[1] K. Adi Narayana Reddy
[2] L. Lakshmi
[3] K. Srinivasa Rao
[4] D. Krishna Madhuri
[5] K Brahma Naidu
[6] K Varapraad Rao

Multi-Document Summarization using Discrete Bat Optimization

Journal of Electrical Systems

Abstract: - With the World Wide Web, we now have a wide range of data that was previously unavailable. Therefore, it has become a complex problem to find useful information in large datasets. In recent years, text summarization has emerged as a viable option for mining relevant data from massive collections of texts. We may classify summarizing as either "single document" or "multi document" depending on how many source documents we are working with. Finding an accurate summary from a collection of documents is more difficult for researchers than doing it from a single document. For this reason, this research proposes a Discrete Bat algorithm Optimization based multi document summarizer (DBAT-MDS) to tackle the issue of multi document summarizing. Comparisons are made between the proposed DBAT-MDS based model and three different summarization algorithms that take their inspiration from the natural world. All methods are evaluated in relation to the benchmark Document Understanding Conference (DUC) datasets using a variety of criteria, such as the ROUGE score and the F score. Compared to the other summarizers used in the experiment, the suggested method performs much better.

Keywords: Text Summarization, Classification, Document Summarization, Discrete Bat Algorithm, ROUGE score.

I. Introduction

Lot of data is available due to the advancement of Technology. To analyse or understand the text data becomes tedious and time consuming. Text summarization is one of the best ways to understand text. Summarization produces the concise summary without loss of original semantic information [1]. Summarization is a time-saving technique that may also be used as a fast reference to your interest in the subject matter at hand. Extractive or abstractive summarization is the method of summary generation [2,3]. An extraction-based summary is one that is created by the most relevant parts of the source material. However, generating stronger sentences from the source text is an important aspect of constructing an abstractive-based summary [4,5]. Extraction-based summaries may be split into two types, general and query emphasis, depending on the focus of the extraction itself [6]. A generic summary is a short description of the main points of the original papers that doesn't include any context or background. In contrast, query-focused summaries only include data that directly answers the questions asked [7,8]. The summarization of text is carried out from single or multiple documents, which results in single-document summarization (SDS) and multi-document summarization (MDS) [9,10]. The single document summarization may not produce effective summary as it does not use most recent or related documents. MDS generates more effective and accurate summaries from multiple documents written in different times and perspectives but the process is more complicated as it contains redundant information [11]. Models struggle to preserve the most important information of complex input sequences while producing a logical, non-redundant, factually consistent, and grammatically accessible summary. As a result, MDS demands models to be more capable of analysing incoming documents, recognizing and integrating consistent information. MDS The search area for multi document summarization is bigger than that for single document summary, making it more difficult to isolate key phrases and sentences. Multi-document summarizing, in this view, is an optimization problem whose solution is an excellent summary of the original documents' most informative phrases. The applications of MDS include summarization of product reviews, news, scientific articles, feedback, Wikipedia articles, medical documents and software activities.

[1, 3, 5, 6] Department of CSE, Faculty of Science and Technology, (IcfaiTech), ICFAI Foundation for Higher Education (IFHE), Hyderabad, Telangana, India

[2, 4] Department of Artificial Intelligence and Data Science, Faculty of Science and Technology, (IcfaiTech), ICFAI Foundation for Higher Education (IFHE), Hyderabad, Telangana, India

In this paper, the contributions are listed as follows:

- A novel Discrete Bat algorithm is proposed for handling binary optimization problems.
- The multi-document summarization problem is represented in the form of binary solutions so that it can be efficiently handled by a discrete bat algorithm.
- The proposed model is evaluated with standard datasets and the renowned ROUGE tool is used for evaluation.

The paper is organized as follows: The related works of multi-document summarization are discussed in section 2. Section 3 defines the problem of multi-document summarization along with its objective function. Section 4 discusses standard and discrete bat algorithms. section 5 discusses experimental analysis carried out to prove the significance of the proposed algorithm. section 6 concludes the paper with its future work enhancements.

II. Literature Survey

Multi-document summarization is carried out using traditional techniques like term frequency-inverse document frequency [], graphs [], latent semantic analysis [], and clustering []. Most of the traditional techniques produce the summary with human crafted features like sentence length, proper nouns, sentence position, sentence-to-sentence cohesion and centroid cohesion.

Compression of several documents, speed of sentence extraction, phrase redundancy, and sentence selection are key difficulties in the creation of meaningful summaries in multi-document summarization. Historically, similar problems have been tackled using statistical methods. However, since the year 2000, many researchers have proposed various global optimization techniques like particle swarm optimization (PSO), differential evolution (DE), genetic algorithm (GA), and so on to enhance the performance of text selection in document summarization. This is because statistical tools for text extraction have significantly poor performance.

Optimization method GA was initially employed in [12] to obtain relevant material based on search and relevant judgements when the summarization challenge is seen as an optimization problem. [13] proposed Ide dec-hi technique, which is one of the finest classical approaches to relevance feedback in the information retrieval issue, this study assesses that the GA also preserved the original order of document. In the future, GA-based programming techniques are utilized for fuzzy retrieval systems, which model the user's requirement to extract information depending on query by performing an off-line adaptive process [14].

