



## A Hybrid Text Summarization Approach Using Neural Networks and Metaheuristic Algorithms

Abdulrahman Mohsen Ahmed Zeyad<sup>\*ID</sup>, Arun Biradar<sup>ID</sup>

School of Computer Science and Engineering, REVA University, Bangalore 560064, India

Corresponding Author Email: [amzeyad@outlook.com](mailto:amzeyad@outlook.com)

<https://doi.org/10.18280/ijss.130310>

### ABSTRACT

**Received:** 7 May 2023

**Accepted:** 24 June 2023

#### Keywords:

*text summarization, Recurrent Neural Networks, metaheuristic algorithms*

The rapid growth of text data on the Internet requires effective automatic text summarization techniques. This study proposes a hybrid text summarization approach that combines a Multi-hidden Recurrent Neural Network and a mayfly-harmony search algorithm. The neural network generates a feature vector for each sentence. The mayfly-harmony search algorithm then optimizes the feature weights to extract the most relevant sentences for the summary. This manuscript capacity provides essential information and expertise that can be effectively summarised using Efficient Abstractive Text Summarising (EATS) techniques. This project aimed to extract informative summaries from various articles by utilizing regularly utilized handcrafted elements from literature. A Multi-hidden Recurrent Neural Network (MRNN) was used to generate a feature vector, and a new feature assortment strategy called Mayfly-Harmony Search (MHS) was applied for feature extraction. The number of sentences, word frequency, title similarity, term frequency-inverse sentence frequency, sentence location, sentence length, sentence-sentence similarity, sentence phrases, proper nouns, n-gram co-occurrence, and document length were the features used. By taking diverse Mayfly Algorithm explanations found from other expanses of the search space and processing them with Harmony Search, the suggested hybrid of the Mayfly Algorithm and Harmony Search was employed to produce superior results.

## 1. INTRODUCTION

Text summarization is the process of generating a concise summary from a lengthy text document. It can be classified into extractive and abstractive summarization. Text summarization techniques have many applications such as generating news headlines, search snippet generation, and simplifying access to information [1].

To solve the complex problems in text summarization, bio-inspired metaheuristic algorithms and neural networks can be applied [2]. We propose using a Multi-hidden Recurrent Neural Network (MRNN) to generate a feature vector for each sentence. A novel mayfly-harmony search (MHS) algorithm is then applied to optimize the feature weights for selecting the most relevant sentences [3].

To illustrate the advantages of the text-MRNN-MHS-text summarization technique, we conduct tests on three benchmarks Document Understanding Conference (DUC) datasets [4]. Studies reveal our method performs better than several extractive summarizations and cutting-edge abstractive baselines [5]. A good summary is based on the ability of demonstration and comprehension, whereby empirical evidence also supports the importance of MRNN-MHS of concepts and events in representing and understanding documents [6].

The objectives of this study are:

- To extract informative summaries from multiple documents using textual features and metaheuristic algorithms

- To propose a hybrid MRNN-MHS text summarization approach that combines neural networks and optimization algorithms [7].

The main contributions of this study are:

- A MRNN model to generate sentence feature vectors.
- A MHS algorithm for optimizing feature weights.
- Evaluation on benchmark DUC datasets showing competitive performance [8].

## 2. LITERATURE SURVEY

The text summarization process can be broadly classified into two major categories: abstractive and extractive text summarization. Abstractive summary involves in creating new sentences in summary and are not from the input documents. Extractive summary fetches most relevant keywords from the input document and lists according to the relevance of the context.

The peculiar complexity of the Malayalam language, such as the frequent usage of linking verbs or copulas in sentences and the lack of predicate agreements between subject and verbs about person, gender, and number, are highlighted in the study [9]. The classic S2S attention model is trained using data created by translating Malayalam versions of freely downloadable BBC news corpus. Two frameworks with stacked-LSTM encoders are used in the study [10], notably the time-distributed stacked-LSTM and the attention-based stacked-LSTM models. Data were manually collected from

various news websites and internet platforms to train the Hindi and Marathi ABS frameworks.

To direct, control, and enhance faithfulness, the study [11] uses a variety of external knowledge inputs. As a result, they adopt a Transformer instantiated with BERT or BART for the encoder and decoder components and create a general framework called GSum. Two encoders in their model encode the input text, and the guide signal and a decoder handle both when producing outputs. Oracle extractions are employed during training time to forecast guidance signals, and a MatchSum model is used during testing time.

The ensemble method is used to show that abstractive ATS representations can produce precise that are superior to their original outputs [12]. By developing a two-stage learning general framework, Refactor, where the collective representations share similar limitations, solves various gap difficulties of reranking and stacking approaches. Refactor is specifically taught to find potential summaries from sentences in documents first (pre-trained Refactor) and then trained to find likely premises from various base perfect productions (fine-tuned Refactor). To improve the performance of top-performing abstractive ATS models on the CNN/DM and XSum datasets, Refactor uses the complementarity of models like BART, PEGASUS, and GSUM.

Despite recent improvements, the System's Generated Summaries' Quality is Still Variable. The generated summaries may not include the most pertinent elements or do not capture the most crucial information from the original document. To learn how to produce accurate and meaningful summaries, abstractive summarization algorithms need a substantial amount of training data. Yet, getting such information can be challenging and expensive, particularly for specialized domains.

When a significant number of news stories need to be summarised in real-time, this might be problematic for applications like news summarization. Therefore, it is critical to discover an explanation for the complex of efficient

abstractive manuscript summarization. An efficient and accurate system for abstractive text summaries would benefit various uses, such as news summarization, academic research, and corporate intelligence.

### 3. PROPOSED SYSTEM

An automated multi-document summarising method creates a condensed and comprehensive document from numerous articles. The three steps of the multi-document summarising process include pre-processing, generating feature vectors, extracting features, and producing summaries. An overview of the summarising technique is shown in Figure 1. Many documents are submitted into the summary system, including Docu1, Docu2..., and DocuN. The initial pre-processing of the documents yields the output, which is then passed through feature extraction, feature vector generation, and summary representations to produce the final summary.

The types and lengths of the input documents are varied for the multi-document summarizing task, and they can be loosely categorized into three groups:

Stop word removal, stemming, and sentence splitting make up the pre-processing stage of our suggested solution. The stop-words removal removes the words that are most frequently used but don't have a clear meaning, including articles, prepositions, conjunctions, interrogatives, assisting verbs, etc. Inflected terms stem by going back to their original form. Each sentence in a document is distinguished by sentence breaks [13].

