International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Performance Evaluation of Nature-Inspired Metaheuristic Approaches for Single Document Text Summarization

Pravesh Patel1*, Paresh Solanki2

1*,2 Faculty of Engineering and Technology, Ganpat University, India

Email:- pravesh.patel@ganpatuniversity.ac.in Email:- paresh.solanki@ganpatuniversity.ac.in

*Corresponding author:- Pravesh Patel

*Faculty of Engineering and Technology, Ganpat University, India

Email:- pravesh.patel@ganpatuniversity.Ac.In

ABSTRACT

In today era, day by day huge amount of data is collected on internet. The reading of text document or retrieving important information are time consuming process, so there is need for introducing effective text summarization technique. Text summarization, is the process of retrieving key information from lengthy document, its plays an essential role in information retrieval and content extraction. The paper we presented a comprehensive examination of nature-inspired metaheuristic algorithms, such as firefly, Cuckoo Search(CS) and Particle Swarm Optimization (PSO) to improve text summarization with an emphasis on single document datasets such as DUC-2001 and DUC-2002. The measurement of generated text summaries quality, generated summaries of datasets are compared with existing golden summaries and evaluated using ROUGE score. Our results show that nature-inspired metaheuristic-based approaches show potential for enhancing text summary of individual documents, metaheuristics methods improve summarizing effectiveness while offering a fresh viewpoint on how to handle the process within the confines of a single document dataset.

Keywords: Text summarization, nature-inspired metaheuristic, extractive text summarization, ROUGE score, Natural Language Processing

1. INTRODUCTION

Text summarization is a process of producing a concise and precise summary of a text document[1]. It is very useful in today's information-rich society as it helps to reduce the time spent on reading long articles and documents. It can also be used to quickly identify key points in a text, which can then be used for further research or discussion. Text summarization can also be used to identify key trends in a document, and to create a summary of a large amount of data. This is especially useful for businesses who need to quickly analyses customer feedback or financial data. Single document text summarization is important because it helps to reduce the amount of time that it takes to read and comprehend a document[2]. It also helps to provide a quick overview of the main points of the document. This can be especially useful for people who do not have the time to read through the entire document or for those who are looking for a quick overview of the content. By summarizing the document, readers can quickly get a sense of the main points and skip over the details that are not as important[3]. There was various research proposed taxonomy of text summarization among them such types of text summarization approach based on generated summary are extractive, abstractive, Semantic Analysis, Graph-based and Concept based text summarization. The extractive text summarization selects the sentences that are most relevant and important from the original text and combine them with other sentences from the original text to create a new summary that is shorter than the original text[4]. The generation of a new summary that is based on the major concepts of the original text but is not always an exactly replica of the original text is what is involved in the process of abstractive summarization. Latent Semantic Analysis (LSA) based text summarization uses mathematical and statistical techniques to find the common themes in a text, and then generates a new summary based on those themes[5][6]. The Graph-Based Summarization uses a graph structure to represent the relationships between the concepts in a text, and then generates a new summary based on the graph structure. The concept-based summarization uses Natural Language Processing (NLP) techniques to identify the important concepts in a text, and then generates

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

a new summary based on those concepts[7].
Extractive text summarization is a method of summarizing a document by selecting the most important sentences from the text. The main idea behind extractive summarization is to identify the most significant sentences in a document and assemble them to form a summary[8]. It uses natural language processing (NLP) algorithms to identify the most relevant sentences in a document and combines them to create a short summary. Extractive summarization is usually more accurate than abstractive summarization as it relies on the exact words from the original text. The Fig.1[9] describe the flow of extractive text summarization:

