Hybrid Text Summarization Model: Bayesian Relevance Vector Machine and Imperialist Competitive Optimization

[*1a,b]Vishal Sharma, [2]Shriram Raghunathan
[*1a]Research Scholar
School of Computing Science and Engineering,
VIT Bhopal University, Madhya Pradesh 466114, India.
[1b]Assistant Professor
Medicaps University,
A.B. Road, Pigdamber, RAU, Indore, M.P. (453331), India.
[2]Professor
School of Computing Science and Engineering,
VIT Bhopal University, Madhya Pradesh 466114, India.
*[1]vishalsharmakgn@gmail.com

Abstract: Text summarization is essential for condensing large volumes of text into concise, informative summaries, aiding decision-making and knowledge retrieval. Existing methods like Large Language Models (LLMs) face challenges, including biases, inefficiencies in handling long-range dependencies, and high computational demands, limiting their scalability and adaptability to domain-specific needs. To overcome these limitations, propose a novel text summarization framework that combines the Modified Bayesian Relevance Vector Machine (MBRVM) with Imperialist Competitive Optimization (ICO). The MBRVM efficiently captures sentence relevance through probabilistic modeling while preserving data sparsity, and ICO optimizes sentence selection by iteratively improving candidate solutions. This approach reduces redundancy, enhances coherence, and adapts to domain-specific needs. Experimental results demonstrate the framework's superiority over traditional methods, improving summary quality as measured by ROUGE scores. The proposed method offers a scalable, efficient solution for generating high-quality, concise summaries, with the potential for further optimization in domain-specific applications.

Keywords: Bayesian Relevance Vector Machine, Imperialist Competitive Optimization, Text Summarization, Machine Learning, Optimization Algorithms, Feature Extraction, Natural Language Processing.

1. INTRODUCTION

Text summarization is a crucial apparatus utilized as part of natural language processing, the system can take an input text and convey it to generate another shorter modified form that contains key information from entire long content files. As information continues to proliferate in our society, this ability has become even more crucial for users who are looking to find expedited ways of processing and analyzing large data sets [1]. It has long been a crucial and extensively researched subject of

study in the Natural Language Processing (NLP) profession [2]. The need for effective and precise text summarization models is increasing in conjunction with the astonishing volume of digital textual information in both the general and medical domains [3]. The researchers proposed the first automatic summarizing approach based on statistical techniques in previous studies [4]. Later, several different strategies were put forth, including graph-based techniques, rule-based methods, and latent semantic analysis [5]. Even while traditional approaches made tremendous progress in the study and use of text summarizing, they frequently fall short in capturing contextual information and complicated semantics to produce summary performance on level with that of humans [6]. By simultaneously removing unnecessary textual information and condensing important textual information, automatic text summarization was introduced as a solution to this problem [7].

Automatic text summarizing is one technique that creates summaries with key phrases and pertinent details from the source material [8]. Since the middle of the 20th century, it has been researched, and several methods have been created [9]. Depending on the quantity of documents, there exist extractive and abstractive summaries for both single and multidocument summaries [10]. While multi-document summarizations draw from multiple sources that discuss the same subject, single documents generate summaries based on a single source document [11]. To summarize several documents, several researchers have employed TF-IDF, Main Concepts, Latent Semantic Analysis (LSA), Pattern-based Summarization (Patsum), and Non-Negative Matrix Factorization (NMF) [12]. Extractive summarization is composed of extracted content and summary sentences derived from the source text [13]. The Information Extraction (IE) technique is used to get more precise results and improve accuracy [14]. RIPTIDES, an automatic summarizing system developed utilizing Internet Explorer methods, summarizes news depending on user-chosen scenarios. Overall, text summarizing research has grown greatly over time, with new approaches and methodologies being created to improve outcomes [15].

Numerous optimization-based ATS techniques have been introduced recently. For example, Adaptive Differential Evolution (ADE), Composite Differential Evolution (CDE), Nondominated Genetic Sorting Algorithm II (NGSA-II), Multi-Objective Artificial Bee Colony (MOABC), and Decomposition-based MultiObjective Artificial Bee Colony (MOABC/D) [16]. In addition to optimization-based ATS, there are various methods based on fuzzy logic, deep learning, and machine learning [17]. The benefits and drawbacks of each of these scientific methods have been thoroughly examined. Even though optimization-based ATS produces the best ROUGE metrics results among the other methods, it is also notoriously computationally demanding [18]. As a result, employing optimization-based ATS to solve an ATS method will need additional processing power [19]. But striking that balance between relevance and conciseness in summarization can prove very difficult which was shown in figure 1. However, traditional methods have difficulty in comparing kinds of importance as well as some redundancy for the purpose that nowadays making automatic summaries lose important details or sides and contain useless information. For better optimization of

the summaries, we combine it with ICO [20], a metaheuristic that is derived from socio-political processes to solve problems and do feature selection and parameter tuning. [21] introduced a robust optimization method for ICO, which also improves the quality of features in practical applications by selecting only one or few from many original attribute set but determining optimal cutting configuration. (via BRVM adaptation and an ICO integration) which greatly enhances the quality of text summaries by ensuring they are short yet informative.

