develop a system for managing internet data growth. Text summarization condenses documents, aiding understanding. Extractive summarization receives focus, employing a supervised approach. Results indicate promise with an average ROUGE-1 score of 0.51.

Researchers in [30] address the challenge of navigating vast amounts of information for decision-making. Text summarization condenses texts, requiring robust algorithms. A study evaluates machine learning models using data from BBC News. An ensemble approach improves summarization accuracy.

ETS is the venue for numerous research projects that have resulted in dozens of research methodologies. In these methods, the framework most commonly used is the encoder-decoder, where the encoder and decoder algorithms are selected. The encoder component of the model requires a variety of available structures, which the researchers have extensively experimented with. According to [31], RNNs demonstrated efficiency at capturing language's unique properties. However, compared with the Transformer architecture (as explained in experiments [32,33]), the BERT model performed well in the area of machine translation. Additionally, [34] introduced GNN as a possible encoding method for practical application in ETS.

With respect to the decoder portion of the model assumed that there are two paths. It has been a topic of debate among researchers; for instance, [35,36] advocate for non-autoregressive decoders. On the contrary, philosophy is behind the anarchic decoding that auto-regressive models are based on, which has been shown. Nevertheless, the dark side of those algorithms, despite their nature as sentence-level extractors, is a vexing conundrum conditioned by their design features. They are custom-made to highlight the top individual sentences and elevate them as better or superior writing. The latter may thus become a trap, since the sentence with the highest standalone score may not be the one that will provide the best summary. For its worth noting that progress has been obvious, with an inclination towards solving the given problem. RL in the summary procedure enables researchers to adjust the summary-level criterion for assessment boards, resulting in a more comprehensive evaluation of work quality. This development, according to [35,36], attests to the momentum of the motive towards the higher quality and condensed creation of extracts.

Scoring-based extractive summarization methods struggle with feature identification for sentence importance. There has been a lack of concerted effort in identifying features that effectively differentiate important sentences from less important ones in document summarization. Existing machine learning approaches in this area have introduced and focused on identifying discriminative features, but it has still limitations in performance of accuracy, precision and generates relevance summary

TABLE I: Review of Related Work in Text Summarization

Study Reference	Method	Contribution	Limitation
[1]	Weighted feature scoring method for extractive summarization	New formula for scoring sentences	Domain-specific, statistical technique for selecting weights for each feature
[22]	BERT fine-tuning with added Transformer layers capped by sigmoid classifier (BERT-SUMEXT)	Improved extractive summarization with enhanced sentence encoding	The transformer encoder is not used for full document encoding

Study Reference	Method	Contribution	Limitation
[23]	Hierarchical Transformer encoder (HIBERT) with separate sentence and document encoders	A dual-purpose encoder enhances ETS by capturing both sentence-level and document-level features.	Complexity increases, potentially affecting scalability.
[24]	RNN-based model (ExtSum-LG) integrating local and global contextual knowledge, with redundancy reduction (Trigram Blocking, RdLoss, MMR-Select, MMR-Select+)	Reduced repetition and enhanced summary quality, achieving SOTA on PubMed and arXiv datasets	May still face challenges with extremely long documents and computational overhead
[29]	Supervised extractive summarization system for managing internet data growth	Demonstrated promising results with an average ROUGE-1 score of 0.51	Limited ROUGE score indicates room for improvement
[30]	Machine learning models evaluation on the BBC news dataset with an ensemble approach	Improved summarization accuracy leveraging ensemble learning	Evaluation limited to the BBC dataset may lack generalizability
[31]	RNNs for encoding language properties in ETS	Showed efficiency in capturing sequential language features	Underperforms compared to Transformer-based architectures
[34]	Graph Neural Network (GNN) for encoding in ETS	Introduced GNN as a promising alternative encoding approach in extractive summarization	The practical application is still in its early stages, with potential scalability issues.

2. PROPOSED METHODOLOGY

The proposed model, depicted in the flowchart in Figure 1, offers a comprehensive framework for text summarization and classification, leveraging a variety of data preprocessing, feature extraction, and machine learning techniques. The process begins with the creation and preparation of data, where articles are sourced from databases like Kaggle and BBC News. An initial data structure is prepared by iterating over articles to populate an empty list with new rows of data. Next, the text data is preprocessed: cleaned and organized in the most effective manner possible. Among them are dropping rows with missing values (NaN); HTML tags and URLs removed; lemmatization conducted to reduce the words to base or dictionary form; the identification of the Word Net positions; the tokenization of each row in the data set, which refers to the breaking down of text into tokens or words; and the elimination of stopwords that add little semantic value to the sentences. After the data is preprocessed, the model extracts the most important features that might help in summarization or classification. It checks whether a sentence from the article is present within the summary sentences. Major features extracted include the length of the sentence, position within the article, named entity recognition for important information, and its classification. Additionally, sentiment analysis is used to infer the tone of the text, a similarity matrix is built to check relationships between sentences, and text rank scores are calculated based on the importance of the sentences. In addition to feature extraction, the model performs Exploratory Data Analysis (EDA) to understand the dataset. This includes the presence of summaries, sentence lengths, data categorization, and sentiment distribution in the text. This EDA helps one fine-tune the data preparation for further classification and summarization. Both in classification and summarization, this data is split into an 80% training set and a 20% testing set. The model includes various machine learning algorithms, such as Decision Trees, Support Vector Classifier, Gradient Boosting, Random Forest, K-Nearest Neighbors, and Logistic Regression, among others, through which it learns the selected features. Indeed, many characteristics of these algorithms can potentially address different aspects of the summarization and classification challenges due to their inherent strengths. The flow chart represents the structural approach to the complex processes involved in text summarization and classification, ultimately leading to actionable and precise summaries and classifications of textual data.

3.1. BBC News Summary Dataset

The BBC dataset is a collection of news articles compiled for an extractive text summarization study covering the time frame between 2004 and 2005. Alongside these, manual curation of five summaries is provided for each article, which gives another dimension to condensation approaches. The first sentence, which represents its title, is a good place for the summary and helps give context to the story [1].

The provenance of the dataset originates from a larger dataset, originally used for research purposes to classify documents into specific topic areas, comprising 2225 documents from five domains collected from the BBC News website. This dataset has been a crucial source of subject matter for academic research, particularly for Greene and Cunningham's work on overcoming the challenges in kernel document clustering at ICML 2006. The BBC continues

to hold copyright over the materials, which can be accessed via their repository that deals with the dataset. Through this, we highlight our ethical issue in research practice and the recognition of intellectual contribution [1].