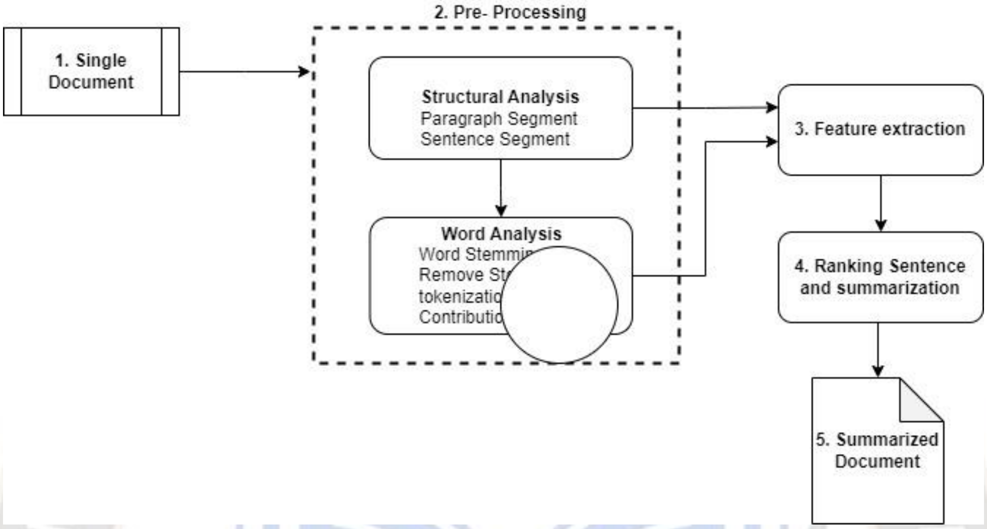

Fig 1 . Flow of Extractive text summarization[9]

1. Input: A longer source document or text.
2. Pre-Processing: The text is divided into sentences, tokenized, and lemmatized.
3. Feature Extraction (Sentence scoring): Features such as term frequency, in-verse document frequency (IDF), and sentence position are extracted.
4. Summarization: The most important sentences are selected based on their similarity scores.
5. Output: A shorter, more succinct version of the original text.

Role of metaheuristic optimization in text summarization process
Text summarizing techniques that extract text from a single document heavily rely on metaheuristic optimization. The goal of extractive text summarizing is to automatically pick a subset of a lengthier document's sentences or phrases that best convey the most significant information in order to produce a metaheuristic optimization techniques are useful because they efficiently search across a wide solution space to identify succinct and logical summary[10]. In this situation, the best subset of phrases to include in the summary. The role of metaheuristic optimization in single document text summarization the following ways[11][12][2][13][14][15]:

Objection Function: An objective function is established in extractive text summarizing in order to assess a summary's quality. This function takes into account a number of variables, including coherence, redundancy, and sentence importance. Maximizing this objective function is the aim of optimization, suggesting a concise summary.
Solution Representation: A binary vector, with each part denoting whether a sentence is picked or not, is commonly used to represent the set of sentences that can be included in the summary. These binary representations are worked with by metaheuristic optimization techniques.
Metaheuristic Techniques: To get the best summary, one can use a variety of metaheuristic optimization techniques[16][17]. Particle swarm optimization, Cuckoo search and firefly optimization[18] are a few often employed methods. The optimal subset of phrases may be chosen thanks to these algorithms' effective exploration of the solution space.
Initialization and Population: Usually, metaheuristic algorithms begin with a population of potential solutions. These preliminary answers may be produced at random or by applying heuristics, such picking the phrases with the greatest information.
Iterative Search: Iteratively enhancing the potential solutions is part of the optimization process. To explore and take advantage of the solution space, metaheuristic algorithms use operators including mutation, crossover, and local search. In order to guarantee the discovery of a globally optimum or nearly optimal summary, these operators are made to strike a balance between exploration and exploitation.
Termination Criteria: Until specific termination requirements are satisfied, the optimization process keeps going. These standards may consist of a time constraint, a convergence criterion, or a maximum number of repetitions.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Post-processing: The final summary is formed by extracting the chosen sentences when the optimization procedure is finished. The consistency and readability of the summary can be enhanced by adding further post-processing techniques, such rearranging the phrases or softening the transitions between them.

Evaluation: Evaluation measures such as ROUGE (Recall-Oriented Understudy for Gisting Evaluation [19]evaluate the quality of the produced summary. These metrics assess how well the key details from the original content are conveyed in the summary.

2. NATURE-INSPIRED METAHEURISTIC APPROACHES FOR TEXT SUMMARIZATION

The process of producing a succinct and coherent summary of a single document by choosing the parts of its phrases or sections that contain the most crucial information is known as single document extractive text summarizing. The selection of these sentences may be made as efficient as possible by using heuristic techniques[20][21]. In this case, three metaheuristic algorithms that may be used for text summarization are Particle Swarm Optimization (PSO)[22][23], Cuckoo Search[2], and Firefly Algorithm[12].

Particle Swam Optimization(PSO): PSO is a population-based optimization method modelled after fish or bird social structures. Potential solutions, or sentence subsets, can be represented as particles in the context of text summarization. A candidate summary is represented by the location of each particle in the solution space. As the particles travel across the solution space, they modify their locations in accordance with both the global best solution discovered by the entire swarm and their own best solutions[24]. A summary's quality may be assessed using the fitness function by taking into account factors like coherence and informative-ness.