The pre-processed multi-documents are combined in this section by eliminating unnecessary sentences and ensuring that all relevant topics are covered. The sentence resemblance is first designed as shown below (1).

$$S(t_k, t_{k'}) = \alpha \times S_{WMD}(t_k, t_{k'}) + (1 - \alpha) \times S_{NGD}(t_k, t_{k'}) \quad (1)$$

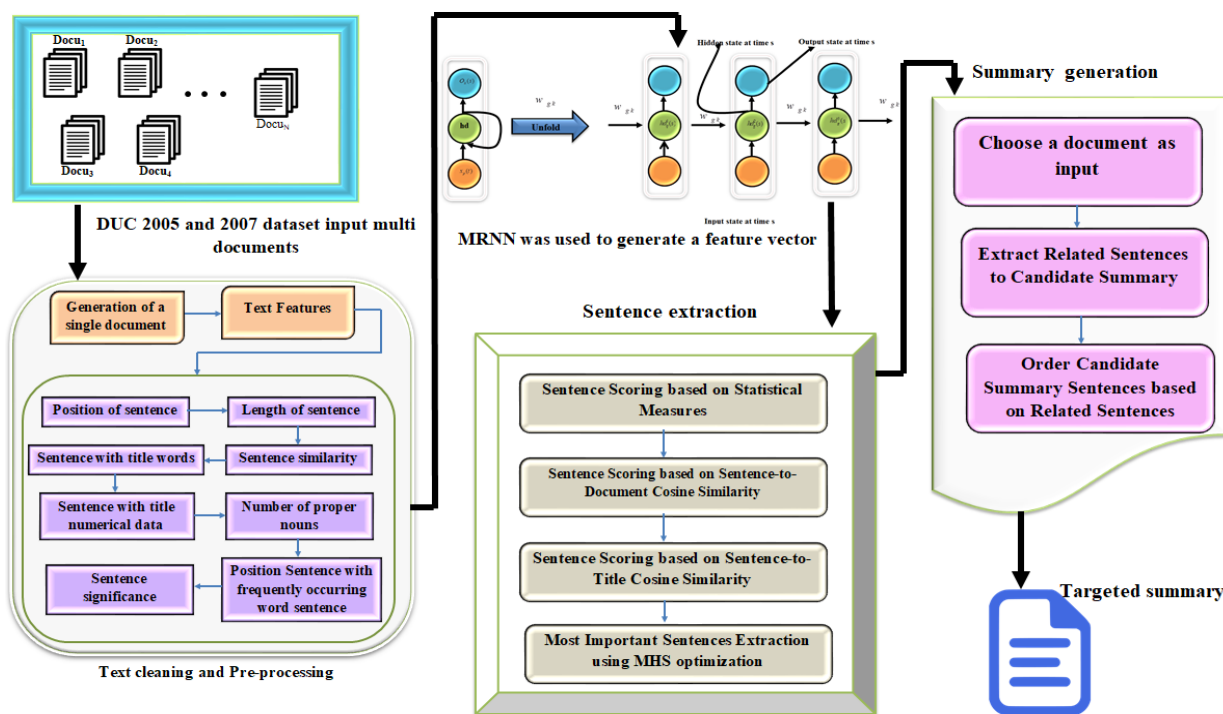


Figure 1. Proposed method of MRNN-MHS

where,

$$S_{WMD}(t_k, t_{k'}) = \frac{\sum_{x_q \in t_k} \sum_{x_{q'} \in t_{k'}} S_{WMD}(x_q, x_{q'})}{|t_k| \cdot |t_{k'}|} \quad (2)$$

And

$$S_{NGD}(t_k, t_{k'}) = \frac{\sum_{x_q \in t_k} \sum_{x_{q'} \in t_{k'}} S_{NGD}(x_q, x_{q'})}{|t_k| \cdot |t_{k'}|} \quad (3)$$

$x_q$  and  $x_{q'}$  are terms in  $t_k$  and  $t_{k'}$ , which are two phrases, respectively.

We have employed a linear grouping of WMD and NGD-founded comparison functions to evaluate the similarities between two condemnations and examine how these functions affect the efficacy of the summarization strategy.

Reducing redundancy and increasing coverage are the goals of single document creation. To account for this, sentences more similar to the document than the average sentence similarity is regarded as having high coverage. We express it mathematically as follows:

$$S(t_k) + S(t_{k'}) \geq \frac{2}{m'} \sum_{k=1}^{m'} S(t_k) \quad (4)$$

If there is less than a threshold value  $P \in (0,1)$  of similarity between two sentences, they are measured to be non-redundant condemnations. The non-redundancy criterion for two curses is distinct below.

$$S(t_k, t_{k'}) \leq P \quad (5)$$

Divide Inequality (4) by Inequality (5) to get (6)

$$\frac{S(t_k) + S(t_{k'})}{S(t_k, t_{k'})} \geq \frac{\frac{2}{m'} \sum_{k=1}^{m'} S(t_k)}{P} \quad (6)$$

The sentence pairs are chosen to be a part of the single document creation if they satisfy inequality (6).

### 3.1 Text features

The sentences in Document D are graded in this part based on various text properties. Give the text feature  $x$  the symbol  $f_x$ . The score of the manuscript feature for condemnation is then shown by  $Rf_x$ . The following provides a succinct description of the text topographies employed in this work.

**Position of Sentence:** It is a crucial component of sentence extraction. Usually, a manuscript's opening and last condemnations contain more important info than the rest. We compute the score as shown in Eq. (7) to account for this factor.

$$Rf_1(t_k) = \frac{|N/2 - k|}{N/2} \quad (7)$$

**Length of the Sentence:** The concise and very long

sentences are filtered out using this function. The following procedure can be used to get the average distance of a batch of condemnations.

$$AV(R) = \frac{\min(|t_k|) + \max(|t_k|)}{2} \quad (8)$$

where,  $AL(S)$  is the document's average sentence length, and  $S$  is the collection of sentences. The equation is used to determine the phrase length score using the average size (9)

$$Rf_2(t_k) = 1 - \frac{|AV(R) - |t_k||}{\max(|t_k|)} \quad (9)$$

**Sentence Similarity:** Discovery of the typical info in the sentences will allow you to score them. It might be stated that it views sentences with more shared information as being more pertinent.

$$Rf_1(t_k) = \frac{\sum_{k'=1, k \neq k'}^{|d_j|} (S(t_k, t_{k'}))}{|t_k|} \quad (10)$$

**Sentence with Title Words:** Sentences with title words highlight the most pertinent sentences in a document. The formula for calculating this score is provided below.

$$Rf_3(t_k) = \frac{\sum_{W \in t_k} (Count(gram_L))}{|W| \cdot |t_k|} \quad (11)$$

where,  $|W|$  represents the entire amount of title arguments,  $W$  represents the title term.