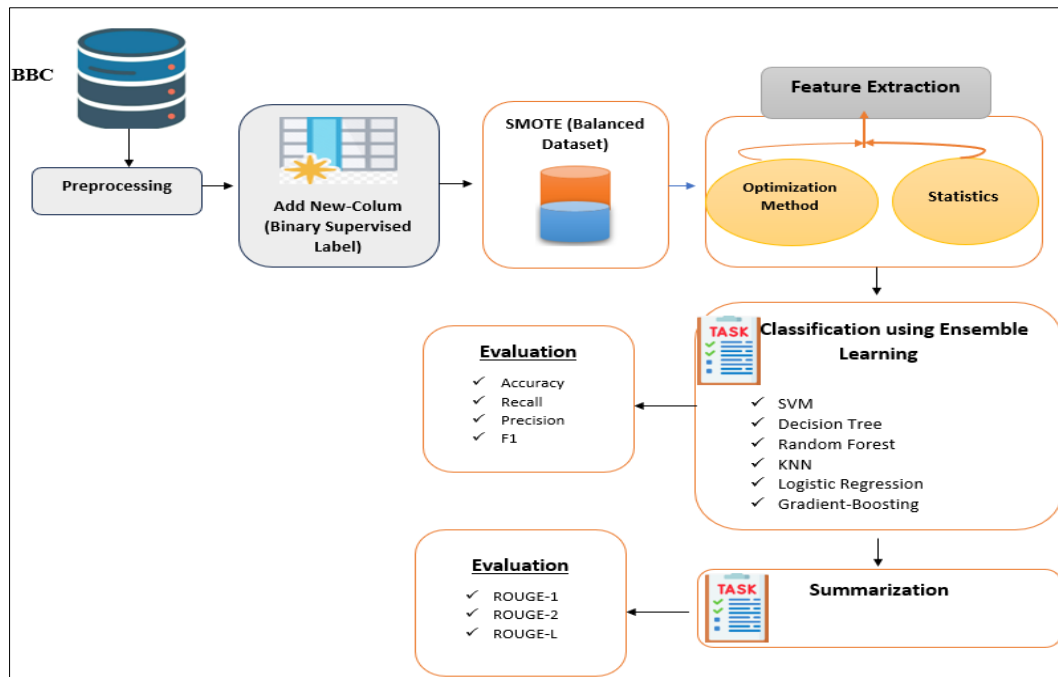


Fig. 1. General structure of hybrid approach for improving text summarization

3.1.1. The Exploratory Analysis for BBC -Dataset

EDA conducted on the dataset is crucial for understanding the distribution and characteristics of the data. Figure. 2 shows the distribution of articles across five categories: entertainment, technology, sports, politics, and business. Figure. 2 shows a balanced distribution between tech and sport, slightly higher than in other categories. Understanding this distribution helps determine if the dataset is balanced across domains, which is necessary for unbiased classification algorithms.

3.2. CNN/Dailymail Dataset

CNN/Daily Mail consists of news articles and their abstractive summaries. ~300,000 CNN & DailyMail articles with multi-sentence human-written summaries. Evaluates models on long documents, coherence, paraphrasing, named entity handling, and factual consistency. We use the standard splits for training, validation, and test (90,266/1,220/1,093 for CNN and DailyMail). This dataset is well-regarded for benchmarking summarization models on complex news content demanding nuanced understanding and summarization quality across multiple dimensions [1,37].

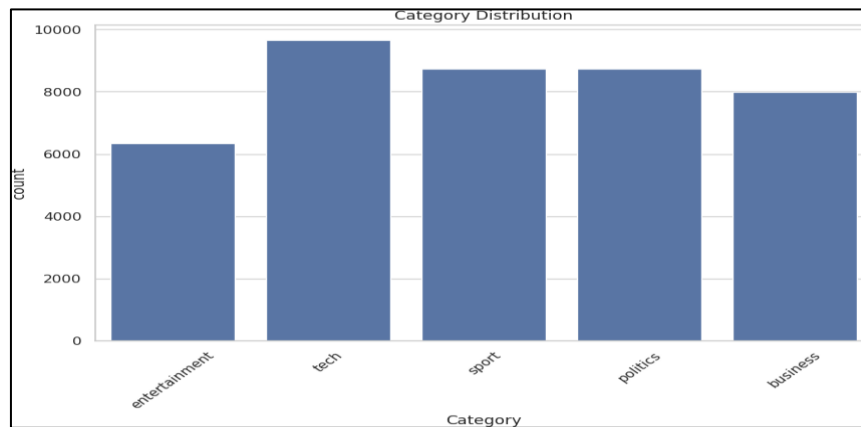


Fig. 2. Category Distribution of BBC dataset.

3.3. Data Preprocessing

Data preprocessing is crucial for text summarization and the construction of classification models. Preprocessing raw textual data is performed before feature extraction and model training in this approach. Preprocessing begins with data gathering. Download both BBC* and CNN/Dailymail** from Kaggle.

When working with raw news articles, the initial text often contains HTML elements (such as <div>, <p>, or <a>) and URLs, which are parts of the webpage formatting rather than meaningful content. These elements can interfere with accurate text analysis for several reasons. First, they use irregular formatting and unusual characters that distort the article. They can also mislead natural language processing algorithms, which prefer clear text over web code. In data preprocessing, HTML tags and URLs are recognised and removed from the text to create a cleaner, more uniform dataset that accurately represents the article's content.

Cleaning is followed by tokenisation. Breaking continuous text into tokens, usually words or subwords, makes it easier to analyse. For instance, "Hellow World" would be tokenised as "Hello" and "World". This allows algorithms to analyse text at a granular level and extract features that study word usage, syntactic structures, and semantic meanings.

Stop words like "the," "is," "and," "in," and others are crucial to tokenisation and preprocessing. These common terms have little semantic substance. Due to their frequency, adding them to the analysis can introduce noise and waste time on unnecessary data. Thus, stop words are deleted to improve analysis quality and speed, allowing algorithms to focus on more informative terms that help them grasp the text.

High-level analysis and effective model training require well-prepared data with properly managed missing values. To maintain data integrity, preprocessing may detect and input or remove missing or invalid (represented as NaN—"Not a Number") elements. Consistently formatted data eliminates disparities from input formats or missing values. Models need homogeneity to learn patterns without being upset by inconsistent or missing input.

Together, these preprocessing steps—cleaning raw text, tokenizing effectively, removing noise, and handling missing data—provide a strong foundation for accurate, efficient, and coherent natural language analysis and summarization tasks.

3.4. Feature Extraction

As the number of characteristics increases, it becomes more challenging to classify data, and machine learning algorithms perform less effectively. The "Curse of Dimensionality," or an increase in data dimensions, causes noisy, irrelevant, and redundant data. Due to model overfitting, this circumstance increases the errors of the learning algorithm. Feature Selection (FS) is a popular Dimensionality Reduction method for these issues. FS simplifies machine learning models, speeding training and enhancing interpretability. Optimal feature selection improves model accuracy and reduces overfitting. The Chi-BCS approach, developed in [18], uses Binary Cuckoo Search, a hybrid Chi-square filter, and a chaotic map to select features. Figure.3 shows the key Chi-BCS-based feature selection steps [18].

Named Entity Recognition (NER), TF-IDF, Sentiment Analysis, Sentence Position, TextRank, Cue phrase, and Part-of-Speech (POS) tagging were chosen for this model. Classification and summarisation require these qualities. Because NER detects critical elements such as persons, organisations, and locations, the model can focus on sentences containing crucial information for both processes. Relevance and document differentiation for TF-IDF classification depend on importance. Sentiment Analysis distinguishes between positive and negative evaluations and summarises emotive sentences. Models benefit from POS and sentence position. Sentence Position highlights a document's opening and closing sentences, which contain key ideas. Grammar and logic are ensured by POS tagging. Text data processing and understanding models benefit from complementary properties. Improves real-world outcomes by improving categorisation and summarisation accuracy and relevance.