Cuckoo Search Optimization(CS): Another population-based optimization approach that draws inspiration from certain cuckoo species' nest parasitism is called Cuckoo Search. You may employ a population of possible solutions (summaries) as nests in text summarization. The potential solutions are represented by cuckoos. By placing eggs in nests, the algorithm creates new solutions and swaps out less suitable answers with better ones. In terms of coherence and topic coverage, the summaries' quality is assessed using the fitness function.

Firefly Optimization: The Firefly Algorithm is a nature-inspired algorithm that simulates firefly' flashing patterns. You may think about each firefly as a possible summary when it comes to text summarization. A firefly's quality determines how enticing it is to another, with brighter fireflies denoting better descriptions. Fireflies seek for brighter surroundings, and the algorithm continuously improves the summaries until they reach a point of convergence. The coherence, relevance, and informative-ness of a summary may all be measured using the fitness function[25].

3. FRAMEWORK FOR SINGLE DOCUMENT TEXT SUMMARIZATION

We describe a new framework for metaheuristic-based text summarization of a single document dataset in this section. The framework for the proposed single document text summary is depicted in Figure 2. The framework employs the particle swam, cuckoo, and firefly optimization algorithms and includes the following steps:

1. Text extraction from XML documents.
2. Data pre-processing.
3. Giving sentences different weights based on their relevance.
4. Choose the most important sentences.
5. Using the nature-inspired optimization approach, generate the summary.
6. Assessment of the generated summary.

DUC2001 and DUC2002 are the datasets that were utilized for the experiments[26]. The National Institute of Standards and Technology (NIST) has published these documents in an effort to encourage academics working in the field of natural language processing (NLP) to have presented the news item documents spanning multiple years with the assistance of Newswire and the New York Times. Additionally, NIST gave reference summaries of the single document that was provided; hence, these datasets are now used as benchmarks for text summarization. The data sets are made available in an XML format. This data is then extracted into CSV files in preparation for the subsequent pre-processing and algorithmic method[27]. Converting single document sentences and words into vectors can be done in a number of different ways, some of which include using TF-IDF, thematic score, and sentence position. The TF-IDF values: Term frequency (TF) and inverse document frequency (IDF) are essential statistics that measure term significance in a document and throughout a collection[28][29]. They determine document phrase weight, which indicates importance. Document term frequency (TF) measures term occurrence. It is commonly determined as the term's frequency in the document divided by its word count[30]. Inverse document frequency (IDF) measures phrase rarity across documents. It is the logarithm of the collection's papers divided by those containing the term[29].