**Sentence with Numerical Data:** In general, sentences with numerical data point to vital information that will probably be included in the summary. The calculation looks like this.

$$Rf_4(t_k) = \frac{\sum_{nd \in t_k} (Count(gram_L))}{|nd| \cdot |t_k|} \quad (12)$$

where,  $nd$  stands for the mathematical information and  $|nd|$  for all the numerical data combined.

**The Number of Proper Names:** An adequate noun designates a particular individual, location, or institution. It is the one trait that may express more information in a phrase than any other. The equation looks like this.

$$Rf_5(t_k) = \frac{\sum_{PN \in t_k} (Count(gram_L))}{|PN| \cdot |t_k|} \quad (13)$$

where, "PN" stands for "proper noun."

**Sentence with Frequently Occurring Words:** Often occurring words can also describe the document's title, which contains essential information. The top 30% of this work's most often occurring terms are regarded as frequent words. We divided the entire amount of frequent arguments by amount as the whole of views to normalize them. The calculation looks like this.

$$Rf_6(t_k) = \frac{\sum_{fw \in t_k} (Count(gram_L))}{|fw| \cdot |t_k|} \quad (14)$$

where,  $fw$  stands for the familiar words.

**Sentence Significance:** Sentence significance provides information on the contribution of the sentence. Briefing the importance of each word part of the judgment yields the phrase's significance score. Thus, the *tf-idf* approach is used to determine the extent of arguments. The following formula is used to determine a sentence's relevance score.

$$Rf_7(t_k) = \frac{\sum Sig(x_{qk})}{|t_k|} \quad (15)$$

where,

$$Sig(x_{qk}) = fe_q \times \log \frac{m'}{m_q} \quad (16)$$

Here,  $Sig(x_{qk})$  denotes the importance of the word  $x_q$  in the condemnation  $t_k$ . The amount of sentences in  $np$  is  $x$ .

A MRNN is used to produce a feature vector. Several text qualities with their optimum weights were used to score the phrases in the text summary task to identify the pertinent ones. This ensures that the system-generated summaries are accurate. This technique may produce a text feature vector exceptionally effectively. By interpreting the relevant portion at each generational step, an attention-encoder will provide some conditioning to ensure that only the input words are focused. This method allows for creating a shorter version of a given statement while maintaining its meaning.

The Mayfly algorithm and Harmony search are combined to get superior results. This technique entailed extracting several solutions from diverse districts of the search space using the Mayfly Algorithm and processing them with Harmony Search. Cosine similarity and statistical traits are also used in the hybrid Mayfly Algorithm and Harmony Search (MHS) algorithm. In additional arguments, a mixture of four data analyses and two cosine similarity-based metrics was used as the impartial purpose to assess each sentence's quality, to be maximized during the evolution process. As a result, the recommended approach's summarising problem is a maximizing problem, and the MHS aims to select the original text phrases with the top scores.

The purpose of immediate illustration is to produce instantaneous document sets containing valuable info. The most effective sentence selection strategy chooses the key sentences that convey the summary by associating the condemnation revealing score produced by the optimization procedure with deference to a prearranged threshold assessment.

### 3.2 MRNN to generate a feature vector

Recurrent Neural Networks (RNNs) are viewed as neural networks with bidirectional data flow in contrast to Multi-Layer Perceptrons, which receive information from lower layers and feed it to developed ones. From early processing steps to earlier processing stages, the data flow propagates. We implement Jeff Elman's proposed small Recurrent Neural

Network in this investigation. The model depicted in Figure 2 makes use of three network layers. Context neurons record the output of each hidden neuron at a time  $(s - 1)$  and then send it back to the hidden layer at a time  $(s)$ . Context neurons continuously hold copies of the hidden neurons' initial values until a parameter-updating rule is functional at the period due to propagation across the repeat connections  $(s - 1)$ .  $(s - 1)$ . As a result, a set of the state summarising earlier inputs is retained and acquired by the network model.

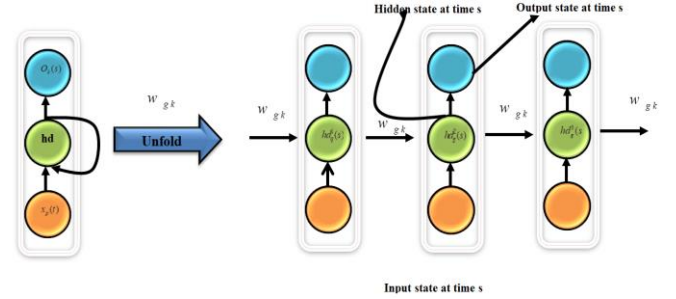


Figure 2. A structure of RNN

The perceptron model used in the RNN-based neural network comprises two hidden layers and two setting layers (one framework for each hidden layer). What follows is a survey of the perfect: As shown in the study [14], the network consists of 18 input neurons, ten first-hidden neurons, ten first-context neurons, ten second-hidden neurons, and one output neuron. There are no differences between the neurons in the first and second context layers, which are concealed (see calculations lower).

$$CT_1^1(t) = hd_q^1(s - 1) \quad (17)$$

$CT_1^1(t)$  equals  $hd_q^1(s - 1)$  if it is the  $l$ th neuron in the first situation layer at period  $s$ .

$$CT_m^2(t) = hd_g^2(s - 1) \quad (18)$$

$CT_m^2(t)$  either equal or indicates the  $l$ th neuron in the second contextual layer at period  $t$ .