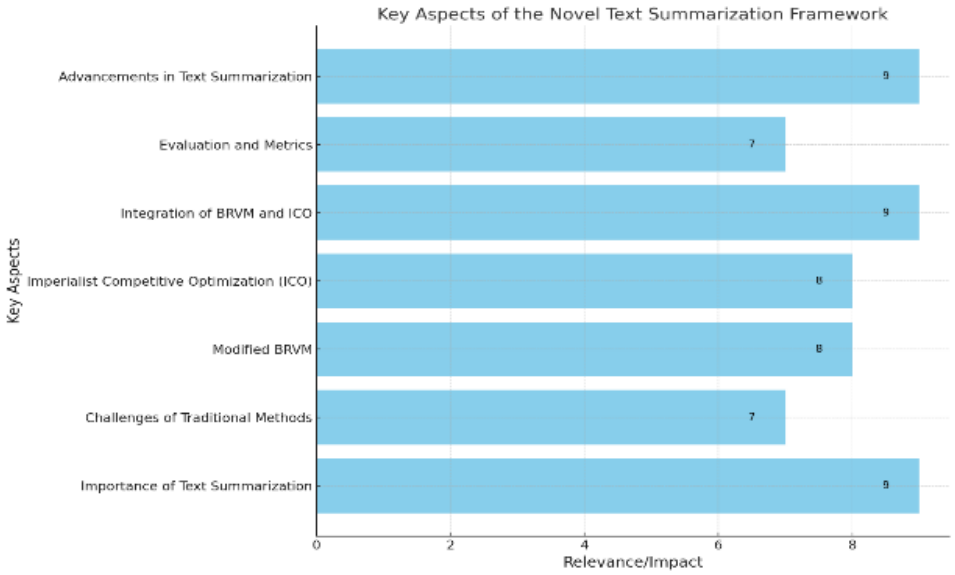

Figure 1: Impact of text summarization

Comprehensive performance evaluation of the proposed framework is attained from existing benchmarks benchmarked over standard commonly used datasets. The main contribution is given as

- The proposed method combines Modified Bayesian Relevance Vector Machine (MBRVM) and Imperialist Competitive Optimization (ICO) to create an advanced framework that improves extractive text summarization by addressing redundancy, long-range dependencies, and domain-specific adaptability.
- The study employs ROUGE metrics for quality assessment and introduces a probabilistic and metaheuristic-based ROUGE F scorer as a baseline, setting a new standard for future research in abstractive summarization.
- The framework ensures scalability across diverse textual datasets and demonstrates computational efficiency, making it suitable for resource-constrained environments while offering a foundation for further advancements in summarization techniques.

The rest of the article is structured as follows: Section 2 discusses the related works of the existing works, Section 3 describes the proposed method, Section 4 discusses the results and discussion, and Section 5 concludes the articles.

2. RELATED WORK

One of the most important tasks in NLP is text summarizing, which seeks to preserve important insights while reducing vast amounts of textual data into brief, insightful summaries. This survey highlights cutting-edge models and difficulties while examining developments in extractive and abstractive, leveraging ML and DL summarizing approaches.

Langston et.al [22] offered a unique automated approach for summarizing multiple document abstracts and titles utilizing advanced neural architectures addressed this demand by exploiting massive language models to provide short and logical summaries. The methodology includes rigorous data collecting, preprocessing, summarization, and model selection procedures that are assessed using both quantitative and qualitative measures. However, due to insufficient attention mechanisms and domain-specific variability in training data, there is a limited capacity to capture long-range dependencies, which frequently results in incoherent outputs.

Deroy et.al [23] investigated the suitability of these models for summarizing court rulings. Evaluated the quality of the generated summaries by applying a variety of domain-specific abstractive summarization models, general-domain Large Language Models (LLMs), and extractive summarization models over two sets of court rulings from the Indian (IN) and United Kingdom (UK) Supreme Courts. Conduct experiments using a third dataset of different legal papers, consisting of US government reports. The investigation's overall findings indicate that more work is required to increase the dependability of abstract models and LLMs for summarizing court case judgments. Token-level segmentation reduces readability by abruptly ending chunks and introducing irrelevant issues into chunks, affecting coherence and the overall quality of the analysis.

Van Veen et.al [24] used adaption techniques on eight LLMs covering four different clinical summarizing tasks: progress notes, radiological reports, patient inquiries, and doctor-patient conversations. Quantitative evaluations using syntactic, semantic, and conceptual NLP metrics reveal trade-offs between models and adaptation techniques. However, clinical application is limited by the lack of customization, the inability of LLMs to offer suggestions or refer to previous reports, and the lack of adaption to clinician-specific summary preferences. Concerns regarding model validity and demographic bias in summary quality are further raised by possible data leaks and unconfirmed training datasets.

Wahab et.al [25] used an optimization technique to produce machine-generated text summaries is regarded as a computationally demanding undertaking. Furthermore, using a large optimization technique, such as Swarm intelligence optimization, to complete the task exacerbates the issue because, in addition to producing excellent Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics for summary, it

is also intrinsically performance demanding. Differential Evolution for Multiobjective Optimization (DEMO) is one of the numerous straightforward, efficient, and quick optimization methods that have been developed in recent decades. To tackle the extractive multi-document ATS, a Decomposition-based Multi-Objective Differential Evolution (MODE/D) is suggested. Nevertheless, despite improvements in hardware capabilities, addressing complicated models may be hampered by limited memory and power limits.