By taking into account content coverage and redundancy feature, the authors of Rautray and Balabantaray's 2015 [15] paper describe a general summarizer for a single text based on the PSO technique. The goal function is developed for such a task by averaging the qualities of content coverage and redundancy. In [16], PSO-based single-document summarizer is presented; this one-use objective function as [15], but instead of using sentence weights as inputs to the model, it uses attributes of the text. In [17], the authors suggest an extractive summarizer based on PSO, with objective functions based on ROUGE. The PSO-based summary is also offered in [18], with emphasis on the summary's readability, length, and breadth of information. In [19], a PSO-based multi-document summarizing system is introduced, which employs the notion of sentence clustering by calculating inter-sentence similarity between sentences and sentence-to-document set to accomplish content coverage and variety of summary. Instead, the similarity measure is employed in [20] to achieve content coverage, variety, and summary length across various document sets.

In recent years several significant applications have been explored by evolutionary algorithms [21-24]. And to the best of the author's knowledge there exists no research work carried out on conversion of BAT algorithm to discrete for solving multi-document summarization.

III. Problem Statement

In this section, the multi-document summarization has been defined and the processes that the multiple documents undergo so that it can be addressed by evolutionary optimization algorithms were discussed.

A. *Multi-Document Summarization*

An automated procedure that creates a succinct and complete document from numerous documents is called Multi-Document Summarization. For summarizing the contents of multiple documents into a single concise document that holds the information of complete documents contents can be processed in three phases namely preprocessing, Computation of sentence score and Sentence similarity computation. In this section, all three phases are discussed in detail.

B. Preprocessing

The collection of documents D={D_1,D_2,…,D_N } where D_i denotes the i^th document.

C. Segmentation

Every document D_i, will be subject to sentence fragmentation and it can be represented as D_i={S_i1,S_i2,…,S_in} where S_(i,j)represents the j^th sentence of i^thdocument. And the n represents the maximum number of sentence in document i.

D. Token

Every sentence $S_{i,j}$ are subject to further break out as terms in the sentence and it can be represented as $S_{i,j} = \{t_{ij1}, t_{ij2}, \dots, t_{ijm}\}$ representing the distinct terms in the sentence j of i^{th} document. And m represents the total number of distinct terms in the respective sentence and it varies from sentence to sentence.

Removal of stop words: It is a standard procedure in document summarization where the articulation words such as “a, an, the, etc.” will be removed from the document.

E. Stemming

Stemming is the process of fixing the derived words with its root word. For example, “playing”, “plays”, “played” and all are connected to the root word called “play”. Hence the places where these stem extended words are there in the tokens, it can be replaced with the stem word. This process will be carried out in our preprocessing of multi-document summarization to reduce the time distinct number of words in the token that will impact the complexity of the problem in a huge number.

F. Sentence score Computation

The sentence of every document will be subject to quantification for computation purposes. In this regard, since the representation of multi-document summarization does not need any tracking of which document the sentence comes from, the index of the document can be relaxed from representation. From now on, the sentences of all the documents shall be serialized and the tokens of each sentence will be marked as $t_{i,j}$where i represents the sentence and j represents the j^{th} word of i^{th} sentence.

For each sentence S_i a sum of term frequencies value needs to be computed for quantification of the sentence and it is called as sentence score (S). The computation of sentence score can be represented mathematically as

$$S_{i,j} = T \times \log\left(\frac{n}{n_j}\right) \tag{1}$$

$$T = |t_j| \in S_i \tag{2}$$

Where T represents the term frequency of the term j in sentence i, (i.e. the number of times term j occurs in sentence S_i)

G. Sentence Similarity Computation

The similarity between the sentences can be quantified mathematically as

$$SIM(S_i, S_k) = \frac{\sum_{j=1}^{m} S_{i,j} \times S_{k,j}}{\sqrt{\sum_{j=1}^{m} S_{i,j}^2 \times S_{k,j}^2}} \tag{3}$$

Where the S_i and S_k represents two different documents and j represents the term in the document.

H. Objective Function

The objective of multi-document summarization is to concise the content of multiple documents in a readable form without repetition of contents and with all vital information. All these three are three different objectives, and hence the computational factor of these three objectives to be carried out for every generated summary in different forms.

I. Coverage of Vital Content

The coverage of vital content present in the sentence S_i with respect to the actual output summary (O) can be formulated as

$$f_1(S_i) = SIM(S_i, O) \tag{4}$$

Where i ranges from 1 to n (i.e. the sum of total number of sentences in D).O is a collection of sentences of the final summary and it can be represented as $O = \{S_{O1}, S_{O2}, \dots, S_{Oy}\}$ such that y is the total number of sentences in O.

J. Cohesion between sentences

Similarity between the contents in the sentences can be evaluated as

$$f_2(S_i) = 1 - SIM(S_i, S_k) \quad (5)$$

Where i is the current sentence and $k = 1, \dots, n$ representing all the other sentences in D.

K. Readability

Readability defines the readiness of the document to be the summary of relevant information and it can be represented as

$$f_3(S_i) = SIM(S_i, S_k) \quad (6)$$

Where i is the current sentence and $k = 1, \dots, n$ representing all the other sentences in D.
Summarizing all the objectives of every sentence, the objective of every sentence can be represented as

$$f(S_i) = \sum_{z=1}^{3} f_z(S_i) \quad (7)$$

And the objective formulation of multi-document summarization can be defined as

$$\text{Maximize } \sum_{i=1}^{i=R} f(S_i) \quad (8)$$

Where R is the number of sentences in the final predicted summary.