In the preceding example,  $hd_g^2(s - 1)$  acted as the  $j$ th neuron in the second hidden layer. The feed-forward to the first hidden layer can be expressed in the following way:

$$hd_q^1(s) = f \left( \sum_p v_{pq}^1 x_p(s) \right) + f \left( \sum_1^{Con^1} u_{pq}^1 CT_1^1(s) \right) \quad (19)$$

$$f(net) = f \frac{1}{1 + e^{-net}} \quad (20)$$

where,  $f(net)$  is an experimental stimulation purpose in which each hidden neuron at the hidden layers uses both Sigmoid and Softamax.  $v_{pq}^1$  and  $u_{pq}^1$  represent heaviness associates between the first hidden layer  $hd_q^1(s)$  and the input layer  $x_p(t)$ , and between the first hidden layer  $hd_q^1(s)$  and the first context layer  $CT_1^1$ , respectively.

The feed-forward to the second hidden layer can be stated as follows:

$$hd_g^2(s) = f\left(\sum_q^{HD1} v_{qg}^2 hd_q^1(s)\right) + f\left(\sum_m^{Con^2} u_{mg}^2 CT_m^2(s)\right) \quad (21)$$

where,  $v_{qg}^2$  and  $u_{mg}^2$  denote weight associates among the second hidden layer  $hd_g^2(s)$  and the first layer  $hd_q^1$ , respectively, and among the second hidden layer  $hd_g^2$  and the second setting layer  $CT_m^2$ .

The following is one possible formulation for the feed-forward to the output layer:

$$O_k(s) = f\sum_g^{HD2} w_{gk} hd_g^2(s) \quad (22)$$

where,  $w_{gk}$  stands for the weighted link among the second hidden layer  $hd_g^2(s)$  and the output layer  $O_k(s)$ .

Also, this model's training aims to achieve the highest classification with the lowest Mean Square Error (MSE), which measures the difference between the upgraded RNN with GWO's projected output and the intended result. The MSE is determined as follows:

$$MSE_{pt} = \frac{1}{n} \sum_{k=1}^n (O_k(s) - d_k(s))^2 \quad (23)$$

where,  $n$  denotes the amount of production neurons and  $d_k(s)$  and  $O_k(s)$  signify the  $k^{\text{th}}$  neuron's intended and actual outputs. The overall MSE through all examples is given as surveys:

$$Total_{MSE} = \frac{1}{n} \sum_{pt=1}^T MSE_{pt} \quad (24)$$

where,  $pt$  is an example design, and  $T$  is the number of training configurations.

By processing numerous Mayfly Algorithm results from various search space areas through Harmony Search, improved solutions were discovered by combining the Harmony Search and Mayfly Algorithm.

The class of insects known as Palaeoptera includes the order Ephemeroptera, which provides for mayflies. These insects are known as "mayflies" since they mostly appear in May in the UK. Before they are prepared to rise to the superficial as mature mayflies, immature mayflies spend numerous centuries developing as aquatic nymphs. A few meters above the water, most adult males congregate in swarms to appeal to the females. They customarily dance for their wedding, going up and down while establishing a beat. For mating, female mayflies travel to these swarms. After a short period is spent mating, they dump their eggs into the water to resume the cycle [15].

The MA enhancements improve the method's performance for both small and large-scale feature sets. The following list includes the MA components:

**Male Mayfly Movement:** Eq. (25) changes a male mayfly's location as follows:

$$y_m^{t+1} = y_m^t + u_m^{t+1} \quad (25)$$

where,  $y_m^t$  represents the male mayfly's current position, and

$y_m^{t+1}$  represents the position after adding  $u_m^{t+1}$  respectively. The male mayflies have remarkable velocity and always have insufficient patterns overhead the water's surface. Eq. (26) calculates the speed of a male mayfly as follows:

$$u_{kn}^{t+1} = g * u_{kn}^t + b_1 * e^{-\beta r_p^2} (pbest_{kn} - y_{kn}^t) + b_2 * e^{-\beta r_g^2} * (gbest_n - y_{kn}^t) \quad (26)$$

where,  $u_{kn}^t$  is a mayfly's velocity in measurement  $j$  at period  $t$ ,  $y_{kn}^t$  is its position at period  $t$ ,  $b_1$  and  $b_2$  are optimistic attraction coefficients used to measure the contributions of the cognitive and social apparatuses. Correspondingly,  $g$  is a gravitational constant and secure discernibility constant that maximizes a mayfly's discernibility to other mayflies. The finest position that a specific mayfly,  $k$ , has ever visited is represented by  $pbest_k$ , while the  $j^{\text{th}}$  element of the location of the greatest male mayfly is represented by  $gbest_n$ . Since this is a minimization problem, the following changes are made to  $pbest_k$ :

$$pbest_k = \begin{cases} y_k^{t+1} \\ \text{if } fitness(y_k^{t+1}) < fitness(pbest_k) \end{cases} \quad (27)$$

where,  $fitness(y_k^{t+1})$  offers the solution's quality, or the position's fitness value. Last but not least,  $r_p$  is the Cartesian expanse among  $r_g$  and  $pbest_k$  while  $r_g$  is the expanse among  $r_g$  and  $gbest$ . According to the calculations shown in Eq. (28).

$$|y_k - Y_k| = \sqrt{\sum_{j=1}^n (y_{kn} - Y_{kn})^2} \quad (28)$$

where,  $Y_k$  either represents  $pbest_k$  or  $gbest$  and  $y_{kn}$  indicates the location of the  $j^{\text{th}}$  component of the  $k^{\text{th}}$  mayfly. The algorithm's stochastic component depends on the best mayflies at a specific period continuing to complete the nuptial dance. This dance is mathematically represented in Eq. (29).

$$u_{kn}^{t+1} = g * u_{kn}^t + d * r \quad (29)$$

where,  $r$  is an arbitrary amount among  $\in [1, 1]$  and  $d$  is the nuptial dance constant. As  $d_{itr} = d_0 * \delta^{itr}$ , the nuptial dance constant progressively reductions. It is the current iteration count with a random value between  $\in [0, 1]$ , where  $d_0$  is the starting assessment of the nuptial dance constant.