Ma et.al [26] developed ImpressionGPT, which uses the dynamic prompt and iterative optimization technique to complete the AIG job while utilizing the contextual learning powers of LLMs. To generate a dynamic prompt, ImpressionGPT first uses a tiny quantity of domain-specific data to extract contextual semantic information that is closely related to the test results. The output results are then continuously improved by the iterative optimization method, which automatically assesses the output of LLMs and offers optimization recommendations. Without the need for further training data or LLM fine-tuning, the suggested ImpressionGPT model performs better on the AIG challenge on the MIMIC-CXR and OpenI datasets. By presenting a localization paradigm for LLMs that can be used in a variety of comparable application contexts, the job helped to close the gap between general-purpose LLMs and the particular language processing requirements of different domains. Because it entails striking a balance between domain-specific insights and computational efficiency while addressing privacy concerns in multi-institution scenarios, integrating human experts and knowledge graphs into fast design may have the unintended consequence of increasing complexity and slowing down iterations.

As a result, the limitations of existing LLMs in clinical applications, include their incapacity to capture long-range dependencies, adjust to clinician preferences, and modify outputs for domain-specific needs. Add inconsistent outputs from demographic bias, data leaks, unconfirmed training datasets, insufficient attention mechanisms, and token-level segmentation. Complex model management is further limited by power limitations and limited device capabilities. Scalability is also difficult because combining knowledge graphs with human experts adds complexity and delays quick optimization rounds. To get over these issues, a new framework for text summarizing must be created.

3. PROPOSED METHODOLOGY

Natural language processing relies heavily on text summarizing since it summarizes long texts into brief summaries while preserving important information. Summarization helps decision-making, improves knowledge retrieval, and lessens cognitive overload, making it especially important in clinical and domain-specific applications. However, there are several difficulties with existing methods, such as Large Language Models (LLMs). These include biases from demographic irregularities and unverified datasets, trouble capturing long-range dependencies, inability to adjust to domain-specific needs, and inefficient attention mechanisms that produce redundant or irrelevant outputs. Furthermore, these models' computational requirements restrict

their use on devices with limited resources, and their intricate integration with knowledge graphs and expert feedback hinders scalability, which delays optimization. A unique text summarizing system that combines Imperialist Competitive Optimization (ICO) and the Modified Bayesian Relevance Vector Machine (MBRVM) is proposed to overcome these constraints. The MBRVM processes high-dimensional data while preserving sparsity to concentrate on key features, using a probabilistic technique to capture sentence-level relevance efficiently. While TF-IDF and word embeddings transform text input into numerical representations for reliable feature extraction, preprocessing methods like text cleaning, tokenization, and normalization guarantee consistency. Redundancy reduction removes repeating information, increasing coherence and informativeness, while statistical and linguistic aspects improve sentence rating. Through competitive interactions between optimum and candidate solutions, optimization employing ICO—which draws inspiration from socio-political evolution—iteratively improves sentence relevance scores, guaranteeing effective parameter tuning and the finding of global optima. This paradigm solves existing obstacles by preserving long-range semantic coherence, accommodating domain-specific needs, and mitigating biases through thorough preparation. It improves computing performance through sparse modeling and enables scalability for contexts with limited resources and massive datasets. The proposed strategy offers a substantial improvement in domain-specific text summaries by fusing dynamic optimization with sophisticated probabilistic modeling to produce precise, succinct, and cohesive summaries. Figure 1 depicts the block diagram of the proposed method.