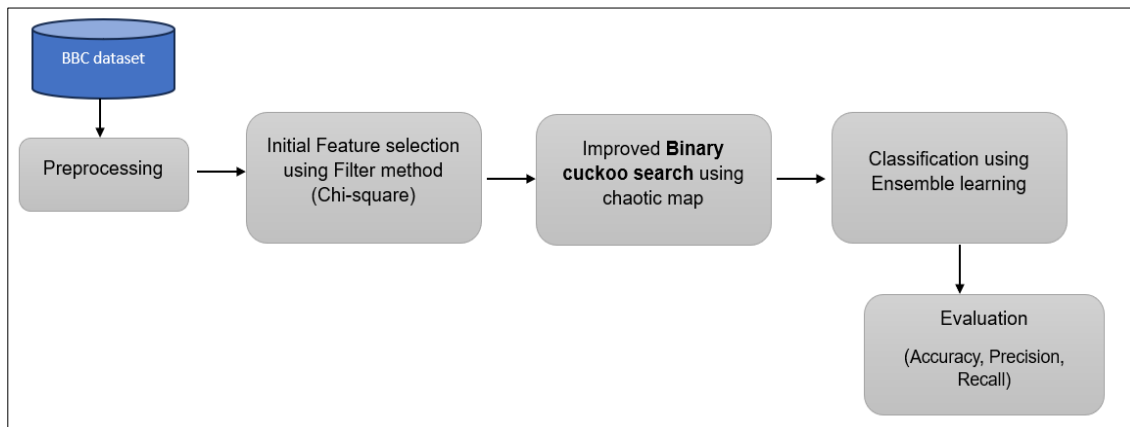


Fig. 3. Feature Selection for Dataset document using Chi-BCS optimization method

The chosen features improve classification and summarisation together. When a person or organisation with emotional significance has a certain emotional tone, Name Entity Recognition and Sentiment Analysis work together. Moreover, Sentence Position and TextRank associations help keep important structural and contextual information in the summary. The model uses these associations to classify more accurately and extract structure, sentiment, and context from the original text. This combination significantly enhances the model's ability to process and interpret textual data.

Algorithm: Chi-BCS for Feature Selection

Input:

- (D) - Labeled dataset
- $(MaxIter)$ - Maximum number of iterations
- (n_{nests}) - Number of nests
- (d) - Number of dimensions (features)
- (p_a) - Probability of abandoning a nest
- (α) - Step size parameter (initial)

Output:

Best feature subset with the highest classification accuracy

Procedure: Chi-BCS

Nest Initialization

- 1.1 For each feature in dataset D , apply the Chi-square filter to obtain a score for each feature.
- 1.2 Calculate the mean or median score across all features.
- 1.3 For each nest:
 - 1.3.1 For each feature in the nest:
 - If feature score $>$ mean/median, set feature = 1 (select feature).
 - Else, initialize randomly with either 0 (not selected) or 1 (selected).
 - 1.3.2 Evaluate the classification accuracy of the feature subset represented by the nest.
 - 1.3.3 Store the fitness score (classification accuracy) for each nest.

Local Search (Lévy Flight)

- 2.1 Set iteration counter $t=0$.
- 2.2 While $t < MaxIter$ or (termination criterion is not met):
 - 2.2.1 For each nest:
 - Perform Lévy flight to generate a new solution based on step size α .
 - Convert the new solution to binary format (0 or 1) using a threshold.
 - Train the classifier using the selected feature subset and evaluate accuracy.
 - If new accuracy $>$ current best accuracy for the nest:
 - * Update nest solution with new feature subset.
 - * Update fitness score for the nest.

- 2.2.2 Update α using chaotic sinusoidal map:
 $\alpha \leftarrow m \cdot \alpha \cdot \sin(\pi \cdot \alpha)$ (where m is a constant, e.g., 0.5)
 Abandonment of Worst Nests
 3.1 Identify and abandon a fraction p_{abn} of the worst-performing nests.
 3.2 Replace abandoned nests with new solutions generated through Chi-square-based initialization.
 3.3 Sort nests by fitness score and keep the top-performing solutions.
 Update Iteration
 4.1 Increment iteration counter: $t=t+1$
 Final Selection
 5.1 Identify the nest with the highest fitness score.
 5.2 Return the feature subset represented by the best-performing nest.
 End Procedure

3.5. Integrated Text's Features for Sentence Labelling

After feature selection for sentences, the next step is typically to perform sentence labeling in a binary classification setup for summarization tasks. This means using the integrated feature vectors to classify each sentence as either summary-worthy (label 1) or non-summary (label 0). Labeling sentences is essential because it directly determines which sentences will be selected to form the extractive summary. The final vector, some of which is shown in Table II, is a collection of diverse high-level attributes aggregated into a comprehensive portrait of the source data text at the sentence level. The first dimension, Article_ID, represents a sentence-by-document link and serves as the primary identifier for the grouping measure. The sentence, a second dimension, is defined as a string of text in focus, whereas Is_in_Summary is a binary flag indicating whether the sentence was included in the summary, thereby indicating its importance. Sentence_Length, a fourth dimension, expresses the length of the sentence as a word count used to assess its verbosity or conciseness. Sentence_Position, a fifth dimension, is a measure of the sentence's relative position in the set, a frequently applicable factor of its relative importance. Further granularity is supplied via Named_Entity_Recognition, a sixth dimension, which accounts for the number of named entities in the string, enhancing semantic analysis. POS_Tags, a seventh dimension, reference the sentence's grammatical makeup through assigned Parts of Speech tags, aiding syntactic analysis. Sentiment Analysis, the eighth dimension, provides a sentiment score that measures emotional tone. Cuephrase, the ninth, acts as a signal or guide within a text to help the reader recognize important information. TextRank scores consist of numerical, non-linear weights that assess the significance of a sentence, hence the count-based Average TextRank Score dimension. TFIDF_Score, a thirteenth attribute, expresses the sentence's term gravity weight within the document collection. Cue phrases, the sentence that contains significant information with the phrases 'In the abstract', 'In conclusion', etc. These dimensions result in a comprehensive sentence vector, generating discriminative features for processing in machine learning, natural language processing, and other advanced fields. Table II. shows the details of feature extraction for each sentence after preprocessing.

TABLE II: Scoring the features for each sentence in BBC dataset documents

Article ID	Sentence	Is_in_Summary	Sentence-Length	Sentence-Position	Name Entity Recognition	POS-Tag
Article 0	Indie film nomination announce mike.....	1	22	0.014	4	22
Article 0	Venice winner faces stiff competition shane.....	1	19	0.0588	4	19
Article 0	Summer love Stalker drama...	0	11	0.117	2	11

After establishing the summarized status of sentences, it determines which sentence is more critical. The scores eventually provide a comprehensive understanding of the importance of a sentence, from which extractive summarization can now be used to extract useful details for downstream applications, such as document summarization and information extraction.