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

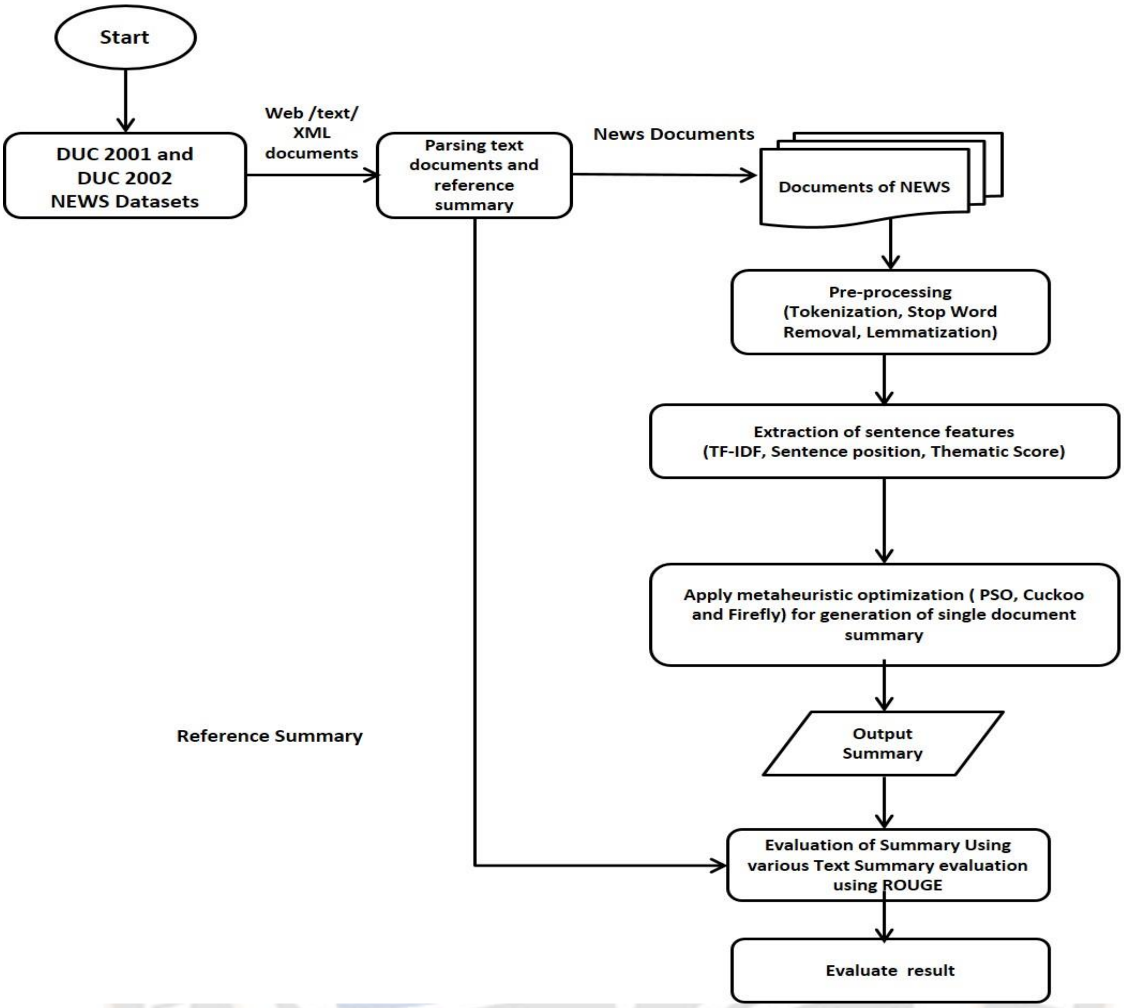

Fig. 2 Framework for metaheuristic based single document text summarization

$$TF_{ij} = n_{ij} / \sum_k n_{ij} \qquad \text{................(1)}$$

$$idf(w) = \log\left(\frac{N}{df_t}\right) \qquad \text{................(2)}$$

The word weight is calculated by multiplying the TF and IDF values, which measures the term's frequency in the document and rarity across the collection.

$$W_{ij} = TF_{ij} * IDF_i \qquad \text{..................(3)}$$

Sentence Position: In this sentence scoring method, sentences were scored based on location of the sentence in the text document[31]. The position of a sentence in a document is often considered to be an important factor in determining its importance. The first and last few sentences of a document typically contain the most important concepts, while the middle sentences are often less important.

$$Sp(sj) = \frac{|j|}{|n|/2} \qquad \text{----------(4)}$$

Where j = number of line of the sentence in the document n = input document sentence count and Sp = Score based on sentence position

Thematic Feature: The phrases[32] that appear most frequently in the text. The top n words that appeared the most frequently were considered to be thematic words[33].

$$Thematic\ score = \frac{No.\ of\ thematic\ words\ in\ sentence}{Total\ No.\ of\ words\ in\ sentence} \qquad ..(5)$$

The purpose of defining the summary scoring function is to obtain a summary that is more accurate as compared to summaries created by humans. The TF-IDF, Sentence position(SP) and Thematic score (TS) are all calculated based on the values of the sentences that are used in the summary scoring function for all nature-inspired metaheuristic optimization techniques. α, β and γ is weight of sentence features. The fitness function is define based on the these sentence features using equation (6)

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

$$Fitness = \frac{(\alpha * TF - IDF \ + \ \beta * SP \ + \gamma * TS\,)}{\alpha \ + \ \beta + \ \gamma} \quad .(6)$$

4. EXPERIMENTAL SETUP AND SUMMARY EVALUATION

In this section, we will begin by defining the parameters that will be used in our proposed algorithm, as well as the characteristics of the datasets and manual summaries that will be required beforehand. After that, we provide a condensed description of our algorithm and compare the summaries produced by our models to the summaries that were used as references (manuals). At each iteration, the parameters are controlled by a few fixed parameters, and this is done in order to find the randomness in the algorithm so that it can achieve a higher convergence rate. Experiments typically make use of the benchmark datasets known as DUC 2001 and DUC 2002. The features of datasets are outlined in Table 1. The Avg. ROUGE score of DUC 2001 and DUC 2002 dataset using three sentence features TF-IDF, Thematic score and sentence position is showing in below Table 2 and Table 3.