**Female Mayfly Movement:** Female mayflies seek out males and fly near them during mating. The Mayfly is currently located where:

$$x_m^{t+1} = x_m^t + u_m^{t+1} \quad (30)$$

where,  $x_m^t$  represents the female mayfly's present location at period  $t$ , which is updated by its velocity,  $u_m^{t+1}$ . The superiority of the present explanation influences men's and women's attraction to one another; as a result, the best-performing woman is drawn to the best-performing guy, and so on. The equation is used to update a female's velocity (31).



$$u_{kn}^{t+1} = \begin{cases} \text{if } fitness(x_k) > fitness(y_k) \\ g * u_{kn}^t + b_2 * e^{-\beta r_{mf}^2} * (y_{kn}^t - x_{kn}^t) \\ \text{else if } fitness(x_k) \leq fitness(y_k) \\ g * u_{kn}^t + fl * r \end{cases} \quad (31)$$

where,  $u_{kn}^{t+1}$  is the location of the female mayfly  $k$  in measurement  $j$  at period  $t$ ,  $y_{kn}^t$  is the  $j^{\text{th}}$  section of the location of the male mayfly  $k$  at period  $t$ , and  $x_{kn}^t$  is the  $j^{\text{th}}$  constituent of the velocity of the  $k^{\text{th}}$  female mayfly's velocity at period  $t$ .  $r$  is a arbitrary number between  $[1, 1]$  and  $r_{if}$  is the Cartesian expanse among male and female mayflies, which is assumed in Calculation 4.  $a_2$  and are the before distinct attraction continuous and visibility measurement, correspondingly.  $g$  is the gravity coefficient previously distinct in Calculation 2. In the situation where a female is not involved to a man,  $fl$  is an arbitrary walk coefficient, and  $fl_{itr} = fl_0 \times \delta^{itr}$ . Here,  $itr$  are two variables from Calculation that have already been defined.

**Mayfly Crossover:** The crossover operation requires the selection of a male and a female mayfly. The assortment of breeding partners is founded on suitability assessment, with the best male mating with the best female. Equation illustrates that a crossover results in the birth of two offspring (32 and 33).

$$offspring1 = r_{of} * male + (1 - r_{of}) * female \quad (32)$$

$$offspring2 = r_{of} * female + (1 - r_{of}) * male \quad (33)$$

As a result, the male replaces the male mayfly parent, the female returns the female mayfly parent, and  $r_{of}$  is a constant number between 0 and 1. The offspring's initial velocities are established to 0.

**Mayfly Effect:** Newly-born offspring are altered to improve the algorithm's exploratory capabilities. The offspring's variable is supplemented by a regularly distributed random number, as explained in

$$offspring'_n = oggspring_n + k \quad (34)$$

where,  $k$ , the randomly chosen value, has a normal distribution.

### Harmony Search

In the natural world, a specific connection exists between several sound waves with various frequencies. Humans frequently find comfort in musical performances. Its aesthetic evaluation determines the extent of the calming impact.

A musician aspires to the highest aesthetic state comparable to the world's best. As a result, choosing the optimal aesthetic condition is an optimization challenge with a predetermined objective function. The usual resonances produced by tools connected simultaneously estimate aesthetic value. Practice can raise the artistic value of the sound produced. The HS pseudo code is shown in Algorithm 1.

### MHS Algorithm

The MA discussed in paragraphs III-A was initially established for ongoing optimization issues. In Algorithm 2, the redesigned FS algorithm known as MA-HS is described. Each linear combination in MA is first reduced to its binary form or to 0s and 1s before being assessed. This conversion is accomplished by using the S-shaped transmission purpose.

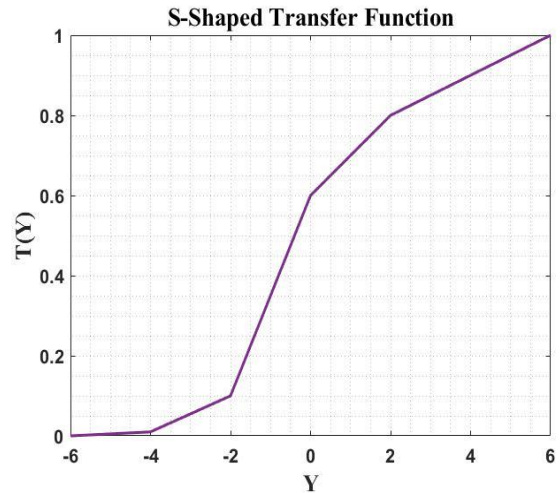
This purpose provides the likelihood of selecting a specific feature in an explanation vector. It is a dependable purpose utilized by numerous investigators in the past. Figure 3 demonstrates the S-shaped transmission meaning employed in this approach.

```

Algorithm 1 Pseudo Code for Harmony Search
Input: MaxHS, NumVariables, raccept, rpa, brange
Output: Best Solution  $Y = (y_1, y_2, \dots, y_d)$ 

Initialize HM
for itr ← 1 to MaxHS do
  for i ← 1 to NumVariables do
    if rand1 < raccept then
      Choose a value from HM
      for the  $i^{\text{th}}$  variable
        if rand2 < rpa then
          Adjust the value by an amount of brange * rand3
        end if
      else
        Choose a random value
      end if
    end for
    Accept the new solution if better
  end for
Return the current best solution

```



**Figure 3.** S-shaped transfer function for binary representation of the continuous Mayfly search space

$$S(y) = \frac{1}{1 + e^{-y}} \quad (35)$$

Eq. (35) updates the agent's feature during the conversion process.

$$P_d^{t+1} = \begin{cases} 1 & \text{if } S(P_d^{t+1}) > rand \\ 0 & \text{if } S(P_d^{t+1}) \leq rand \end{cases} \quad (36)$$

where,  $P_d^{t+1}$  is the agent's updated feature subset, and is an arbitrary amount among 0 and 1, and  $S(P_d^{t+1})$  is the S-shaped transmission purpose as earlier distinct in Eq. (36). Figure 4 depicts the proposed framework's general layout.