IV. DISCRETE BAT ALGORITHM FOR MULTI-DOCUMENT SUMMARIZATION

In this section, the standard bat algorithm and the proposed discrete bat algorithm for Multi-Document Summarization are described.

A. Standard BAT algorithm

In 2010, Xin-She Yang presented the Bat technique [25] for addressing continuous optimization issues. It was designed to address problems with single-objective optimization. The foundations of the bat algorithm are as follows. Echolocation is a kind of sonar that bats may utilize to locate prey. Typically, bats locate objects by making a loud noise and listening for the echo. This method is based on bat behavior and takes the following aspects into account:
Bats utilize echolocation to detect distance and can distinguish between food, prey, and obstacles.
From the wavelengthsx_i point, bats fly at a random velocity v_i and volume A_0.The wavelength of each bat may be dynamically adjusted based on the target's distance.The loudness is a dynamic value ranging between A_0 andA_{min}.

B. Motion of Bats

Each bat moves closer to those with better responses. In the interim, the frequency and speed of each bat are updated across the amount of iterations. For the subsequent iteration (t+1), the following adjustments are made to each of the bat:

$$f_i = Fit_{min} + (Fit_{max} - Fit_{min}) \times \beta \quad (9)$$

$$v_i^{t+1} = v_i^t + (x_i^t - x^*) \times f_i \quad (10)$$

$$x_i^{t+1} = v_i^{t+1} + x_i^t \quad (11)$$

Where β is a random integer between 0 and 1, and x^* is the best global solution found from iteration 1 to t.In the bat algorithm, the neighborhood search is conducted using a random walk, which is represented by a random number

$$x_{new} = x_{old} + \varepsilon A^t \quad (12)$$

Where A^t is the average loudness of all bats and ε is a vector with values ranging from -1 to +1.

C. Pulse Emission and loudness

The pulse emission and loudness of a bat are inversely associated; when the bat finds food, its loudness decreases and its pulse emission increases, and vice versa. The pulse emission and loudness are mathematically represented as

$$A_i^{t+1} = \alpha A_i^t \quad (13)$$

$$r_i^t = r_i^0[1 - exp\ (-\gamma t)] \quad (14)$$

Where α and γ are constants.

D. Discrete BAT algorithm

The representation of solution is in binary form where 1 represents the sentence is selected to be placed in the predictive summary and the 0 represents it is not.
Hence the position update policy and velocity update equations can be reframed as

$$v_i^{t+1} = v_i^t \oplus (x_i^t \ominus x^*) \quad (15)$$

$$x_i^{t+1} = v_i^{t+1} \oplus x_i^t \quad (16)$$

Where $\ominus$ represents the Boolean difference operator and $\oplus$ represents the Boolean adder operator.

$$x_{new} = x_{old} \oplus \varepsilon\ A^t \quad (17)$$

Algorithm 1: DiscreteBat algorithm for Multi-Document Summarization (DBAT-MDS)

Input:The parameters $\rho, \alpha, \beta, \gamma$ upper and lower bound
Objective Function f
Set the parameters Pulse Frequency PF_i, Pulse Rates r_i and Loudness A_i
Initialize Pop- Number of Bats, M_t, $t = 1$
for each $i \in 1, \dots, Pop$do
$x_i \leftarrow rand(0,1,n)$
for each $z \in 1{:}\,3$do
$[Fit_{i,z}] \leftarrow f_z(x_i)$
end for
end for
repeat
for each $z \in 1{:}\,3$do
$[GBest_z] \leftarrow x(max([Fit_z]))$
end for
for each $i \in 1, \dots, Pop$do
$Update\ the\ velocity\ using\ Eq.(15)$w.r.t. all objectives
end for
for each $i \in 1, \dots, Pop$do
$Update\ the\ position\ using\ Eq.(16)$w.r.t. all objectives
end for
$y_i \leftarrow$ Initializing Random Solution using Equation (17)
if ($rand$<A_i&& $Fit(x_i) < f(Gbest_z)$) then
$x_i \leftarrow y_i$
Increase r_i
Deduce A_i
end if
$t \leftarrow t + 1$
until ($M_t \geq t$)

OUTPUT: Gbest for objectives z=1,2,3

V. Experimental Analysis

In this part, the experimental setup used to test the proposed model is described, together with the performance measures, dataset, and experimental findings

A. Experimental Setup and Dataset

The proposed algorithm is implemented in MATLAB 2018a version in a computer system with Intel Core i5 processor with 2.1 GHz clock speed with 8 GB RAM and 512 SSD. The datasets used to evaluate the proposed model include Document Understanding Conference (DUC). There are 2 datasets in DUC in which one with 50 clusters and the other with 45 clusters respectively. Each cluster will have 25 documents that are to be summarized to a maximum of 250 words. The average number of sentences in every document in DUC2006 is 30.12 and in DUC2007 it is 37.5 sentences.

B. Performance Metrics

To evaluate the performance of the proposed model, an evaluation metric tool namely ROUGE-1.5.5 is used. Among the methods available in ROUGE for evaluation, we used ROUGE-N where the N represents the N-gram match between predicted and actual summaries.

$$ROUGE - N = \frac{\sum_{S\epsilon O} \quad \sum_{N-gram\in S} \quad C_M}{\sum_{S\epsilon O} \quad \sum_{N-gram\in S} \quad C} \tag{18}$$

Where N represents the N-gram value, C_M represents the maximum number of occurrence of words both in predicted and actual summary and C represents the total occurrence of words in actual summary. And N represents the words count. For example, $N = 1$ represents the single words and $N = 2$ represents the two words together as like actual summary.