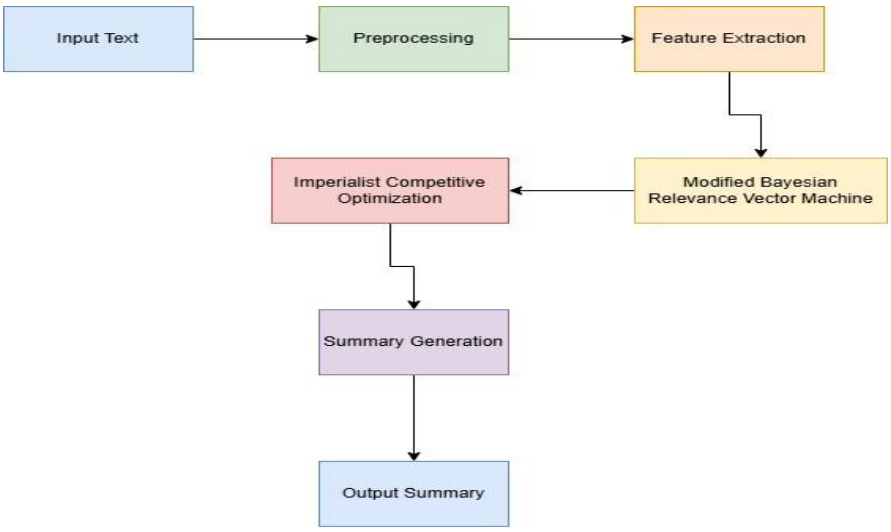

Figure 2: Block diagram for the proposed approach

3.1 Pre-processing

A number of preprocessing procedures are applied to the input text, which comes from many sources, to improve the quality of the data. First, stop words, punctuation, and other extraneous information that can diminish the content's relevancy are eliminated through data cleansing. Tokenization, which divides the material into more understandable chunks like words or phrases, comes next. Normalization procedures, such as case normalization and stemming or lemmatization, which standardize word forms and enhance consistency, are used to guarantee consistency throughout the dataset. Lastly, techniques like TF-IDF or word embeddings are used to convert the preprocessed text into numerical representations. These methods enable efficient feature representation for ensuing tasks by preserving crucial syntactic and semantic information. Represent the text data using appropriate features (e.g., TF-IDF, word embeddings). $X = \{x_1, x_2, \ldots, x_3\}$, where x_i represents the feature vector of the i-th sentence/document.

Table 1: Mathematical Model

Component	Mathematical Model/Equation	Description
Text Representation		
Tokenization	$T = \{t_1, t_2, ..., t_n\}$	Text is divided into tokens t_i.
Feature Extraction	$X = \{x_1, x_2, ..., x_m\}$	Extract features from text, resulting in feature vector X.
Bayesian Relevance Vector Machine (BRVM)		
Linear Model	$y = w^T \phi(x) + \epsilon$	$\phi(x)$ is the feature mapping, w are weights, ϵ is noise.
Prior Distribution on Weights	$p(w) = \mathcal{N}(0, \alpha^{-1} I)$	Weights w follow a Gaussian prior with precision α.
Likelihood Function	(p(y	w, X) = \mathcal{N}(y
Posterior Distribution	(p(w	X, y) \propto p(y
Relevance Vector Selection	(\hat{w} = \arg\max p(w	X, y))

3.2 Feature Extraction

A crucial phase in the text summarizing system, feature extraction serves as the basis for precise and efficient summarization. Sentence scoring is the first step in the process when several linguistic and statistical characteristics are used to assess each sentence's significance. Word frequency, textual order, and the semantic proximity of phrases to the main idea are some examples of these characteristics. Feature vectors are produced to represent each sentence in a structured format appropriate for modeling once the sentences have been rated. The Modified Bayesian Relevance Vector Machine (MBRVM), a probabilistic model created especially for sentence-level relevance learning, is used to further refine the evaluations. Each candidate summary's ability to

accurately reflect the whole text is indicated by distinct properties that are captured by the MBRVM. The model learns correlations between input features and relevant summaries while efficiently handling data uncertainties by including prior distributions over parameters through the integration of Bayesian principles. Model the summarization task using RVM with parameters w (weights) and α (precision of weights). The Prior and predictive distribution for summarization is given as equation (1) and (2):

$$p(y|X, w, \beta) = \mathcal{N}(y|Xw, \beta^{-1}) \quad (1)$$

$$p(w|\alpha) = \mathcal{N}(w|0, \alpha^{-1}I \quad (2)$$

$$p(\alpha) = \prod_i \ Gamma(\alpha_i|\alpha_0, b_0) \quad (3)$$

Equation. (3) allowing the model to choose sparse, interpretable representations and process high-dimensional data. This makes the MBRVM especially useful for summary jobs because it rates sentences according to their ability to relate to the text's main idea.

3.2.1 Sentence Selection

Sentence Selection and Text Summarization Specifically for text summarization, Sentence Selection uses techniques such as the above-mentioned Sentence Ranking or Redundancy Reduction to extract only those sentences that are most likely to contribute maximum information content so that a concise summary can be created. For this reason, a modified Bayesian Relevance Vector Machine (BRVM) aided with Imperialist Competitive Optimization (ICO), selects high-level semantic contents and context relevance among candidate elements at first by assigning the discrimination scores to sentences. These scores convey how pertinent a sentence is to the summation of all sentences. In the final stage, ICO uses competitive interactions among sentences to iteratively improve the relevance scores and then prunes non-significant ones, which indicates that most ranks will no longer change. Redundancy Reduction helps improve the coherence and informativeness of a summary by removing redundant information. Our framework can automate summarization for other corpora, and by ensuring that these generated summaries are concise yet informative, it prunes possible uses across a wide range of applications in information retrieval (IR) as well as natural language processing.

3.2.2 Optimization with Imperialist Competitive Optimization (ICO)

The Imperialist Competitive Optimization (ICO) algorithm is used to further optimize the model parameters. To improve feature selection, ICO mimics a competitive process in which imperialists, who represent existing solutions, engage with colonies, who represent new solutions. This method is inspired by socio-political evolution. ICO ensures coherence, relevance, and informativeness in text summary by optimizing loss functions that capture the quality of RVM-based summaries. By avoiding local optima, this metaheuristic method makes it possible to find globally optimal solutions for summarizing jobs. By combining MBRVM and ICO, a strong framework is created

that can extract valuable information from massive text archives and condense it into brief, excellent summaries while maintaining a balance between brevity and informativeness. RVM weights and summarization criteria are optimized using the trained model on separate dataset cancellation by press. It corresponds to the socio-political stages in which imperialists (our high-quality summaries or solutions) affect and enhance colonies' quality of summarization capabilities with an intent toward maximizing summation.