**Algorithm 2 Pseudo Code for the MA-HS Algorithm Suggested**

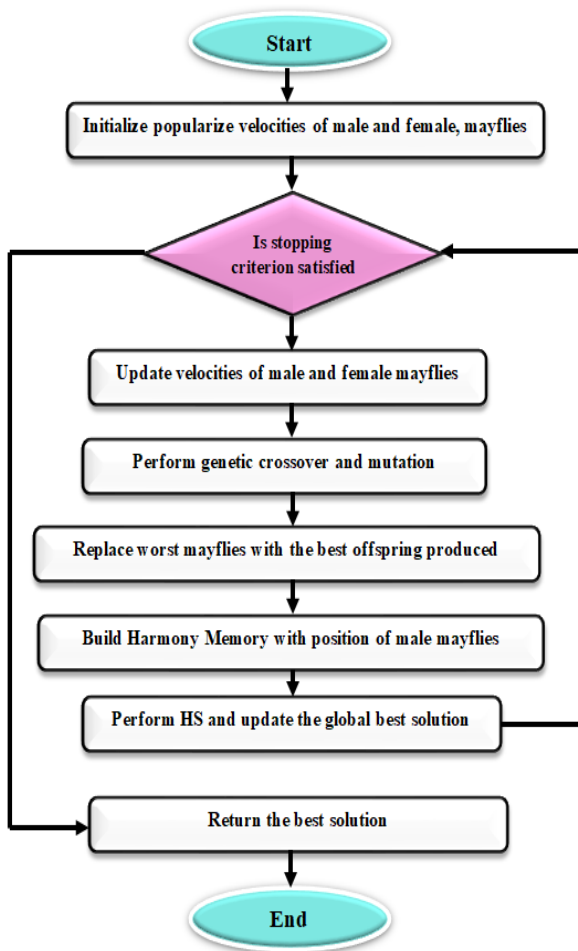
Input: PopSize, MaxIter

Output: Best Agent  $Y = (y_1, y_2, \dots, y_d)$ 

```

initiate population and male and female velocities
sporadic mayflies
Analyze the population, then choose the best
for itr ← 1 to MaxIter do
  for i ← 1 to PopSize do
    Update pbest
    Assess and update the mayfly's male and female velocities.
  end for
  Sort and rate the mayflies.
  cross over and produce both male and female offspring
  Change the progeny
  The best new offspring produced should replace the poorest mayflies.
  Conduct HS on male mayflies.
  Update gbest
end for

```



**Figure 4.** A description of the MHS algorithm, which is used to solve FS problems

### Fitness Function

In this section, the algorithm assesses the effectiveness of an explanation. Since it is a wrapper-based solution, a learning procedure has been employed. The classification accuracy is determined using the Multi-hidden Recurrent Neural Network (MRNN) classifier. The number of features and the classification error make up the fitness function. The authors of FS want to decrease the number of characteristics while improving accuracy simultaneously. Accuracy isn't employed in this case; categorization error is. This is so that the

inaccuracy and the number of topographies can be concentrated. Thus, merging these two will lower the fitness purpose to a single impartial purpose. Figure 4 shows the flowchart of MHS. The formula for calculating the value of a feature subset is found in Eq. (37).

$$\downarrow Fitness = \gamma \times \lambda + (1 - \gamma) \times \frac{|f|}{|F|} \quad (37)$$

where,  $|f|$  is the amount of topographies in the feature subsection,  $|F|$  is the amount of topographies in the supplied dataset,  $\gamma \in [0,1]$  is a limitation that indicates the comparative influence of the organization error on the number of topographies, and  $\lambda$  is the classification error.

## 4. RESULTS AND DISCUSSIONS

The experimental setup for the recommended strategy is covered in this section, besides the evaluation dataset, metrics, and performance analysis.

The DUC (Document Understanding Conference) datasets DUC05, DUC06, and DUC07 are the benchmark datasets in manuscript summarization and comprise a group of news documents besides reference (human-produced) reviews. These datasets are used in the experiments to appraise the projected technique. We have only considered materials in Hindi and English to make the summaries easier to grasp by hand.

To construct the system summary in this study, we qualified our perfect using the training information. To do this, the training dataset was created by choosing 75% of the datasets' data, while the remaining 25% was used as the test data.

We compare our system's overall performance to many extractive baselines and several cutting-edge abstractive baselines, and we also present the 95% confidence intervals. We immediately used the data presented, and the confidence intervals were not reported for the abstractive technique MRNN-MHS. We assessed the efficiency of the suggested strategy using the ROUGE- 1.5.5 toolset. Two ROUGE metrics are used: ROUGE-1 (based on unigrams) and ROUGE-2 (based on bigrams). We have presented the ROUGE-1 and ROUGE-2 outcomes. By comparing the reference summary with the system summary, the ROUGE-1 reflects one word at a period. The ROUGE-2 requires a system summary of two terms in a row to match the reference summary.

### (1) Precision analysis

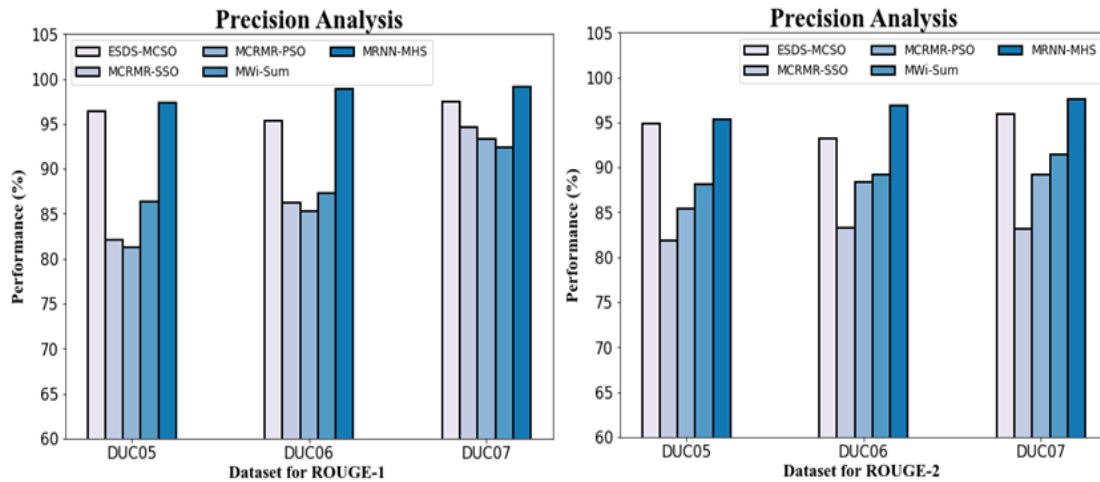
An assessment of the precision of the MRNN-MHS method and other existing methods is shown in Figure 5 and Table 1. The figure validates that the MRNN-MHS has achieved higher performance with precision using the Efficient Text Summarization technique. For instance, the MRNN-MHS model has a precision value of 97.32% for the data set DUC05 on ROUGE-1, compared to accuracy values of 96.39%, 82.05%, 81.21%, and 86.39% for the ESDS-MCSO, MCRMR-SSO, MCRMR-PSO, and MWi-Sum models.