3.6. Preparation Data for Classification and Summarization

Analyzing the dataset's class distribution before and after applying the Synthetic Minority Over-sampling Technique (SMOTE) is crucial for understanding its distribution and characteristics across different dimensions. Fig.4, a, b,

illustrate the class before and after applying SMOTE. Initially, the class distribution is highly imbalanced, as shown in Figure 4a, with a significant majority in one class compared to the other. Fig. 4b shows the effect of applying SMOTE, which is used to artificially balance the dataset by increasing the representation of the minority class through synthetic data generation. This balancing is vital for training models that perform well across all classes, preventing biases towards the majority class.

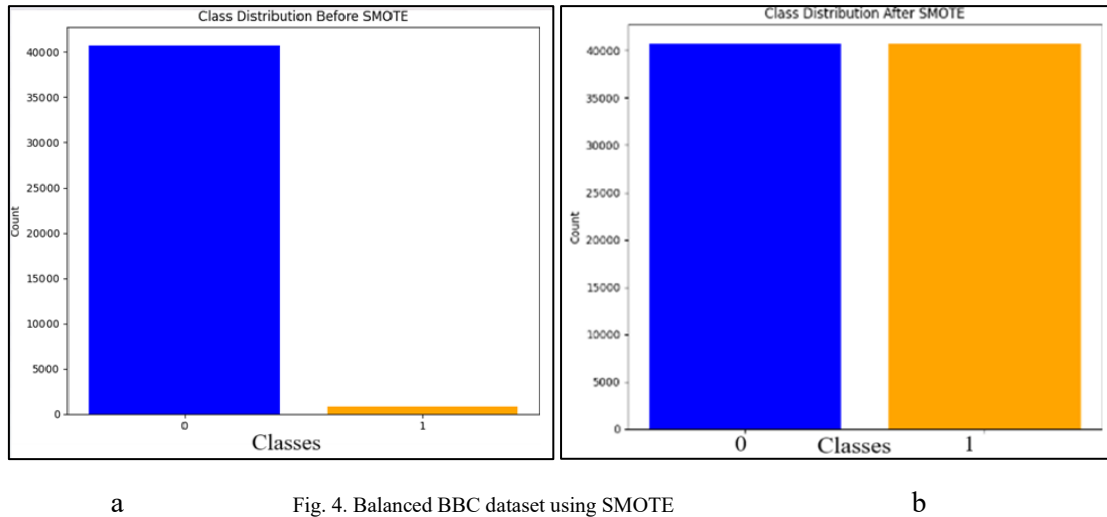


Fig. 4. Balanced BBC dataset using SMOTE

3.7. Machine Learning Models

Classification and summarization of textual data can be easily accomplished with the aid of machine learning methods. In this section, the procedure of selecting specific algorithms used in the given research, along with their applicability for analyzing large volumes of data, is discussed.

Random Forest is one of the ensemble methods that constructs a very large number of decision trees, and the final decision is made by aggregating those results with a voting system, which reduces the overfitting problem. This technique proves to be quite useful when handling various and large data attributes characteristic of textual datasets[38].

SVC stands for Support Vector Classifier, is a type of machine learning algorithm. It is a core component of Support Vector Machines (SVMs) [18], The fundamental idea behind SVC is to find a hyperplane (a decision boundary) that optimally separates data points belonging to different classes in a high-dimensional space.

Logistic Regression, despite its name, which may suggest simple, proves to be a powerful model for binary classification. This model employs a logistic function to identify the possibility of a binary outcome contingent on the input variables, especially when it is suitable for categorizing data into two different sets [38].

Decision Trees are decision-making diagrams analogous to flowcharts, with feature splits used to categorize instances by breaking down the data in a manner of recursive partitions. The technique is applicable because it enables the revelation of the decision-making process and the rationale behind the decisions made in these fields [39].

In addition, the Ensemble Learning method integrates several models, such as Random Forest, Decision Tree, and K-Nearest Neighbors (KNN), among others, by employing the majority voting feature. This approach is a type of decision-making that combines the results of various algorithms to make more efficient and accurate decisions. When applying this combination of promising methods in the field of machine learning, the presented models are able to overcome common obstacles in text classification and text summarization, thereby enhancing the efficient identification of essential information and improving problem-solving in various fields [39].

Table III provides a concise summary of the classification models employed in our experiment, along with their corresponding parameters. Each method is accompanied by a brief description elucidating its key characteristics and utility in the context of text classification and summarization tasks.

TABLE III. Summary of Classification Models and Parameters

Method	Short Description	Parameters
<i>Random Forest</i>	Ensemble method that uses a combination of decision trees to improve predictions and control overfitting.	random_state=18
<i>SVC</i>	Effective in high-dimensional spaces.	random_state=18, probability=True
<i>Logistic Regression</i>	A statistical model that, in its basic form, uses a logistic function to model a binary dependent variable.	random_state=18, max_iter=1000
<i>Decision Tree</i>	A decision support tool that uses a tree-like model of decisions and their possible consequences.	random_state=18
<i>Gradient Boosting</i>	A machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models.	random_state=18
<i>KNN</i>	A type of instance-based learning or non-generalizing learning.	n_neighbors=18
<i>Ensemble</i>	Combines predictions from multiple models to improve accuracy. Uses majority voting.	voting='hard'

3.8. Integration of Classification and Summarization

By creating a secondary relationship between categorisation and summarisation, the approach seeks to merge them. By determining what should be included in a brief summary, classification enhances summarization. Decision Trees, SVC, Random Forests, and Gradient Boosting classify textual data by sentiment, named entities (people, organisations, and geolocations), and sentence position. These characteristics will help the summarisation task choose contextually relevant and theoretically accurate statements. Sentiment analysis identifies texts by emotion, providing effective summaries. However, Named Entity Recognition enables the model to identify sentences containing important person or theme data. Sentence placement also marks the beginning or ending of sentences that often contain key ideas, helping the model choose sentences that summarize quickly but thoroughly.

Classification also enhances summarization by condensing lengthy or redundant text into concise, relevant points, thereby improving efficiency and accuracy. However, an algorithm can neutralize sounds made by itself and highlight the most crucial facts for effective categorization if the text is still in two sentences. Furthermore, by classifying text before summarization, the model can quickly access summaries that are tailored to specific interests or needs. This is crucial in environments with vast amounts of data, such as academic or news datasets. This way, the effect ensures that classification becomes guided toward summarization, and this combination lifts the bar on both generating summaries and categorizing into a top-quality form, raising overall performance levels. This model works brilliantly for real-world applications, such as processing very large volumes of text efficiently and with high accuracy.

4. RESULTS AND DISCUSSION

4.1. Evaluation Metric

Evaluation of binary classification models, such as those used for summarization tasks where phrases are classified as present (1) or absent (0), relies on a variety of metrics to assess performance accurately. In the context of summarization, "0" indicates the non-existence of a phrase in the summary, while "1" signifies its presence. The primary metrics used for this purpose include accuracy, precision, recall, and F1-score, as depicted in Table IV.