Table 1. *Datasets information*

Dataset	No of cluster	Docs Per Cluster	Golden Summaries	Total Documents
DUC 2001	30	~10.3	4 Human Summaries	308
DUC 2002	59	~9.6	4 Human Summaries	567

Table 2. *Avg. ROUGE Result for dataset DUC 2001*

ROUGE Metrics	Result	TF-IDF	Sentence Position	Thematic Score
ROUGE -1	Recall	0.2726	0.2877	0.2987
	Precision	0.2540	0.2754	0.2914
	F-score	0.2607	0.2785	0.2928
ROUGE -2	Recall	0.0904	0.1019	0.1090
	Precision	0.0855	0.0987	0.1044
	F-score	0.0872	0.0992	0.1061
ROUGE -L	Recall	0.2497	0.2683	0.2750
	Precision	0.2327	0.2568	0.2683
	F-score	0.2388	0.2597	0.2696

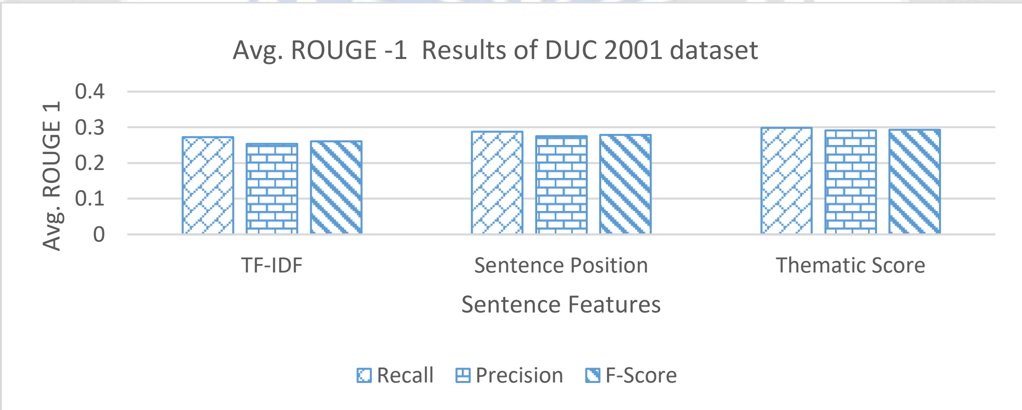

Fig. 3 Avg. ROUGE Result for dataset DUC 2001

Table 3 *Avg. ROUGE Result for dataset DUC 2002*

ROUGE Metrics	Result	TF-IDF	Sentence Position	Thematic Score
ROUGE -1	Recall	0.3154	0.3366	0.3438
	Precision	0.3078	0.3330	0.3515
	F-score	0.3086	0.3317	0.3446
ROUGE -2	Recall	0.1199	0.1360	0.1418
	Precision	0.1223	0.1400	0.1463
	F-score	0.1204	0.1372	0.1433
ROUGE -L	Recall	0.2895	0.3134	0.3182
	Precision	0.2828	0.3102	0.3255
	F-score	0.2833	0.3089	0.3191

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 12 Issue: 1
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

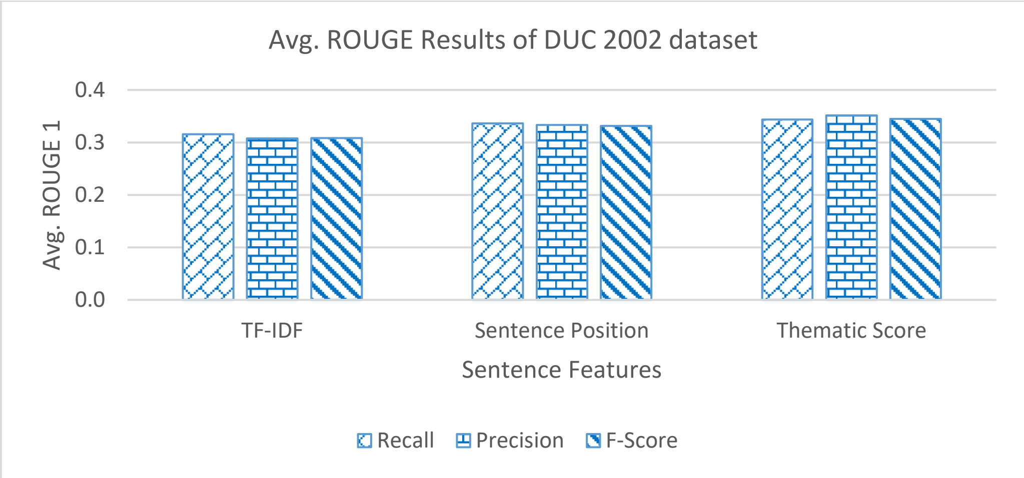

Fig. 4 Avg. ROUGE Result for dataset DUC 2002

Table 4 Avg. *ROUGE Result for dataset DUC 2001 using metaheuristic optimization*