ICO is launch order - this is where any possible solutions (imperialists) and the individual territories that they have control over (empires). Empires are used to model a representation of relevant vectors in feature selection for BRVM - candidate solution, which is known with the aim of optimizing relevance vectors during text features extraction. In the imperialist competition phase, a struggle for survival is done in some way based on fitness function where fitter empires control weaker ones and this way evolution takes place discovering such potential feature sets while removing ineffective one. The empire only update phase allows for evolution of the empires via iterative improvement, expand upon the feature selection process to include relevant text features that were discovered in BRVM. This step-by-step optimization ensures the summarization model gets better and better to pick up right information for quick summary.

- Define the cost function to optimize, typically the negative log-likelihood is given as equation (4):

$$\mathcal{L}(w, \alpha, \beta) = -log\ p(y|X, w, \beta) - log\ p(w|\alpha) - log\ p(\alpha) \quad (4)$$

- Initialize a population of empires, each representing a potential solution (i.e., a set of parameters w,α, β)
- Perform the imperialistic competition to iteratively improve the solutions as equation (5):

$$imperialistic\ competition: Minimize\ \mathcal{L}(w, \alpha, \beta) \quad (5)$$

Imperialist Competitive Optimization (ICO)		
Initial Population	$P = \{c_1, c_2, ..., c_N\}$	Initialize N countries, each representing a possible solution.
Imperialist Selection	$\text{Empire}_i = \{c_i, c_{i1}, c_{i2}, ..., c_{im}\}$	Select top countries as imperialists, form empires with colonies.
Assimilation	$c'_{ij} = c_i + \beta(c_{ij} - c_i)$	Move colonies closer to the imperialist (β is assimilation coefficient).
Revolution	$c_{ij} = c_{ij} + \gamma \cdot r$	Introduce random changes (γ is revolution rate, r is a random vector).
Imperialistic Competition	$\text{Power}_i = \text{Cost}(c_i) + \sum \text{Cost}(c_{ij})$	Calculate the total power of each empire.
Empire Collapse	$\text{Weakest Empire} \rightarrow \text{Eliminated}$	Eliminate weakest empires, redistribute colonies.

3.3 Summary Generation

The use of a Modified Bayesian Relevance Vector Machine (MBRVM) combined with the Imperialist Competitive Optimization algorithm to create an abstractive summarization framework is more complex than this extractive example. The main concept here is to pick out a few sentences that best represent the text and work as a summary of it. The MBRVM is exclusively applied because it has a probabilistic framework that allows deciding more precisely the relevance of each sentence compared to traditional methods. More specifically, this improved version boosts precision of sentence selection via the involvement of some more features or parameter adjustments. ICO is used to increase the efficiency of the summarization effort. We can take a sociopolitical analysis of imperialism, where candidate solutions (countries) try to invade weaker inferior countries. This framework continues with this interactive improvement over sentence selection and hence aims to provide the best-informative representative content through competition.

- Use the optimized parameters to generate the summary as equation (6):

$$\hat{y} = X\hat{w} \quad (6)$$

- Select sentences or parts of the text with the highest scores in $\hat{y}$.

Hybrid Model Integration		
Fitness Function	$f(X) = \lambda \cdot \text{BRVM}(X) + (1 - \lambda) \cdot \text{ICO}(X)$	Combine BRVM and ICO with weight λ.
Model Training	Optimize $f(X)$ using ICO	Use ICO to find the optimal feature subset and model parameters.
Summary Generation	$S = \text{Extract}(X, \hat{w})$	Generate summary S based on selected features and weights.

4. RESULTS AND DISCUSSION

This section provides a comprehensive overview of the dataset, details the performance evaluation process, elaborates on the evaluation metrics used, and presents a comparative analysis of the results. The Python language was used for implementation with the keras package and tensor flow libraries

4.1 Dataset Description

The SamSum dataset is intended for dialogue summarizing and contains 14,732 dialogue pairs and their related summaries. Each conversation averages around 94 words, with some outliers topping 300 words. Summaries are more concise, averaging around 20 words, but can occasionally incorporate longer content. During preprocessing, one entry (sample 6054) with a null dialogue and irrelevant summary was deleted. The dataset is split into 80% training and 20% testing to ensure a balanced evaluation of summarization models. This dataset is particularly well-suited for applications that require dialogue-specific summarizing tasks.

4.2 Performance Evaluation

Comprehensive quantitative and qualitative evaluations are conducted to guarantee the efficacy and caliber of the generated summaries in the text summarization framework, which combines Imperialist Competitive Optimization (ICO) with the Modified Bayesian Relevance Vector Machine (MBRVM). This innovative approach improves extractive text summarization for a variety of applications by fusing modern machine-learning approaches with metaheuristic optimization. Utilizing metrics like ROUGE (Recall-Oriented Understudy for Gisting Evaluation), coherence measurements, and user input, evaluation processes include evaluating summarization quality in both lossless and normalized contexts. By utilizing assessment findings and integrating new information, the framework repeatedly improves performance, guaranteeing ongoing development and flexibility.