TABLE IV. Performance Metrics for Binary Classification

Metric	Definition	Equation
Accuracy [40]	The fraction of correct predictions	$\frac{TP+TN}{TP+TN+FP+FN}$ (1)
Precision [40]	The fraction of true positives among positive predictions	$\frac{TP}{TP+FP}$ (2)
Recall [40]	The fraction of true positives among actual positives	$\frac{TP}{TP+FN}$ (3)
F1-score [1,40]	The harmonic means of precision and recall	$\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$ (4)

Evaluation of text summarization models is typically done by comparing the generated summaries against their corresponding reference summaries. ROUGE (Recall-Oriented Understudy for Gisting Evaluation), which counts overlapping n-grams to compare generated and reference summaries, is the standard measure. At the unigram, bigram, and longest common subsequence levels, ROUGE-1, ROUGE-2, and ROUGE-L scores measure the similarity

between the generated and reference summaries. Precision, recall, and F-measure are calculated from the number of overlapping n-grams in the generated and reference summaries and their total number [41].

Rouge-1:

$$P = \frac{\text{number of overlapping unigrams}}{\text{number of unigrams in the predicted summary}} \quad (5)$$

Rouge-2:

$$P = \frac{\text{number of overlapping bigrams}}{\text{number of bigrams in the predicted summary}} \quad (6)$$

Rouge-L:

$$P = \frac{\text{length of longest common subsequence}}{\text{number of words in the predicted summary}} \quad (7)$$

4.2. Performance Evaluation for Classification Models

Based on the results provided in Table V and Figures 5 and 6, this section presents a detailed discussion of the performance metrics for the machine learning models used for binary classification on the BBC and CNN/Daily Mail datasets.

4.2.1. BBC Dataset Analysis

Random Forest performed well on the BBC dataset, with high precision, recall, and F1-scores for both classes. Precision and recall were well balanced with class 0 and class 1 F1-scores of 0.9671 and 0.9685, respectively. The Decision Tree and KNN models performed well, although less than the Random Forest and Ensemble models.

In contrast, the Gradient Boosting model had a lower performance, with F1-scores of 0.7741 and 0.7944. This suggests it struggled more with the classification task compared to the other models.

However, the Logistic Regression model had the weakest performance on the BBC dataset, with F1-scores hovering around 0.60 for both classes. This indicates that a simpler linear model was not as suitable for this specific task.

Finally, the Ensemble (Hard-voting) model also showed similar top-tier performance, with F1-scores of 0.9708 and 0.9717 for classes 0 and 1, respectively. This suggests that combining the outputs of multiple models via hard-voting was an effective strategy.

4.2.2. CNN/Dailymail Dataset Analysis

Similar to the BBC dataset, the Random Forest model was the top performer on the CNN/Dailymail dataset, with F1-scores of 0.9322 and 0.9389.

The Gradient Boosting and Decision Tree models also performed reasonably well, with F1-scores in the high 0.80.

The KNN model exhibited similar performance, with F1-scores of 0.8897 and 0.8900, indicating it was a solid, if not outstanding, choice.

The Logistic Regression model performed worse again, with F1-scores of 0.8585 and 0.8707. Better than its BBC dataset performance, it still ranks last for this dataset, illustrating its limits for this task.

The ensemble (Hard-voting) model ranked second with F1-scores of 0.9270 and 0.9330. This supports the idea that Random Forest and ensemble methods are effective for this categorisation.

The BBC and CNN/Dailymail datasets showed that the Random Forest and Ensemble (Hard-voting) models performed best. This shows that non-linear, tree-based approaches and multiple models work well for binary classification. Decision Tree, Gradient Boosting, and KNN performed well but were less dependable.

The Logistic Regression model consistently underperformed compared to the other methods, indicating that a simple linear classifier is likely not powerful enough to capture the complexities of the data in either dataset. This suggests that for extractive summarization as a binary classification problem, more complex models are necessary to achieve high performance.

In general, Figure 5 and 6 serve as a quick summary and comprehensive performance update for both dataset, allowing for an easy and structured comparison of how models perform with the given type of classification.

TABLE V. Performance Metrics for Machine Learning Models based on their classes for BBC dataset

Models	Dataset	Class	Precision	Recall	F1-score
Random Forest	BBC	0	0.9828	0.9519	0.9671
		1	0.9539	0.9835	0.9685
Logistic Regression		0	0.6053	0.6098	0.6075
		1	0.6116	0.6071	0.6093
Decision Tree		0	0.9474	0.9374	0.9424
		1	0.9388	0.9486	0.9437
Gradient Boosting		0	0.8091	0.7420	0.7741
		1	0.7643	0.8270	0.7944
KNN		0	0.9187	0.9008	0.9097
		1	0.9039	0.9212	0.9124
Ensemble (Hard-voting)		0	0.9803	0.9614	0.9708
		1	0.9626	0.9809	0.9717
Random Forest	CNN/Dailymail	0	0.9638	0.9026	0.9322
		1	0.9120	0.9675	0.9389
Logistic Regression		0	0.8804	0.8377	0.8585
		1	0.8514	0.8909	0.8707
Decision Tree		0	0.8891	0.8802	0.8846
		1	0.8863	0.8947	0.8905
Gradient Boosting		0	0.9191	0.8562	0.8865
		1	0.8707	0.9278	0.8983
KNN		0	0.8724	0.9076	0.8897
		1	0.9079	0.8727	0.8900
SVC		0	0.9126	0.8597	0.8853
		1	0.8726	0.9211	0.8962
Ensemble (Hard-voting)		0	0.9483	0.9066	0.9270
		1	0.9141	0.9526	0.9330

4.3. Performance Evaluation for Extractive Summarization Text

4.3.1. BBC dataset

The comparison of the proposed model with state-of-the-art methods reveals notable insights into the effectiveness of different approaches for text summarization. Table VI provides a comprehensive summary of the ROUGE metrics for different models evaluated in the context of text summarization. Firstly, the Supervised Approach, as presented in 29, demonstrates moderate performance across ROUGE-1, ROUGE-2, and ROUGE-L metrics, with F1 scores ranging from 0.5112 to 0.583. In contrast, the Ensemble Approach introduced in 30 achieves significantly higher scores across all metrics, with F1 scores ranging from 0.76 to 0.83. This improvement highlights the efficacy of ensemble learning in enhancing summarization accuracy.

Notably, the proposed Ensemble method with Hard Voting surpasses all other models in terms of summarization accuracy. With F1 scores of 0.83 for ROUGE-1, 0.799 for ROUGE-2, and 0.818 for ROUGE-L, this approach achieves the highest overall performance. This outcome underscores the effectiveness of ensemble learning techniques, particularly Hard Voting, in leveraging the strengths of multiple models to produce superior summarization results. Thus, the Ensemble (Hard Voting) method stands out as a promising approach for text summarization, offering significant enhancements over existing methods and underscoring its potential for real-world applications.

Classification helps categorize and segment material, which improves summarization by highlighting pertinent content within each category. The combined method produces a more focused and relevant summary.