Apart from ROUGE-N, we computed the statistical metrics such as F-Score, Precision and Recall using Actual summary (O_A) and predicted summary (O_P).

$$Precision = \frac{|O_A \cap O_P|}{|O_P|} \tag{19}$$

$$Recall = \frac{|O_A \cap O_P|}{|O_A|} \tag{20}$$

$$F - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{21}$$

C. Performance analysis

To prove the performance of the proposed model, the state-of-the-art algorithms are used to compare with respect to the performance factors as stated above. The compared algorithms are Cat Search Optimization algorithm (Cat) [26], Harmony Search algorithm (Har) [20], Particle Swarm Optimization algorithm (Par.Swarm) [27].

Table 1: Comparison of proposed Vs existing algorithms w.r.t. performance matrices

Algorithms	**DUC2006**		
	ROUGE-1	**ROUGE-2**	**F-Score**
Cat	0.4331	0.0914	0.55
Har	0.4246	0.0856	0.4735
Par.Swarm	0.4124	0.0742	0.4465
DBAT-MDS	**0.4377**	**0.0926**	**0.5907**
Algorithms	**DUC2007**		
	ROUGE-1	**ROUGE-1**	**ROUGE-1**
Cat	0.4216	0.4216	0.4216
Har	0.4158	0.4158	0.4158
Par.Swarm	0.4005	0.4005	0.4005
DBAT-MDS	**0.4826**	**0.4826**	**0.4826**

Table 1 shows the performance values of proposed DBAT-MDS vs other existing algorithms for both DUC2006 and DUC2007 datasets. On comparing the results of the proposed with existing algorithms it is evident that the proposed model outperforms the existing techniques. Figure 1 and 2 shows the graphical interpretation of result comparison with respect to DUC2006 and DUC2007 datasets respectively.

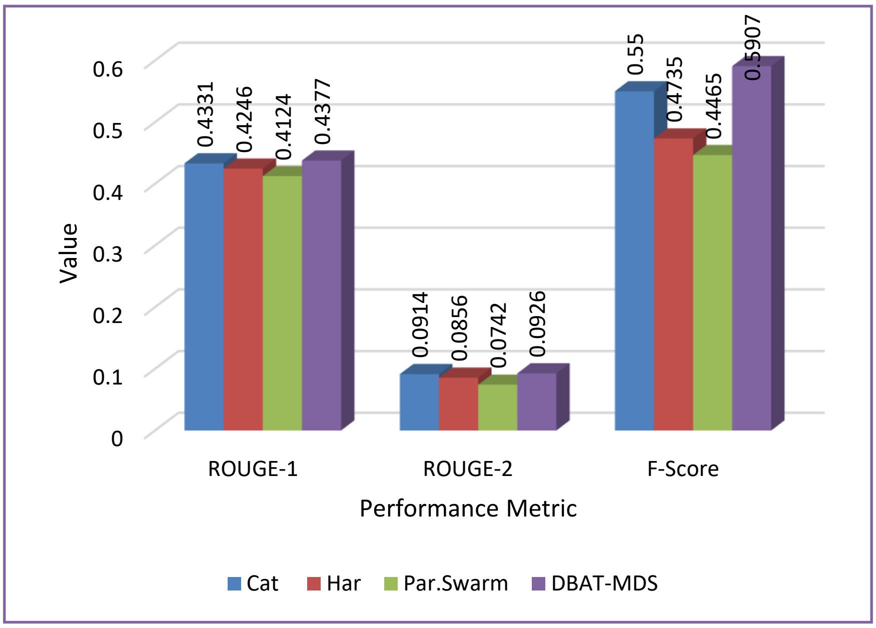

Figure 1: Graphical interpretation of algorithms w.r.t. performance metrics for DUC2006

On comparing the results of ROUGE-1 for DUC2006 dataset, the proposed DBAT-MDS outperforms Cat algorithm with 1%, Harmony Search with 3% and Particle Swarm algorithm with 5.8%. For ROUGE-2, the proposed DBAT-MDS outperforms Cat algorithm with 1.2%, Harmony Search with 7.6% and Particle Swarm algorithm with 19.9%. For F-Score, the proposed DBAT-MDS outperforms Cat algorithm with 6.9%, Harmony Search with 19.8% and Particle Swarm algorithm with 24.4%.