### (2) Recall analysis

A comparison of the MRNN-MHS technique with various known methods is shown in Figure 6 and Table 2. The figure demonstrates enhanced recall performance for the MRNN-

MHS using the Efficient Text Summarization technique. For example, the MRNN-MHS model has a recall value of 94.35% for the data set DUC05 on ROUGE-1, compared to recall

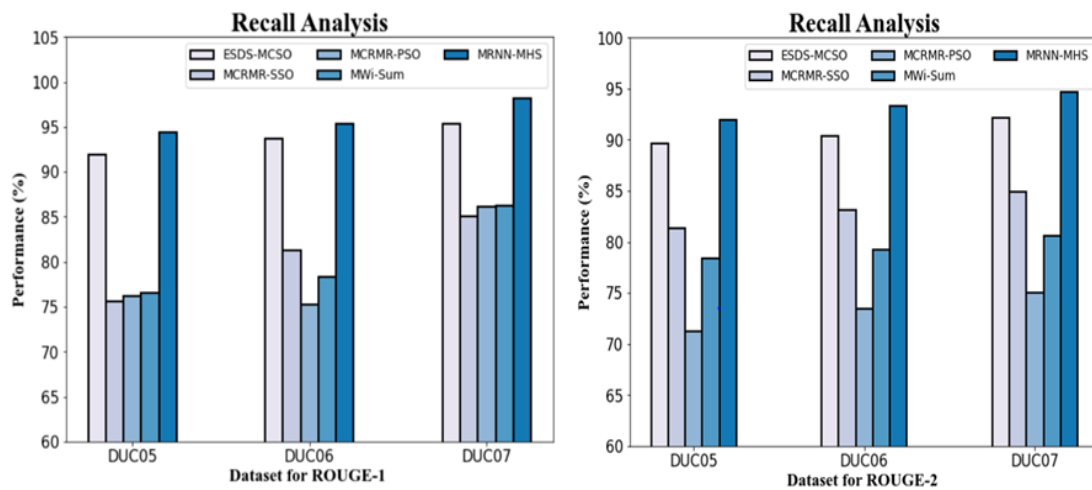
discounts of 91.94%, 75.64%, 76.13%, and 76.57% for the ESDS-MCSO, MCRM-R-SSO, MCRM-R-PSO, and MWi-Sum models.



**Figure 5.** Comparison of MRNN-MHS with current systems for precision analysis

**Table 1.** Comparison of MRNN-MHS with current systems for precision analysis

Dataset	ESDS-MCSO	MCRM-R-SSO	MCRM-R-PSO	MWi-Sum	MRNN-MHS
<b>ROUGE-1</b>					
DUC05	96.39	82.05	81.21	86.39	97.32
DUC06	95.36	86.23	85.28	87.31	98.87
DUC07	97.48	94.61	93.27	92.34	99.16
<b>ROUGE-2</b>					
DUC05	94.89	81.90	85.37	88.10	95.31
DUC06	93.21	83.29	88.36	89.18	96.82
DUC07	95.87	83.21	89.23	91.42	97.56

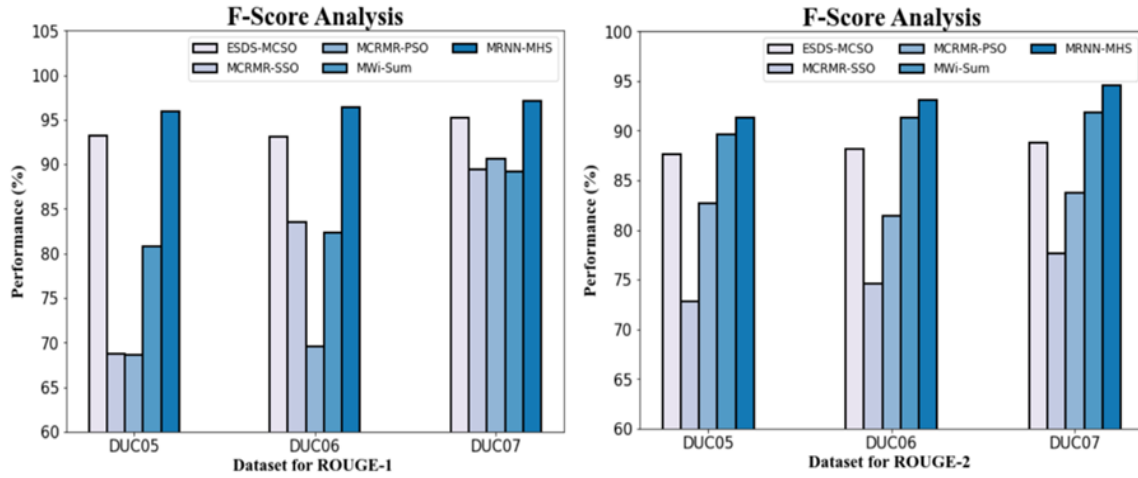


**Figure 6.** Comparison of MRNN-MHS with current systems for recall analysis

**Table 2.** Comparison of MRNN-MHS with current systems for recall analysis

Dataset	ESDS-MCSO	MCRM-R-SSO	MCRM-R-PSO	MWi-Sum	MRNN-MHS
<b>ROUGE-1</b>					
DUC05	91.94	75.64	76.13	76.57	94.35
DUC06	93.68	81.27	75.18	78.29	95.28
DUC07	95.36	85.07	86.16	86.27	98.17
<b>ROUGE-2</b>					
DUC05	89.67	81.34	71.24	78.39	91.89
DUC06	90.32	83.15	73.45	79.21	93.27
DUC07	92.15	84.87	74.98	80.54	94.67