ROUGE Metrics	Result	PSO	Cuckoo Search	Firefly
ROUGE -1	Recall	0.2695	0.3244	0.3012
	Precision	0.2638	0.3125	0.2983
	F-score	0.2642	0.3187	0.3004
ROUGE -2	Recall	0.0885	0.1282	0.1147
	Precision	0.0873	0.0903	0.0998
	F-score	0.0872	0.0909	0.1038
ROUGE -L	Recall	0.2512	0.2784	0.2571
	Precision	0.2458	0.2700	0.2494
	F-score	0.2463	0.2718	0.2511

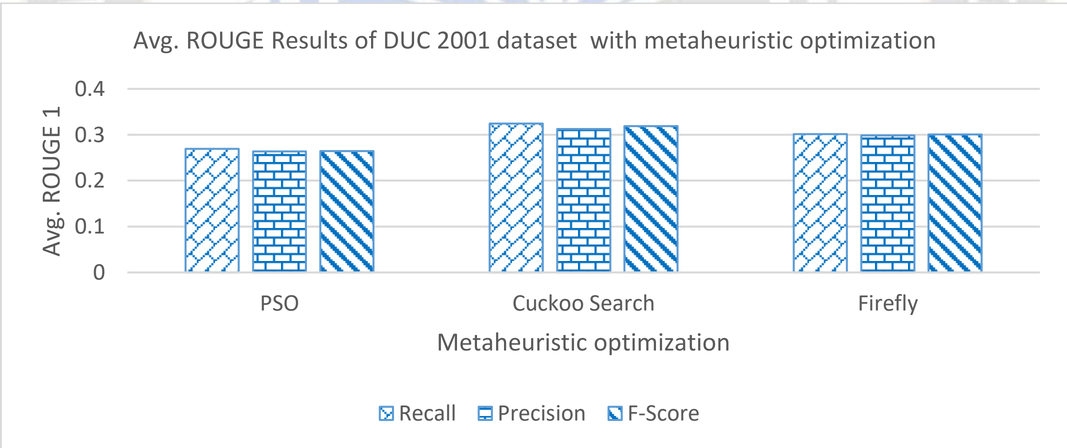

Fig. 5 Avg. ROUGE Result for dataset DUC 2001 using metaheuristic optimization

Table 5 *Avg. ROUGE Result for dataset DUC 2002 using metaheuristic optimization*

ROUGE Metrics	Result	PSO	Cuckoo Search	Firefly
ROUGE -1	Recall	0.2914	0.3451	0.3178
	Precision	0.3071	0.3482	0.3364
	F-score	0.2946	0.3437	0.3227
ROUGE -2	Recall	0.1163	0.1487	0.1324
	Precision	0.1251	0.1535	0.1438
	F-score	0.1194	0.1504	0.1366
ROUGE -L	Recall	0.2754	0.3236	0.2953
	Precision	0.2901	0.3267	0.3126
	F-score	0.2784	0.3223	0.2998