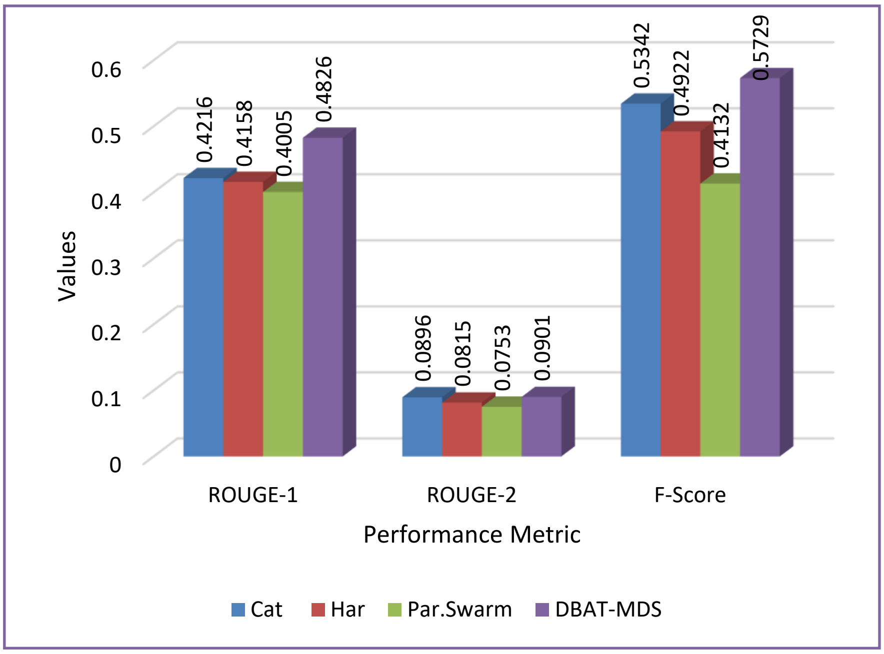

Figure 2: Graphical interpretation of algorithms w.r.t. performance metrices for DUC2007

On comparing the results of ROUGE-1 for DUC2007 dataset, the proposed DBAT-MDS outperforms Cat algorithm with 12.6%, Harmony Search with 13.8% and Particle Swarm algorithm with 17%. For ROUGE-2, the proposed DBAT-MDS outperforms Cat algorithm with 0.6%, Harmony Search with 9.5% and Particle Swarm algorithm with

16.4%. For F-Score, the proposed DBAT-MDS outperforms Cat algorithm with 6.8%, Harmony Search with 14.1% and Particle Swarm algorithm with 27.9%.

Table 2: Comparison of proposed Vs existing algorithms w.r.t. ROUGE on final population individuals for DUC2006 Dataset

Algorithms	ROUGE-1		
	Best	**Worst**	**Mean**
Cat	0.4331	**0.4019**	0.3765
Har	0.4246	0.3842	**0.4103**
Par.Swarm	0.4124	0.3806	0.4008
DBAT-MDS	**0.4377**	0.3648	0.4101
Algorithms	**ROUGE-2**		
	Best	**Worst**	**Mean**
Cat	0.0914	**0.0721**	0.0854
Har	0.0856	0.0636	0.0714
Par.Swarm	0.0742	0.0585	0.0652
DBAT-MDS	**0.0926**	0.0627	**0.0877**

Table 2 shows the ROUGE scores of final population solutions of proposed DBAT-MDS vs other existing algorithms for DUC2006. On comparing the results of the proposed with existing algorithms it is evident that the proposed model outperforms the existing techniques in best results. And the worst results show the impact of diversification. Figure 3 and 4 shows the graphical interpretation of result comparison with respect to DUC2006 for ROUGE-1 and ROUGE-2.

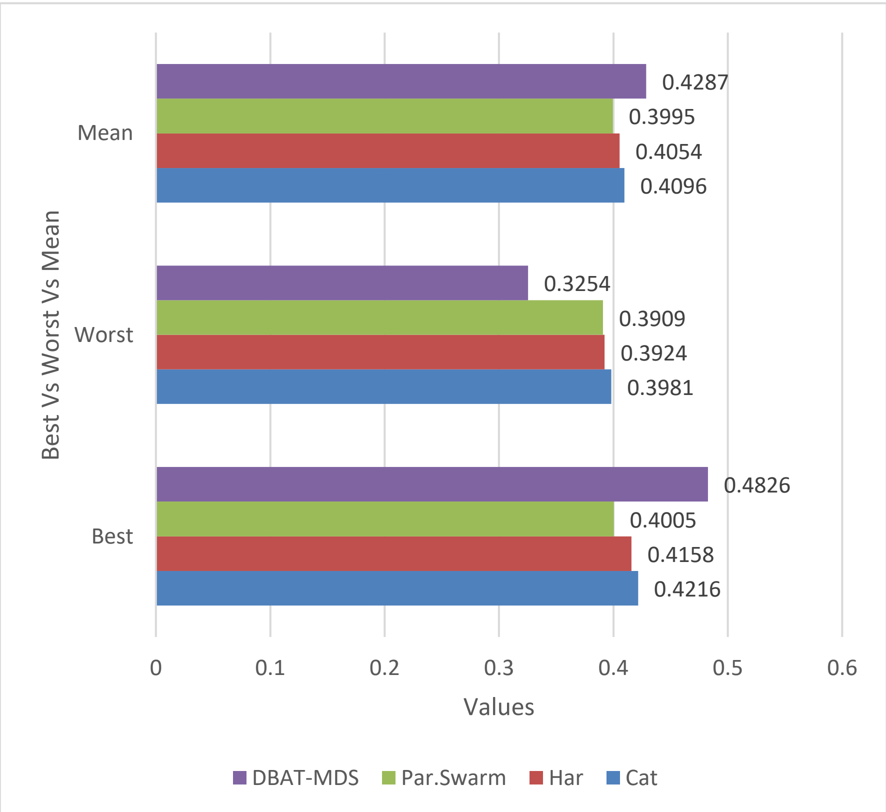

Figure 3: Graphical interpretation of algorithms w.r.t. ROUGE-N values of final population for ROUGE-1 on DUC2006

The worst case of proposed model in ROUGE-1 for DUC2006 dataset is deviated by 16% which shows the diversity is collection of results throughout the search. And the average case of proposed model, falls under the positive curve of the proposed model which intents to show that the proposed model holds more number of optimal results at the end of the search.