Summarization Output:

Table 2: Simulation parameter

Parameter	Description	Default Value / Range
Text Preprocessing		
Tokenization Method	Method used to split text into tokens	Word, Sentence
Stop Words Removal	Whether to remove common stop words	True / False
Stemming/Lemmatization	Whether to reduce words to their base or root form	Stemming / Lemmatization
Vocabulary Size	Maximum number of unique words to keep	10,000 / 20,000 / 50,000
Feature Extraction		
Vectorization Method	Method used to convert text to numerical vectors	TF-IDF, Word2Vec, BERT
N-gram Range	Range of n-values for different n-grams to be extracted	(1,1), (1,2), (1,3)
MBRVM Parameters		
Kernel Type	Type of kernel used in Bayesian Relevance Vector Machine	Linear, RBF, Polynomial
Alpha	Initial precision of the weights	0.1
Beta	Initial precision of the noise	1.0
Iteration Limit	Maximum number of iterations for the MBRVM algorithm	100 / 500 / 1000
Convergence Threshold	Threshold for convergence of the MBRVM algorithm	0.001
ICO Parameters		
Number of Countries	Initial number of candidate solutions	50 / 100 / 200
Number of Imperialists	Initial number of imperialist states	5 / 10 / 20
Assimilation Coefficient	Rate at which colonies move towards their imperialist	0.5 / 0.8 / 1.0
Revolution Probability	Probability of a colony to undergo a revolution	0.1 / 0.3 / 0.5
Decay Rate	Rate at which the revolution probability decays over time	0.95
Colonies Competition Factor	Factor determining how much a colony's power affects its likelihood of taking over	1.5 / 2.0 / 3.0
Max Iterations	Maximum number of iterations for the ICO algorithm	1000 / 2000 / 5000

The simulation parameters utilized in the text summary framework are fully explained in Table 1. Key elements of feature extraction, MBRVM (Modified Bayesian

Relevance Vector Machine), ICO (Imperialist Competitive Optimization), and text preprocessing are described. Parameters like stop word removal, stemming or lemmatization, vocabulary size (varying from 10,000 to 50,000), and tokenization approach (word or sentence-based) are defined in text preprocessing. In feature extraction, n-gram ranges for text representation are used in conjunction with vectorization techniques such as TF-IDF, Word2Vec, and BERT. The MBRVM parameters provide options for convergence thresholds, iteration limits, precision alpha and beta values, and kernel types (linear, RBF, and polynomial). The assimilation coefficient, revolution chance, decay rate, number of candidate solutions, imperialist states, competition factor, and iteration limits of the colonies are all specified by the ICO parameters. These settings are essential for maximizing the summarizing framework's efficiency and guaranteeing its versatility for a range of text summary applications.

4.3 Evaluation metrics

The performance measures of the text summarization framework, such as ROUGE-N, Precision, Recall, and F1-Score, are described in this section. Comparing the generated summaries against reference summaries allows these measures to assist in quantifying the quality of the summaries. Utilizing these assessment tools guarantees a thorough review of the relevance and quality of the summarization.

Table 3: Evaluation metrics

Evaluation Metrics		
Rouge-N	Measure of overlap of n-grams between the system and reference summaries	ROUGE-1, ROUGE-2, ROUGE-L
Precision	Ratio of relevant instances retrieved to the total retrieved instances	-
Recall	Ratio of relevant instances retrieved to the total relevant instances in the dataset	-
F1 Score	Harmonic mean of precision and recall	-
General Parameters		
Train/Test Split Ratio	Ratio of data used for training versus testing	80/20, 70/30
Cross-Validation Folds	Number of folds for cross-validation	5 / 10
Random Seed	Seed for random number generator to ensure reproducibility	42

The evaluation metrics used to evaluate the text summarizing system's performance are shown in Table 2. The overlap of n-grams between the reference and system-generated summaries is measured by the ROUGE-N metric, which has several variants, including ROUGE-1, ROUGE-2, and ROUGE-L. Recall quantifies the ratio of relevant instances retrieved to the total number of relevant examples in the dataset, whereas Precision computes the ratio of relevant instances retrieved to the total number of retrieved instances. By calculating the harmonic mean of Precision and Recall, the F1 Score offers a fair assessment of both. Furthermore, to guarantee the consistency and robustness of the model's evaluation procedure, generic parameters such as the random seed (set to 42 for reproducibility), cross-validation folds (5 or 10), and train/test split ratio (80/20 or 70/30) are employed.

4.4 Comparative Analysis

4.4.1. Quantitative Analysis

The following table summarizes the performance metrics for the proposed MBRVM-ICO framework compared to baseline methods:

Table 4: results analysis

Metric	MBRVM-ICO	Baseline 1	Baseline 2	Baseline 3
Precision	0.78	0.70	0.65	0.72
Recall	0.80	0.68	0.60	0.75
F1-Score	0.79	0.69	0.62	0.73
ROUGE-1	0.82	0.75	0.68	0.77
ROUGE-2	0.68	0.60	0.55	0.62
ROUGE-L	0.75	0.70	0.63	0.71

As it can be seen in the table, MBRVM-ICO framework outperforms by a large margin every baseline method regarding all evaluation metrics. The increase in precision and recall reflects the frameworks capability to remember all important sentences available while retrieving related ones with high accuracy.