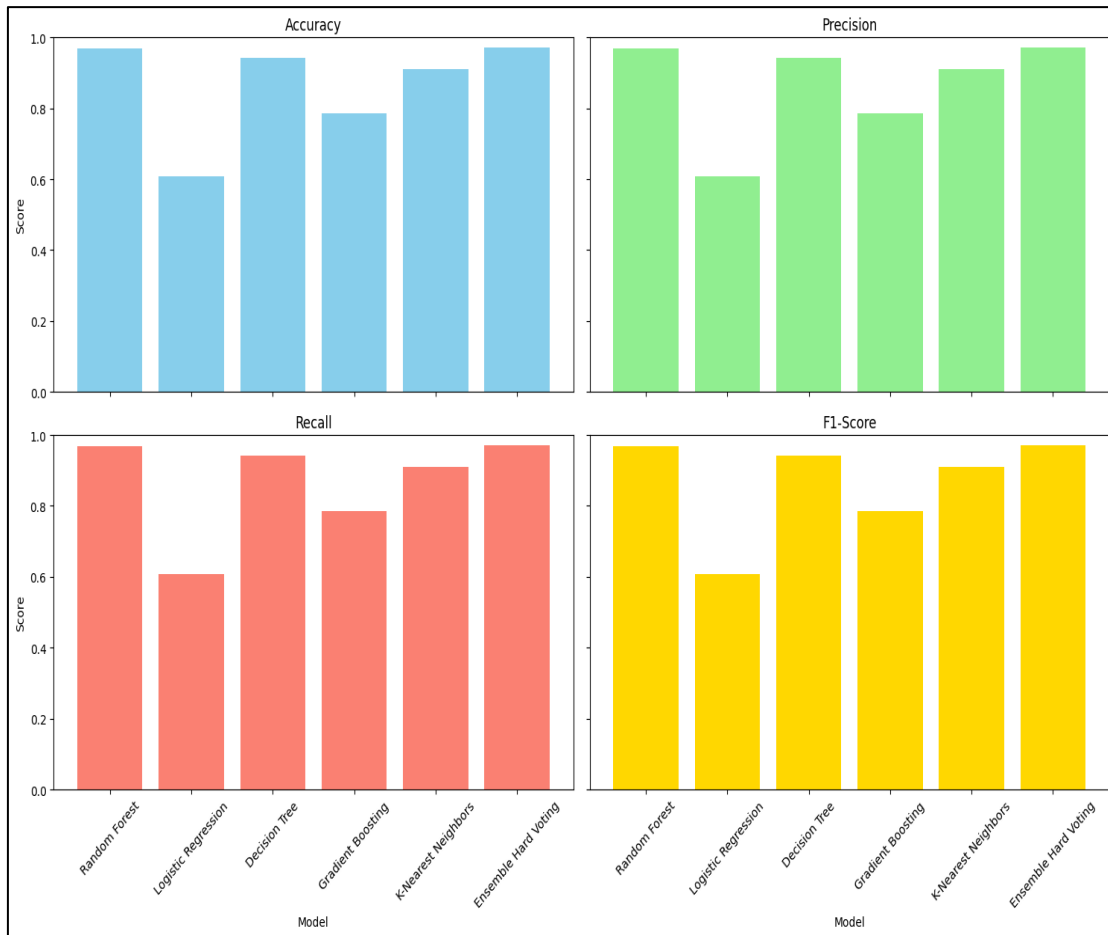


Fig. 5. Overview and visual comparison of various model terms (Accuracy, Precision and F1) score for BBC news dataset

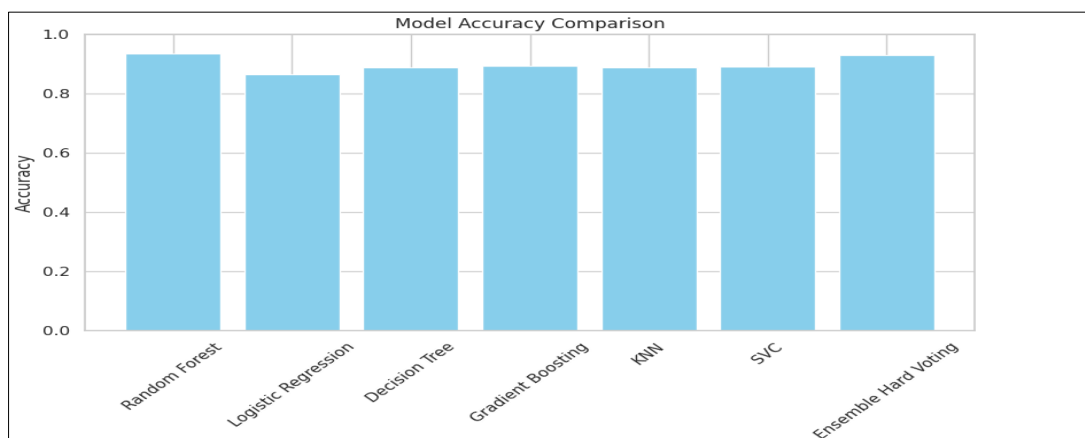


Fig. 6. Overview and visual comparison of various model terms (Accuracy,) score for CNN/DailyMail dataset

TABLE VI. Evaluation various Machine Learning models for summarization performance on BBC Dataset

Models	Metrics	ROUGE-1	ROUGE-2	ROUGE-L
Random Forest	Precision	0.992	0.946	0.971
	Recall	0.665	0.634	0.651
	F1-score	0.796	0.759	0.78
Logistic Regression	Precision	0.976	0.66	0.699
	Recall	0.042	0.029	0.03
	F1-score	0.081	0.055	0.058
Decision Tree	Precision	0.975	0.872	0.899
	Recall	0.603	0.539	0.556
	F1-score	0.746	0.666	0.687
Gradient Boosting	Precision	0.977	0.615	0.645
	Recall	0.064	0.041	0.043
	F1-score	0.121	0.076	0.08
KNN	Precision	0.994	0.937	0.973
	Recall	0.469	0.442	0.459
	F1-score	0.637	0.601	0.624
SVC	Precision	0.987	0.748	0.794
	Recall	0.033	0.025	0.027
	F1-score	0.064	0.048	0.051
Supervised Approach in [29]	Precision	0.597	0.470	0.583
	Recall	0.488	0.368	0.477
	F1-score	0.511	N/A	N/A
Ensemble Approach in [30]	Precision	0.83	0.72	0.79
	Recall	0.74	0.61	0.74
	F1-score	0.78	0.66	0.76
Our proposed Ensemble (Hard-Voting)	Precision	0.993	0.956	0.979
	Recall	0.713	0.686	0.702
	F1-score	0.83	0.799	0.818

4.3.2. CNN/DailyMail Dataset

Text summarization exhibits a distinct performance profile on the CNN/DailyMail dataset compared to other state-of-the-art models, as shown in Table VII. In ROUGE-1 & ROUGE-2 Performance, Our Ensemble Method scores 0.315 for ROUGE-1 and 0.275 for ROUGE-2. These are notably lower than other models like HAHSumLarge (ROUGE-1: 0.4468, ROUGE-2: 0.2130) and MATCHSUM (RoBERTa-base) (ROUGE-1: 0.4441, ROUGE-2: 0.2086). This suggests that your method's summaries may not capture as many exact keywords or short phrases from the reference summaries as the other models. In contrast, the proposed ensemble method achieves a superior ROUGE-L score of 0.5017, which is substantially higher than all other listed models (e.g., HAHSumLarge at 0.4075, BERTEXT at 0.392). This indicates that our method produces summaries with a much longer common subsequence of words, implying better. At the sentence level. This is a crucial advantage for human readability and overall summary quality, even if individual word overlap is lower.