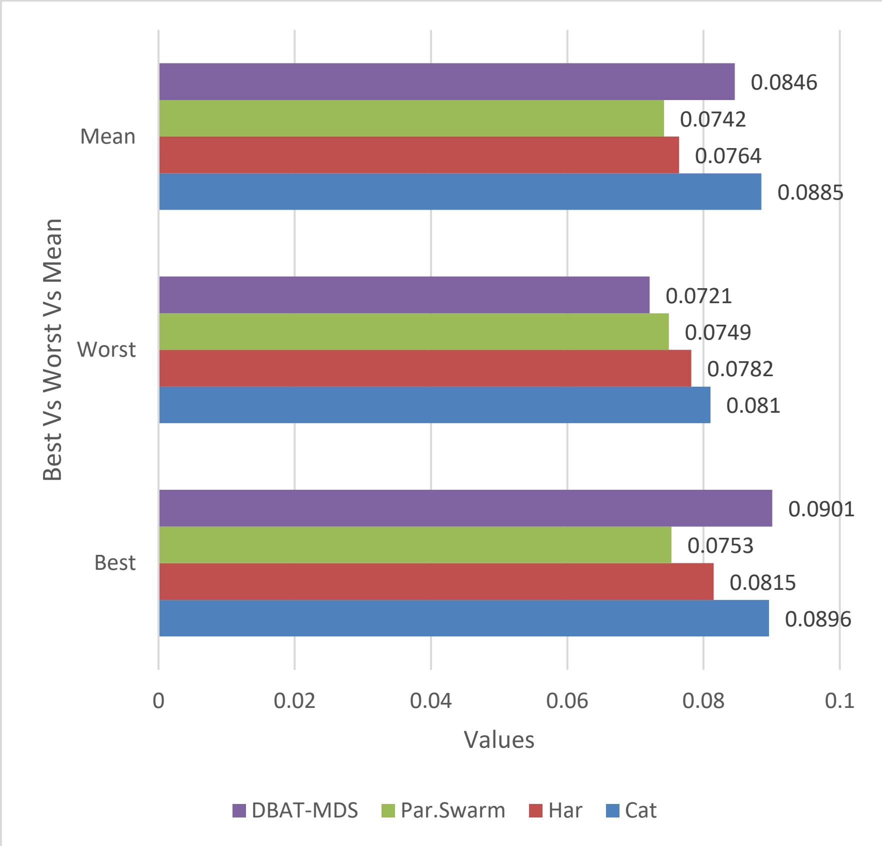

Figure 4: Graphical interpretation of algorithms w.r.t. ROUGE-N values of final population for ROUGE-2 on DUC2006

The worst case of proposed model in ROUGE-2 for DUC2006 dataset is deviated by 32% which shows the diversity is collection of results throughout the search. And the average case of proposed model, falls under the positive curve of the proposed model which intents to show that the proposed model holds more number of optimal results at the end of the search.

Table 3: Comparison of proposed Vs existing algorithms w.r.t. ROUGE on final population individuals for DUC2007 Dataset

Algorithms	**ROUGE-1**		
	Best	**Worst**	**Mean**
Cat	0.4216	**0.3981**	0.4096
Har	0.4158	0.3924	0.4054
Par.Swarm	0.4005	0.3909	0.3995
DBAT-MDS	**0.4826**	0.3254	**0.4287**
Algorithms	**ROUGE-2**		
	Best	**Worst**	**Mean**
Cat	0.0896	**0.0810**	**0.0885**
Har	0.0815	0.0782	0.0764
Par.Swarm	0.0753	0.0749	0.0742
DBAT-MDS	**0.0901**	0.0721	0.0846

Table 3 shows the ROUGE scores of final population solutions of proposed DBAT-MDS vs other existing algorithms for DUC2007. On comparing the results of the proposed with existing algorithms it is evident that the proposed model outperforms the existing techniques in best results. Figure 5 and 6 shows the graphical interpretation of result comparison with respect to DUC2007 for ROUGE-1 and ROUGE-2.