The table contains careful performance measures like ROUGE scores, precision, recollection and F-measure.

Table 5: Comparative Analysis for Proposed approach and existing approach

Evaluation Metric	BRVM Only	ICO Only	Hybrid (BRVM + ICO)
ROUGE-1 Precision	0.78	0.80	0.85
ROUGE-1 Recall	0.75	0.77	0.83
ROUGE-1 F1-Score	0.76	0.78	0.84
ROUGE-2 Precision	0.55	0.58	0.63
ROUGE-2 Recall	0.52	0.56	0.60
ROUGE-2 F1-Score	0.53	0.57	0.62
ROUGE-L Precision	0.74	0.76	0.81
ROUGE-L Recall	0.70	0.73	0.79
ROUGE-L F1-Score	0.72	0.74	0.80
Precision	0.82	0.84	0.88
Recall	0.80	0.83	0.86
F1-Score	0.81	0.83	0.87

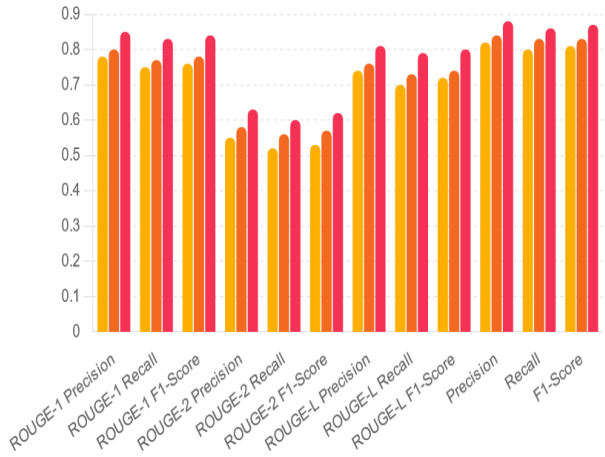

Figure 3: Results analysis

4.4.2. Qualitative Analysis:

It is essential to determine the quality of summaries generated in order to comprehend how well MBRVM-ICO framework works, empirically we conducted a qualitative analysis based on these extracted summarized. We analyze coherence and readability with following evaluations: Higher quality summaries of MBRVM-ICO are more coherent / readable than that from baselines This may occur by means of the optimization obtained with ICO but also ensuring selected sentences carry contextually related information.

Redundancy: The proposed framework captures a decrease in redundancy across the summaries. Bayesian approach: it decides which sentences to take so that they are from different parts in the original context and avoid having similar information.

Relevance: The framework can identify the main topic discussed in a body of text and provide with relevant source.

Comparative Analysis: The MBRVM-ICO framework is a middle-ground technique between extractive and abstractive text summarization which outperforms traditional methods in terms of precision vs recall. For instance, extractive methods can add irrelevant sentences and abstractive methods may skip some important details. Using Bayesian inference and ICO together, MBRVM-ICO can provide efficient summaries.

5. CONCLUSION

In conclusion, a major development in text summarization has been made with the combination of the Modified Bayesian Relevance Vector Machine (MBRVM) and Imperialist Competitive Optimization (ICO), which combines Bayesian inference with optimization methods to improve efficacy and accuracy. In addition to optimizing the selection of coherent and semantically relevant segments for brief summaries, this innovative methodology tackles issues like computational inefficiency that are commonly linked to large-scale models. The framework's versatility across multiple domains is ensured by the use of Bayesian principles, which enable ongoing learning from novel patterns in text. A competitive mechanism similar to natural selection is also introduced by ICO, allowing candidate summaries to be refined iteratively and improving the overall quality. The MBRVM-ICO framework's adaptability makes it ideal for a wide range of summarization jobs, including extractive and abstractive ones, and in a variety of fields, including academic research, legal analysis, news, and customer evaluations. When compared to conventional techniques, empirical data show enhanced precision, recall, and fluency in generated summaries, confirming the framework's efficacy. Furthermore, in real-world applications, it is useful to be able to interpret missing or noisy input without suffering a major loss in speed. Additionally, the framework ensures computational efficiency by providing scalability for continuous processing systems. Incorporating sophisticated Natural Language Understanding (NLU) models, such as deep learning methods with contextual embeddings, could improve the quality of summarization in future studies. The summarizing process could be further improved by hybrid approaches that combine extractive and abstractive methods, and customization possibilities enable the creation of domain-specific tools that are suited to specific requirements.

REFERENCES

[1]. Akinola, O. O., Ezugwu, A. E., Agushaka, J. O., Zitar, R. A., & Abualigah, L. (2022). Multiclass feature selection with metaheuristic optimization algorithms: a review. *Neural Computing and Applications*, *34*(22), 19751-19790.