TABLE VII: Evaluation various Machine Learning models for summarization performance on CNN/Dailymail Dataset

Models	ROUGE-1	ROUGE-2	ROUGE-L
HAHSumLarge [42]	0.4468	0.2130	0.4075
PNBERTBase(2019a)[43]	0.4269	0.1960	0.3885
BERTEXT in [42]	0.4273	0.2013	0.392
MATCHSUM(RoBERTa-base) in [43]	0.4441	0.2086	0.4055
Our proposed Ensemble Method on CNN dataset	0.315	0.275	0.5017

The findings showed that among the features under study, the significant ones were Sentence Position, Sentiment Analysis, Cuephrase, and Named Entity Recognition. Sentence Position targets structurally important sentences, such as introductions and conclusions, which typically summarize the vital information contained within them. Sentiment Analysis separates out positive and negative tones, thus dividing texts with varying tones. On the other hand, NER extracts the key entities that clearly define what an article is about, helping to label large datasets based on names, locations, and other important details. The combination of these features with machine learning algorithms, such as

Random Forest and Gradient Boosting, significantly improves classification metrics, including accuracy and F1-score, enabling the model to distinguish between different categories efficiently. The novel fusion of classification and summarization, combined with advanced NLP features, brought a significant improvement in model performance. The model achieved relatively high precision and recall numbers (97.1241%) by using hard voting in ensemble learning for classification. This credit goes to the inclusion of a variety of features, such as Longest Named Entity, Sentiment Analysis, and Sentence Position, which enabled it to discover more profound patterns. In summary, the model produced high ROUGE scores (0.83 for ROUGE-1, 0.818 for ROUGE-L), indicating that it can effectively classify long sentences as a context based on two senses of the words. These innovations, specifically the use of ensemble learning and more robust feature selection, inevitably played a major role in improving upon earlier methods of model processing.

4.4. Preference Machine Learning on Deep learning

Merging Robust linguistic features with machine learning techniques is significantly advancing. However, this comes with drawbacks in terms of time-consuming, storage, and computational speed. Therefore, one of the research objectives is to resort to accurate, efficient, and computationally simple methods. classical ML methods are computationally simpler and require fewer data points for training ,they are particularly suitable for long texts that require considerable time and processing precision if deep learning or transformer models like (BART and T5) approaches are used.

5. CONCLUSION

Today, nearly every industry generates a substantial amount of textual data, and automatic summarization helps manage it effectively. This paper presents a hybrid optimization approach integrating machine learning techniques for extractive text summarization. By employing advanced feature extraction methods such as Named Entity Recognition, Sentiment Analysis, and Chi-BCS optimization for feature selection, the model effectively captures semantic, syntactic, and contextual attributes of sentences. The application of Synthetic Minority Over-sampling Technique (SMOTE) successfully addresses class imbalance, enhancing classifier robustness. An experimental evaluation on benchmark datasets, including BBC and CNN/Dailymail, demonstrates the proposed ensemble (hard-voting) model's superior performance. For classes 0 and 1, the ensemble had F1-scores of 0.9708 and 0.9717 on the BBC dataset, with precision and recall above 0.96. The ensemble had F1-scores of 0.9270 and 0.9330 for the two classes on the CNN/Dailymail dataset. The model also achieves good ROUGE metric scores, with 0.993 precision and 0.979 F1-score on ROUGE-1 for the BBC dataset, demonstrating accurate and meaningful summarization. Hierarchical sentence modelling, heuristic optimisation, and ensemble learning are effective approaches for extractive summarisation. The work lays the groundwork for larger datasets, more complicated neural architectures, multi-domain flexibility, and multilingual capabilities.

6. FUTURE WORK

Here are potential future work directions for the paper:

- Utilize the proposed model as a pre-processing phase for abstractive summarization to enhance the semantic meaning of the output text summary.
- Future development could involve extending this model to handle more complex neural architectures and broader datasets. This would allow for greater generality and applicability across diverse and multilingual information.
- Advanced Unsupervised Learning Approaches: Exploring advanced unsupervised learning approaches could reveal deeper insights from unlabelled data. This would make the model more applicable and efficient in situations where labelled data is scarce or expensive.
- Application to Diverse Languages and Topics: Future studies can aim to refine the model and apply it to a wider range of languages and topics beyond the current scope. This efficiency is particularly beneficial for healthcare providers who need to make rapid decisions based on current research findings. By providing concise summaries that highlight crucial data, extractive systems enhance decision-making processes in clinical settings.
- Used proposed model as phase pre process for abstractive top phase to support the semantic meaning for input text

Conflict-of-interest Statement



We declare that we have no financial and personal relationships with other people or organizations that could inappropriately influence our work.

References

- [1] Kadhim EA, Feizi-Derakhshi MR, Aghdasi HS, "Advanced Text Summarization Model Incorporating NLP Techniques and Feature-Based Scoring". IEEE Access, vol. PP, no. 99, pp. 1–1, Jan. 2025.
- [2] Azam M, Khalid S, Almutairi S, Khattak HA, Namoun A, Ali A, Bilal HS, "Current Trends and Advances in Extractive Text Summarization: A Comprehensive Review", IEEE Access, vol. 13, pp. 28150-28166, 2025.
- [3] E. A. Kadhim, M.-R. Feizi-Derakhshi, and H. S. Aghdasi, "NIASM: Non-idiomatic abstractive summarization model to reduce hallucination and factual inconsistency problem in text generation, submitted to IEEE ACCESS, 2025.
- [4] Nadif M, Role F., "Unsupervised and self-supervised deep learning approaches for biomedical text mining", Briefings in Bioinformatics. vol. 22, pp. 1592–1603, 2021.
- [5] Dash S, Acharya BR, Mittal M, Abraham A, Kelemen A, editors, "Deep learning techniques for biomedical and health informatics. Cham, Switzerland", Springer International Publishing, Volume 13, Issue 6, 2573 – 2589, Nov, 2022.
- [6] Bernanda DY, Jawawi DN, Abd Halim S, Adikara F., "Natural Language Processing", For Requirement Elicitation In University Using Kmeans And Meanshift Algorithm. Baghdad Science Journal, Vol. 21: Iss. 2, Article 29, Feb, 2024.
- [7] Flayeh AK, Hamodi YI, Zaki ND, "Text Analysis Based on Natural Language Processing (NLP)", 2nd International Conference on Advances in Engineering Science and Technology (AEST)", Babil, Iraq, pp. 774-778, 2022.
- [8] Bani-Almarjeh M, Kurdy MB, "Arabic abstractive text summarization using RNN-based and transformer-based architectures", Information Processing & Management". 1;60(2):103227, Mar 2023.
- [9] Boukhlif M, Hanine M, Kharmoum N, Noriega AR, Obeso DG, Ashraf I, "Natural Language Processing-based Software Testing: A Systematic Literature Review", IEEE Access. vol. 12, pp. 79383-79400, 2024.
- [10] Kang Y, Cai Z, Tan CW, Huang Q, Liu H, "Natural language processing (NLP) in management research: A literature review", Journal of Management Analytics", 2;7(2):139-72. Apr, 2020.
- [11] Allen LK, Creer SD, Poulos MC, "Natural language processing as a technique for conducting text-based research", Language and Linguistics Compass";15(7):e12433. Jul, 2021.
- [12] Shahade AK, Deshmukh PV, "A Unified Approach to Text Summarization: Classical, Machine Learning, and Deep Learning Methods", Ingénierie des Systèmes d'Information" Jan 1;30(1), Jan, 2025.
- [13] Zhang H, Liu X, Zhang J. "Extractive summarization via chatgpt for faithful summary generation". arXiv preprint arXiv:2304.04193. Apr, 2023.
- [14] Saiyyad MM, Patil NN., "Text Summarization Using Deep Learning Techniques: A Review. Engineering Proceedings", 19;59(1):194, Jan, 2024.
- [15] Rajesh B, Chaitanya KN, Govardhan PT, Mahesh KK, Sudarshan B. "Text summarization using NLP", Int Res J Eng Technol (IRJET)." Int. j. inf. tecnol. 14, 2407–2415 (2022).
- [16] Wib Supriyono, Aji Prasetya Wibawa, Suyono, Fachrul Kurniawan, "A survey of text summarization: Techniques, evaluation and challenges", Natural Language Processing Journal", ISSN 2949-7191, Volume 7, 100070, 2024.
- [17] Goyal N, Singh N., " Named entity recognition and relationship extraction for biomedical text: A comprehensive survey, recent advancements, and future research directions", Neurocomputing, Vol- 618, 2025, 129171, ISSN 0925-2312, 2025.
- [18] M. Feizi-Derakhsh MR, Kadhim EA., " An Improved Binary Cuckoo Search Algorithm for Feature Selection Using Filter Method And Chaotic Map", Journal of Applied Science and Engineering; 26(6):895, Sep, 2022.
- [19] Mijwil MM, Aljanabi M, " A comparative analysis of machine learning algorithms for classification of diabetes utilizing confusion matrix analysis", Baghdad Science Journal, Vol. 21: Iss. 5, Article 24, 2024.