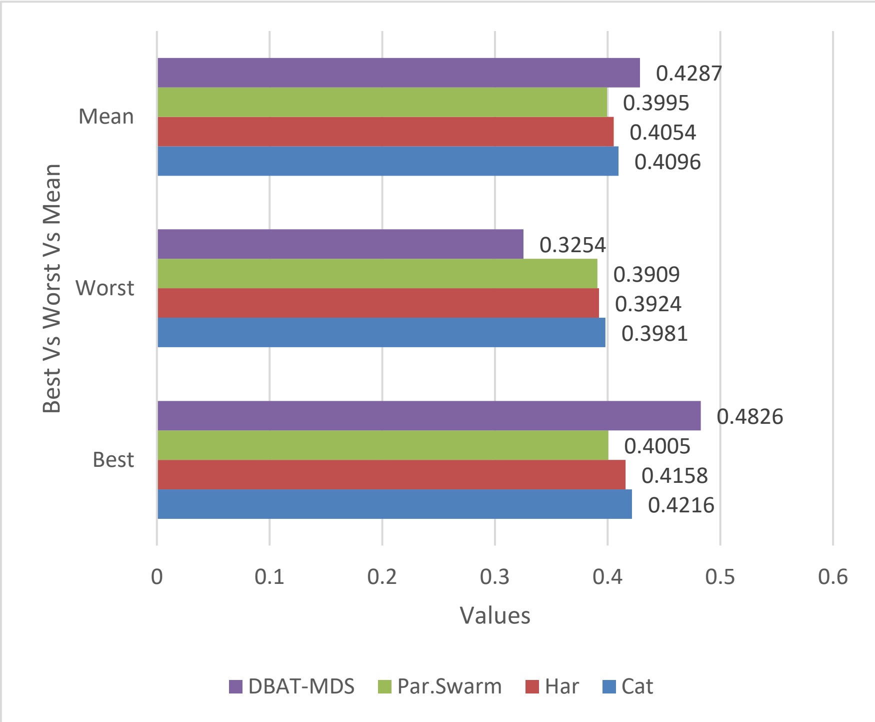

Figure 5: Graphical interpretation of algorithms w.r.t. ROUGE-N values of final population for ROUGE-1 on DUC2007

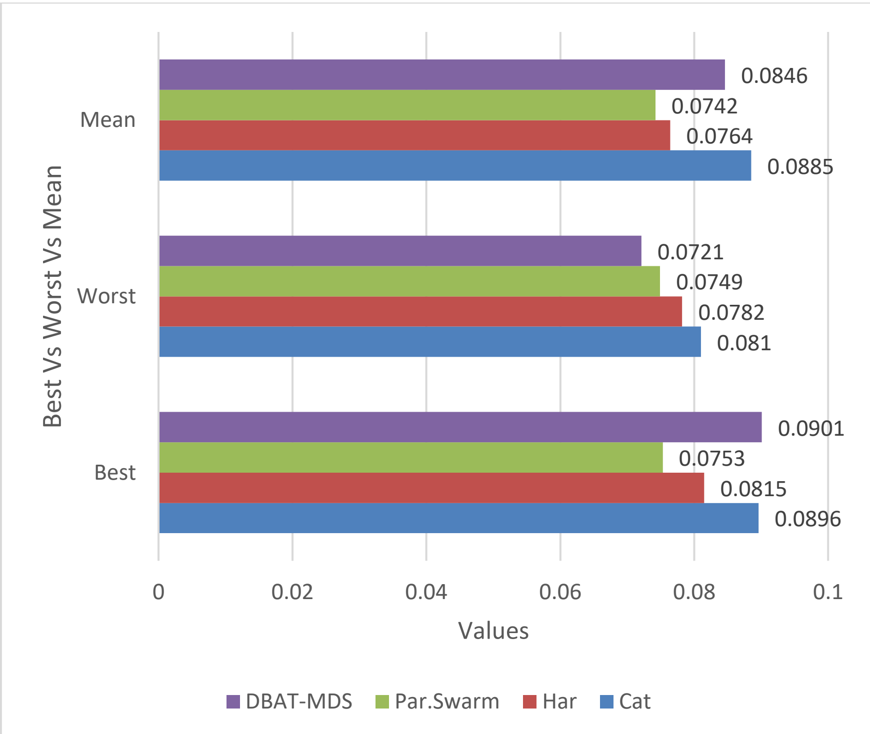

Figure 6: Graphical interpretation of algorithms w.r.t. ROUGE-N values of final population for ROUGE-2 on DUC2007

The worst case of proposed model in ROUGE-1 for DUC2007 dataset is deviated by 32% which shows the diversity is collection of results throughout the search. The worst case of proposed model in ROUGE-2 for DUC2007 dataset is deviated by 20% which shows the diversity is collection of results throughout the search.

VI. Conclusion

In order to generate a standard extractive summary, this research focuses on a Discrete bat algorithm based multi document summarizer (DBAT-MDS). All summarizers are tested on a standard DUC dataset and their performance is compared using the ROUGE and F scores. In light of the above, it is safe to say that DBAT-MDS outperforms Harmoney, particle swarm, and the cat optimization method when it comes to generating summaries. Since the DBAT-MDS issue can only be solved with an evolutionary strategy, that strategy's computing time and regulating parameters are its only constraints. The experimental controls used are completely data-driven. Therefore, in our future study, we will investigate a more methodical approach to parameter tuning.Other capable nature-inspired algorithms may also be used to assess this method's efficacy.

References

[1] R.M. Aliguliyev, A new sentence similarity measure and sentence based extractive technique for automatic text summarization, Expert Syst. Appl. 36 (4) (2009) 7764–7772.

[2] K. Ježek, J. Steinberger, Automatic text summarization (the state of the art 2007 and new challenges), in: Proceedings of Znalosti, pp. 1–12.

[3] E. Lloret, M. Palomar, Text summarisation in progress: a literature review, Artif. Intell. Rev. 37 (1) (2012) 1–41.

[4] M. Mendoza, S. Bonilla, C. Noguera, C. Cobos, E. León, Extractive single-document summarization based on genetic operators and guided local search, Expert Syst. Appl. 41 (9) (2014) 4158–4169.