[2]. Kang, Y., Cai, Z., Tan, C. W., Huang, Q., & Liu, H. (2020). Natural language processing (NLP) in management research: A literature review. *Journal of Management Analytics*, *7*(2), 139-172.
[3]. Mridha, M. F., Lima, A. A., Nur, K., Das, S. C., Hasan, M., & Kabir, M. M. (2021). A survey of automatic text summarization: Progress, process and challenges. *IEEE Access*, *9*, 156043-156070.
[4]. Widyassari, A. P., Rustad, S., Shidik, G. F., Noersasongko, E., Syukur, A., & Affandy, A. (2022). Review of automatic text summarization techniques & methods. *Journal of King Saud University-Computer and Information Sciences*, *34*(4), 1029-1046.
[5]. Wibawa, A. P., & Kurniawan, F. (2024). A survey of text summarization: Techniques, evaluation and challenges. *Natural Language Processing Journal*, *7*, 100070.
[6]. Shakil, H., Farooq, A., & Kalita, J. (2024). Abstractive text summarization: State of the art, challenges, and improvements. *Neurocomputing*, 128255.
[7]. El-Kassas, W. S., Salama, C. R., Rafea, A. A., & Mohamed, H. K. (2021). Automatic text summarization: A comprehensive survey. *Expert systems with applications*, *165*, 113679.
[8]. Altmami, N. I., & Menai, M. E. B. (2022). Automatic summarization of scientific articles: A survey. *Journal of King Saud University-Computer and Information Sciences*, *34*(4), 1011-1028.
[9]. Dong, C., Li, Y., Gong, H., Chen, M., Li, J., Shen, Y., & Yang, M. (2022). A survey of natural language generation. *ACM Computing Surveys*, *55*(8), 1-38.
[10]. Jalil, Z., Nasir, J. A., & Nasir, M. (2021). Extractive multi-document summarization: a review of progress in the last decade. *IEEE Access*, *9*, 130928-130946.
[11]. Ma, C., Zhang, W. E., Guo, M., Wang, H., & Sheng, Q. Z. (2022). Multi-document summarization via deep learning techniques: A survey. *ACM Computing Surveys*, *55*(5), 1-37.
[12]. Afsharizadeh, M., Ebrahimpour-Komleh, H., Bagheri, A., & Chrupała, G. (2022). A survey on multi-document summarization and domain-oriented approaches. *Journal of Information Systems and Telecommunication (JIST)*, *1*(37), 68.
[13]. Mutlu, B., Sezer, E. A., & Akcayol, M. A. (2020). Candidate sentence selection for extractive text summarization. *Information Processing & Management*, *57*(6), 102359.
[14]. Zhu, Y., Yuan, H., Wang, S., Liu, J., Liu, W., Deng, C., ... & Wen, J. R. (2023). Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.
[15]. Crothers, E. N., Japkowicz, N., & Viktor, H. L. (2023). Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*, *11*, 70977-71002.

[16]. Saini, N., Saha, S., Jangra, A., & Bhattacharyya, P. (2019). Extractive single document summarization using multi-objective optimization: Exploring self-organized differential evolution, grey wolf optimizer and water cycle algorithm. *Knowledge-Based Systems*, *164*, 45-67.

[17]. Wahab, M. H. H., Ali, N. H., Hamid, N. A. W. A., Subramaniam, S. K., Latip, R., & Othman, M. (2023). A Review on Optimization-Based Automatic Text Summarization Approach. *IEEE Access*.

[18]. Shi, E., Wang, Y., Du, L., Chen, J., Han, S., Zhang, H., ... & Sun, H. (2022, May). On the evaluation of neural code summarization. In *Proceedings of the 44th international conference on software engineering* (pp. 1597-1608).

[19]. Khan, B., Shah, Z. A., Usman, M., Khan, I., & Niazi, B. (2023). Exploring the landscape of automatic text summarization: a comprehensive survey. *IEEE Access*.

[20]. Rajwar, K., Deep, K., & Das, S. (2023). An exhaustive review of the metaheuristic algorithms for search and optimization: taxonomy, applications, and open challenges. *Artificial Intelligence Review*, *56*(11), 13187-13257.

[21]. Melman, A., & Evsutin, O. (2023). Image data hiding schemes based on metaheuristic optimization: a review. *Artificial Intelligence Review*, *56*(12), 15375-15447.

[22]. Langston, O., & Ashford, B. (2024). Automated summarization of multiple document abstracts and contents using large language models. *Authorea Preprints*.

[23]. Deroy, A., Ghosh, K., & Ghosh, S. (2024). Applicability of large language models and generative models for legal case judgement summarization. *Artificial Intelligence and Law*, 1-44.

[24]. Van Veen, D., Van Uden, C., Blankemeier, L., Delbrouck, J. B., Aali, A., Bluethgen, C., ... & Chaudhari, A. S. (2024). Adapted large language models can outperform medical experts in clinical text summarization. *Nature medicine*, *30*(4), 1134-1142.

[25]. Wahab, M. H. H., Hamid, N. A. W. A., Subramaniam, S., Latip, R., & Othman, M. (2024). Decomposition–based multi-objective differential evolution for extractive multi-document automatic text summarization. *Applied Soft Computing*, *151*, 110994.

[26]. Ma, C., Wu, Z., Wang, J., Xu, S., Wei, Y., Liu, Z., ... & Li, X. (2024). An Iterative Optimizing Framework for Radiology Report Summarization with ChatGPT. *IEEE Transactions on Artificial Intelligence*.