- [20] Nafea AA, Manar AM, Alheeti KM, Alsumaidaie MS, AL-Ani MM. , "A hybrid method of 1D-CNN and machine learning algorithms for breast cancer detection", Baghdad Science Journal. 2024 Oct 1;21(10):3333.
- [21] Liu J. Hadi, Mustafa; Abbas, Ayad R.; and Fadhil, Osamah Y., "A Novel Gravity Optimization Algorithm for Extractive Arabic Text Summarization", Baghdad Science Journal, Vol. 21: Iss. 2, Article 6. 2024.
- [22] Zhang X, Wei F, Zhou M. , "HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization", arXiv preprint arXiv:1905.06566. May, 2019.
- [23] Xiao W, Carenini G., " Extractive summarization of long documents by combining global and local context", arXiv preprint arXiv:1909.08089. Sep, 2017.
- [24] Paulus R, Xiong C, Socher R., "A deep reinforced model for abstractive summarization", arXiv preprint arXiv:1705.04304. May, 2017.
- [25] P. Sharma and M. Chen, "Text Summarization and Keyword Extraction", 14th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), Koriyama, Japan, pp. 369-372, 2023.
- [26] Nallapati R, Zhai F, Zhou B. Summarunner: "A recurrent neural network based sequence model for extractive summarization of documents", In Proceedings of the AAAI conference on artificial intelligence, Vol. 31, No. 1, 2023.
- [27] Zhou Q, Yang N, Wei F, Huang S, Zhou M, Zhao T , "Neural document summarization by jointly learning to score and select sentences", arXiv preprint arXiv:1807.02305, Jul, 2018.
- [28] Zhong M, Wang D, Liu P, Qiu X, Huang X. , "A closer look at data bias in neural extractive summarization models", arXiv preprint arXiv:1909.13705, Sep, 2019 .
- [29] Krishnan D, Bharathy P, Venugopalan M., "A supervised approach for extractive text summarization using minimal robust features", In 2019 International Conference on Intelligent Computing and Control Systems (ICCS), IEEE, 15 (pp. 521-527), May, 2019.
- [30] Singh P, Chhikara P, Singh J. "An ensemble approach for extractive text summarization", International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), IEEE, 24 (pp. 1-7) , Feb, 2020.
- [31] Wang D, Liu P, Zhong M, Fu J, Qiu X, Huang X. , "Exploring domain shift in extractive text summarization", arXiv preprint arXiv:1908.11664, Aug, 2019 .
- [32] Wang D, Liu P, Zheng Y, Qiu X, Huang X. , "Heterogeneous graph neural networks for extractive document summarization", arXiv preprint arXiv:2004.12393. Apr 2020.
- [33] Narayan S, Cohen SB, Lapata M., "Ranking sentences for extractive summarization with reinforcement learning", arXiv preprint arXiv:1802.08636. Feb 2018.
- [34] Arumae K, Liu F. "Reinforced extractive summarization with question-focused rewards", arXiv preprint arXiv:1805.10392. May 2018.
- [35] Jadhav A, Rajan V, "Extractive summarization with swap-net: Sentences and words from alternating pointer networks", In Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: Long papers), (pp. 142-151), Jul 2018.
- [36] Bae S, Kim T, Kim J, Lee SG. "Summary level training of sentence rewriting for abstractive summarization", arXiv preprint arXiv:1909.08752. Sep 2019.
- [37] D. Chen, J. Bolton, and C. D. Manning, "A thorough examination of the CNN/Daily Mail reading comprehension task," arXiv preprint arXiv:1606.02858, 2016.
- [38] Farhadi, Z., Bevrani, H., & M. Feizi-Derakhshi, M. R. "Improving random forest algorithm by selecting appropriate penalized method", Communications in Statistics - Simulation and Computation, 53(9), 2022.
- [39] Mohammad Reza Feizi Derakhshi, Nasim Vafsi Sisi, "Text classification with machine learning algorithms", J. Basic. Appl. Sci. Res., 3(1s)31-35, March 2013.
- [40] Taha MA, Jabar HD, Mohammed WK. , "Fake news detection model basing on machine learning algorithms", Baghdad Science Journal, 1;21(8):2771, Aug, 2024.



- [41] Alessia Auriemma Citarella, Marcello Barbella, Madalina G. Ciobanu, Fabiola De Marco, Luigi Di Biasi, Genevveffa Tortora, " Assessing the effectiveness of ROUGE as unbiased metric in Extractive vs. Abstractive summarization techniques", Journal of Computational Science, Volume 87, 102571, 2025.
- [42] Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. "Extractive Summarization as Text Matching". In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6197–6208, Online. Association for Computational Linguistics,2020
- [43] Ruipeng Jia, Yanan Cao, Hengzhu Tang, Fang Fang, Cong Cao, and Shi Wang. 2020. "Neural Extractive Summarization with Hierarchical Attentive Heterogeneous Graph Network". In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3622–3631, Online. Association for Computational Linguistics,2020.