***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

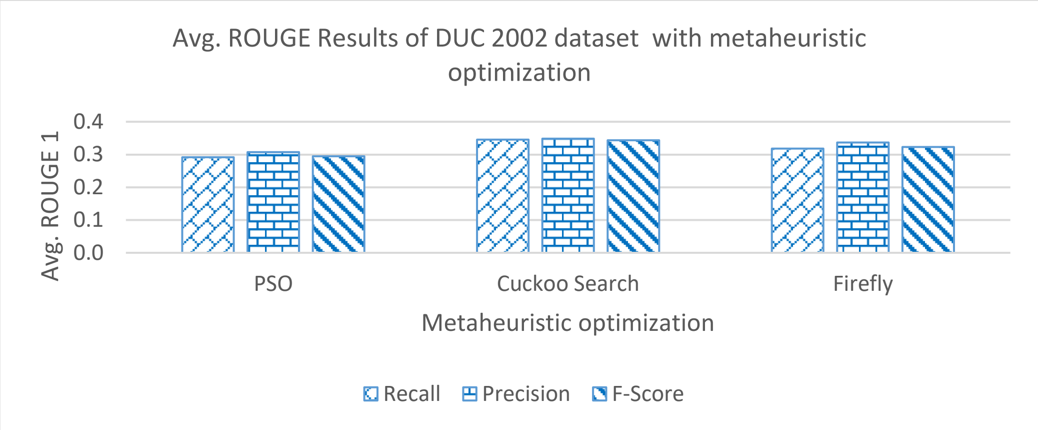

Fig. 6 Avg. ROUGE Result for dataset DUC 2002 using metaheuristic optimization

5. CONCLUSION

In this paper, we investigated the performance of three different metaheuristic algorithms for the purpose of single document text summarization. These methods were Particle Swarm Optimization (PSO), Firefly Algorithm (FA), and Cuckoo Search (CS). According to the results of our tests, the Cuckoo Search and FA algorithm performed better than the PSO algorithm and the CS algorithm when measured against the ROUGE-1, ROUGE-2, and ROUGE-L criteria. This is probably because the FA algorithm is better at searching the search space and locating the best feasible summary text than other algorithms are. In addition, we discovered that the thematic score as well as the cuckoo search were extremely helpful when it came to determining which sentences in the document included significant information. This shows that these traits could be useful for boosting the performance of algorithms that summarized single documents of text.

ACKNOWLEDGMENT

The author expresses sincere gratitude to Ganpat university for their invaluable support for providing computing infrastructure for experiment. This study received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

1. J. R. Thomas, S. K. Bharti, and K. S. Babu, “Automatic keyword extraction for text summarization in e-newspapers,” *ACM Int. Conf. Proceeding Ser.*, vol. 25-26-Augu, 2016, doi: 10.1145/2980258.2980442.
2. S. Mandal, G. K. Singh, and A. Pal, “Single document text summarization technique using optimal combination of cuckoo search algorithm, sentence scoring and sentiment score,” *Int. J. Inf. Technol.*, vol. 13, no. 5, pp. 1805–1813, 2021, doi: 10.1007/s41870-021-00739-2.
3. A. P. Widyassari *et al.*, “Review of automatic text summarization techniques & methods,” *J. King Saud Univ. - Comput. Inf. Sci.*, May 2020, doi: 10.1016/J.JKSUCI.2020.05.006.
4. J. N. Madhuri and R. Ganesh Kumar, “Extractive Text Summarization Using Sentence Ranking,” in *2019 International Conference on Data Science and Communication (IconDSC)*, 2019, pp. 1–3, doi: 10.1109/IconDSC.2019.8817040.
5. P. Verma, S. Pal, and H. Om, “A comparative analysis on Hindi and English extractive text summarization,” *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 18, no. 3, 2019, doi: 10.1145/3308754.
6. J. Steinberger and K. Ježek, “Evaluation measures for text summarization,” *Comput. Informatics*, vol. 28, no. 2, pp. 251–275, 2009.
7. S. Gupta and S. K. Gupta, “Abstractive summarization: An overview of the state of the art,” *Expert Syst. Appl.*, vol. 121, pp. 49–65, 2019, doi: https://doi.org/10.1016/j.eswa.2018.12.011.
8. A. R. Mishra, V. K. Panchal, and P. Kumar, “Extractive Text Summarization - An effective approach to extract information from Text,” in *2019 International Conference on contemporary Computing and Informatics (IC3I)*, 2019, pp. 252–255, doi: 10.1109/IC3I46837.2019.9055636.
9. N. Ibrahim Altmami and M. El Bachir Menai, “Automatic summarization of scientific articles: A survey,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1011–1028, 2022, doi: https://doi.org/10.1016/j.jksuci.2020.04.020.
10. Z. Kerui, H. Haichao, and L. Yuxia, “Automatic text summarization on social media,” *ACM Int. Conf. Proceeding Ser.*, 2020, doi: 10.1145/ 3440084.3441182.
11. R. M. Alguliev, R. M. Aliguliyev, M. S. Hajirahimova, and C. A. Mehdiyev, “MCMR: Maximum coverage and minimum redundant text summarization model,” *Expert Syst. Appl.*, vol. 38, no. 12, pp. 14514–14522, 2011, doi: https://doi. org/ 10.1016/j.eswa.2011.05.033.
12. M. Tomer and M. Kumar, “Multi-document extractive text summarization based on firefly algorithm,” *J. King Saud Univ. - Comput. Inf. Sci.*, 2021, doi: https://doi.org/10.1016/j.jksuci.2021.04 .004.
13. Lucky and A. S. Girsang, “Multi-Objective Ant Colony Optimization for Automatic Social Media Comments Summarization,” *Int. J. Adv. Comput. Sci. Appl.*, vol.