[5] H. Oliveira, R. Ferreira, R. Lima, R.D. Lins, F. Freitas, M. Riss, S.K. Simske, Assessing shallow sentence scoring techniques and combinations for single and multi-document summarization, Expert Syst. Appl. 65 (2016) 68–86.

[6] R.M. Alguliev, R.M. Aliguliyev, M.S. Hajirahimova, GenDocSum+ MCLR: Generic document summarization based on maximum coverage and less redundancy, Expert Syst. Appl. 39 (16) (2012) 12460–12473.

[7] I. Mani, M.T. Maybury, Advances in Automatic Text Summarization, MIT Press, 1999.

[8] X. Wan, J. Xiao, Exploiting neighborhood knowledge for single document summarization and keyphrase extraction, ACM Trans. Inf. Syst. (TOIS) 28 (2) (2010) 8.

[9] M.A. Fattah, F. Ren, GA, MR, FFNN, PNN and GMM based models for automatic text summarization, Comput. Speech Lang. 23 (1) (2009) 126–144.

[10] R. Rautray, R.C. Balabantaray, A. Bhardwaj, Document summarization using sentence features, Int. J. Inform. Retr. Res. (IJIRR) 5 (1) (2015) 36–47.

[11] R.M. Alguliev, R.M. Aliguliyev, N.R. Isazade, Multiple documents summarization based on evolutionary optimization algorithm, Expert Syst. Appl. 40 (5) (2013) 1675–1689.

[12] M.D. Gordon, Probabilistic and genetic algorithms for document retrieval, Commun. ACM 31 (10) (1988) 1208–1218.

[13] C. López-Pujalte, V.P. Guerrero-Bote, F. de Moya-Anegón, Order-based fitness functions for genetic algorithms applied to relevance feedback, J. Am. Soc. Inf. Sci. Technol. 54 (2) (2003) 152–160.

[14] [24] O.C. García, F. de Moya Anegón, C. Zarco, A GA-P algorithm to automatically formulate extended Boolean queries for a fuzzy information retrieval system, Mathw. Soft Comput. 7 (2) (2000) 309–322

[15] R. Rautray, R.C. Balabantaray, Comparative study of DE and PSO over document summarization, in: Intelligent Computing, Communication and Devices, Springer, India, 2015, pp. 371–377

[16] M.A. Fattah, F. Ren, GA, MR, FFNN, PNN and GMM based models for automatic text summarization, Comput. Speech Lang. 23 (1) (2009) 126–144

[17] H. Asgari, B. Masoumi, O.S. Sheijani, Automatic text summarization based on multi-agent particle swarm optimization, in: 2014 Iranian Conference on Intelligent Systems (ICIS), IEEE, 2014, pp. 1–5

[18] M.S. Binwahlan, N. Salim, L. Suanmali, Swarm based text summarization, in: International Association of Computer Science and Information Technology-Spring Conference, IACSITSC'09, IEEE, 2009, pp. 145–150.

[19] R.M. Alguliev, R.M. Aliguliyev, C.A. Mehdiyev, Sentence selection for generic document summarization using an adaptive differential evolution algorithm, Swarm Evol. Comput. 1 (4) (2011) 213–222.

[20] R.M. Alguliev, R.M. Aliguliyev, M.S. Hajirahimova, C.A. Mehdiyev, MCMR: Maximum coverage and minimum redundant text summarization model, Expert Syst. Appl. 38 (12) (2011) 14514–14522.

[21] Thirugnanasambandam, K., Rajeswari, M., Bhattacharyya, D., & Kim, J. Y. (2022). Directed Artificial Bee Colony algorithm with revamped search strategy to solve global numerical optimization problems. Automated Software Engineering, 29(1), 1-31

[22] Raghav, R. S., Thirugnanasambandam, K., Varadarajan, V., Vairavasundaram, S., & Ravi, L. (2022). Artificial Bee Colony Reinforced Extended Kalman Filter Localization Algorithm in Internet of Things with Big Data Blending Technique for Finding the Accurate Position of Reference Nodes. Big Data, 10(3), 186-203.

[23] Thirugnansambandam, K., Bhattacharyya, D., Frnda, J., Anguraj, D. K., &Nedoma, J. (2021). Augmented Node Placement Model in t-WSN Through Multi objective Approach. Comput. Mater. Contin. CMC Tech Sci. Press, 69, 3629-3644.

[24] Thirugnanasambandam, K., Raghav, R. S., Anguraj, D. K., Saravanan, D., &Janakiraman, S. (2021). Multi-objective Binary Reinforced Cuckoo Search Algorithm for Solving Connected Coverage target based WSN with Critical Targets. Wireless Personal Communications, 121(3), 2301-2325.

[25] Yang, Xin-She. "A new metaheuristic bat-inspired algorithm." Nature inspired cooperative strategies for optimization (NICSO 2010). Springer, Berlin, Heidelberg, 2010. 65-74.

[26] Rautray, Rasmita, and Rakesh Chandra Balabantaray. "Cat swarm optimization based evolutionary framework for multi document summarization." Physica a: statistical mechanics and its applications 477 (2017): 174-186.

[27] E. Shareghi, L.S. Hassanabadi, Text summarization with harmony search algorithm-based sentence extraction, in: Proceedings of the 5th International Conference on Soft Computing as Transdisciplinary Science and Technology, ACM, Cergy-Pontoise, France, 2008, pp. 226–231.