10, no. 3, 2019, doi: 10.14569/ IJACSA.2019.0100352.
14. M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: a survey," *Artif. Intell. Rev.*, vol. 47, no. 1, pp. 1–66, 2017, doi: 10.1007/s10462-016-9475-9.
15. K. Shimada, R. Tadano, and T. Endo, "Multi-aspects review summarization with objective information," *Procedia - Soc. Behav. Sci.*, vol. 27, pp. 140–149, 2011, doi: https://doi.org/ 10.1016/ j.sbspro.2011.10.592.
16. R. S. Selvan and K. Arutchelvan, "Improved Cuckoo Search Optimization Algorithm based Multi-document Summarization Model," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, 2021, pp. 735–739, doi: 10.1109/ICCMC5 1019.2021.9418473.
17. A. Hatamlou, S. Abdullah, and H. Nezamabadi-pour, "A combined approach for clustering based on K-means and gravitational search algorithms," *Swarm Evol. Comput.*, vol. 6, pp. 47–52, 2012, doi: https://doi.org/10.1016/j.swevo.2012.02.003.
18. A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," *Comput. Struct.*, vol. 89, no. 23, pp. 2325–2336, 2011, doi: https://doi.org/10.1016/j.compstruc.2011.08.002.
19. C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," 2004.
20. N. Kadhim, "Heuristic Initialization And Similarity Integration Based Model for Improving Extractive Multi-Document Summarization," *J. Mech. Contin. Math. Sci.*, vol. 14, 2019, doi: 10.26782/jmcms.2019.10.00025.
21. K. K. Mamidala and others, "A Heuristic approach for Telugu text summarization with improved sentence ranking," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 3, pp. 4238–4243, 2021.
22. S. Mandal, G. K. Singh, and A. Pal, "PSO-Based Text Summarization Approach Using Sentiment Analysis," in *Computing, Communication and Signal Processing*, 2019, pp. 845–854.
23. A. M. Al-Zahrani and H. Mathkour, "PSO-based feature selection for arabic text summarization," vol. 21, pp. 1454–1469, 2015.
24. J. Too and S. Mirjalili, "A Hyper Learning Binary Dragonfly Algorithm for Feature Selection: A COVID-19 Case Study," *Knowledge-Based Syst.*, vol. 212, p. 106553, Jan. 2021, doi: 10.1016/j.knosys.2020.106553.
25. R. Ferreira *et al.*, "Assessing sentence scoring techniques for extractive text summarization," *Expert Syst. Appl.*, vol. 40, no. 14, pp. 5755–5764, 2013, doi: https://doi.org/10.1016/j.eswa. 2013.04. 023.
26. "Document Understanding Conferences," *DUC Document Understanding Conferences*. https:// duc.nist.gov/ (accessed Jun. 15, 2022).
27. L. Scanlon, S. Zhang, X. Zhang, and M. Sanderson, "Evaluation of Cross Domain Text Summarization," *SIGIR 2020 - Proc. 43rd Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 1853–1856, 2020, doi: 10.1145/3397271.3401285.
28. M. A. Ullah, S. M. Marium, S. A. Begum, and N. S. Dipa, "An algorithm and method for sentiment analysis using the text and emoticon," *ICT Express*, vol. 6, no. 4, pp. 357–360, Dec. 2020, doi: 10.1016/j.icte.2020.07.003.
29. J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Pérez, "The impact of term-weighting schemes and similarity measures on extractive multi-document text summarization," *Expert Syst. Appl.*, vol. 169, p. 114510, 2021, doi: https://doi.org/10.1016/j.eswa.2020.114510.
30. D. Liu, Y. He, D. Ji, and H. Yang, "Genetic Algorithm Based Multi-document Summarization," in *PRICAI 2006: Trends in Artificial Intelligence*, 2006, pp. 1140–1144.
31. P. B. Baxendale, "Machine-Made Index for Technical Literature—An Experiment," *IBM J. Res. Dev.*, vol. 2, no. 4, pp. 354–361, 1958, doi: 10.1147/rd.24.0354.
32. J. Otterbacher and et al., "Using Content-Based Features to Predict the Quality of Text Summaries," 2013.
33. W. Wali, B. Gargouri, and A. Ben Hamadou, "Enhancing the sentence similarity measure by semantic and syntactico-semantic knowledge," *Vietnam J. Comput. Sci.*, vol. 4, no. 1, pp. 51–60, 2017, doi: 10.1007/s40595-016-0080-2.