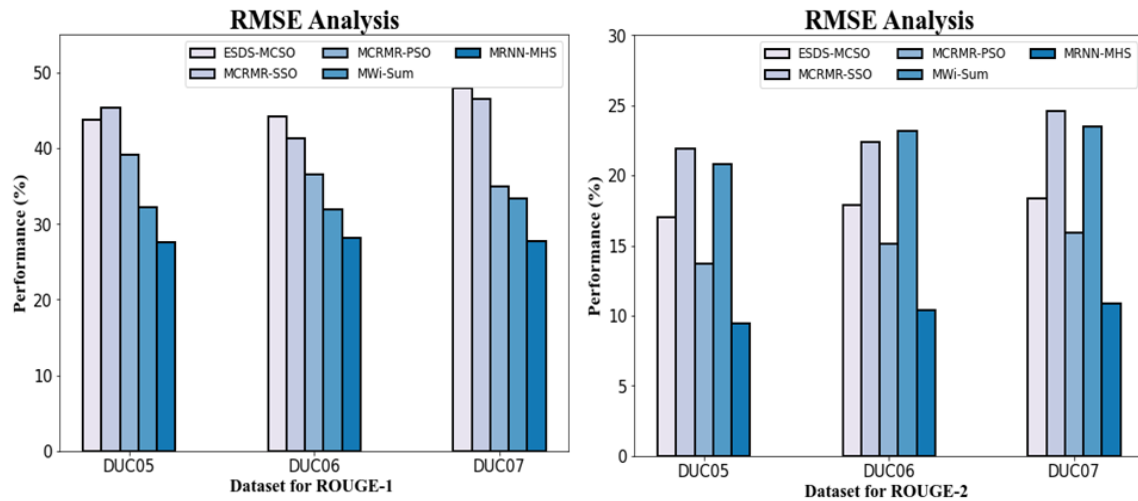




**Figure 7.** Comparison of MRNN-MHS with current systems for F-score analysis

**Table 3.** Comparison of MRNN-MHS with current systems for F-score analysis

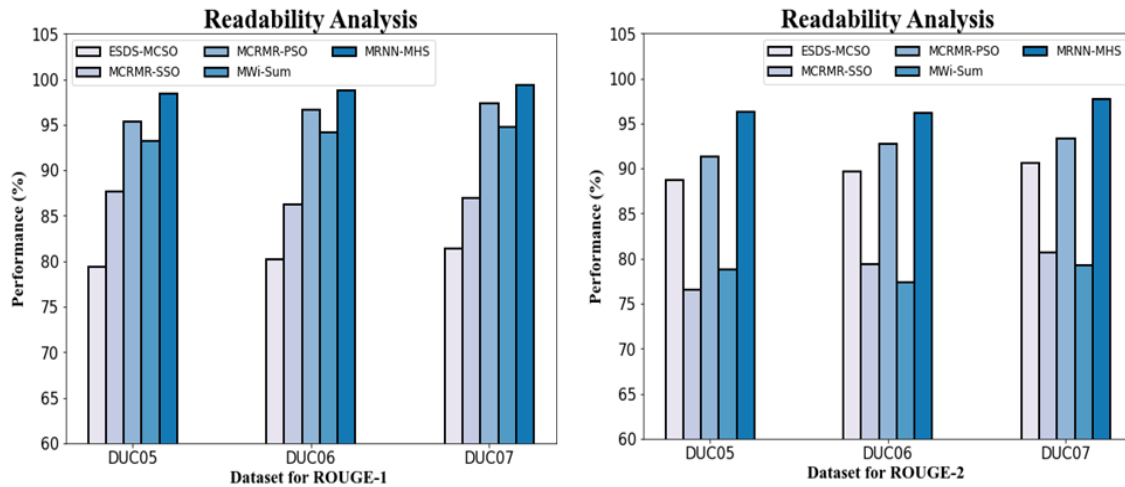
Dataset	ESDS-MCSO	MCRMR-SSO	MCRMR-PSO	MWi-Sum	MRNN-MHS
<b>ROUGE-1</b>					
DUC05	93.24	68.71	68.58	80.82	95.87
DUC06	93.08	83.54	69.54	82.31	96.43
DUC07	95.24	89.37	90.58	89.14	97.13
<b>ROUGE-2</b>					
DUC05	87.65	72.78	82.67	89.67	91.35
DUC06	88.13	74.56	81.39	91.34	93.14
DUC07	88.76	77.68	83.74	91.79	94.57



**Figure 8.** Comparison of MRNN-MHS with current systems for RMSE analysis

**Table 4.** Comparison of MRNN-MHS with current systems for RMSE analysis

Dataset	ESDS-MCSO	MCRMR-SSO	MCRMR-PSO	MWi-Sum	MRNN-MHS
<b>ROUGE-1</b>					
DUC05	43.78	45.31	39.12	32.19	27.54
DUC06	44.23	41.29	36.54	31.89	28.19
DUC07	47.89	46.54	34.89	33.41	27.76
<b>ROUGE-2</b>					
DUC05	16.98	21.89	13.67	20.78	9.43
DUC06	17.90	22.34	15.12	23.18	10.42
DUC07	18.32	24.56	15.89	23.45	10.87



**Figure 9.** Comparison of MRNN-MHS with current systems for readability analysis

**Table 5.** Comparison of MRNN-MHS with current systems for readability analysis

Dataset	ESDS-MCSO	MCRMR-SSO	MCRMR-PSO	MWi-Sum	MRNN-MHS
<b>ROUGE-1</b>					
DUC05	79.32	87.65	95.32	93.21	98.43
DUC06	80.14	86.21	96.67	94.21	98.78
DUC07	81.43	86.89	97.32	94.78	99.32
<b>ROUGE-2</b>					
DUC05	88.73	76.54	91.32	78.76	96.31
DUC06	89.67	79.34	92.76	77.34	96.13
DUC07	90.65	80.73	93.32	79.21	97.67

### (3) F-score analysis

An f-score comparison of the MRNN-MHS technique with other known methods is shown in Figure 7 and Table 3. The figure demonstrates that the MRNN-performance MHSs with the Effective Text Summarization technique have improved in terms of f-score. For example, the MRNN-MHS model's f-score for the data set DUC05 on ROUGE-1 is 95.87%, whereas those for the ESDS-MCSO, MCRMR-SSO, MCRMR-PSO, and MWi-Sum models are 93.24%, 68.71%, 68.58%, and 80.82%, respectively.

### (4) RMSE analysis

Figure 8 and Table 4 describe the RMSE analysis of the MRNN-MHS technique with other existing methods. The data clearly explains that the proposed technique has the least RMSE associated with the different approaches in all characteristics. For the sample, with the DUC05 dataset on ROUGE-1, the proposed method has an RMSE of 27.54%, while it is 43.78%, 45.31%, 39.12%, and 32.19% for ESDS-MCSO, MCRMR-SSO, MCRMR-PSO, and MWi-Sum and for ROUGE-2 the proposed method has an RMSE of 9.43%, while it is 16.98%, 21.89%, 13.67%, and 20.78% for ESDS-MCSO, MCRMR-SSO, MCRMR-PSO, and MWi-Sum respectively.

### (5) Readability analysis

An association of the readability of the MRNN-MHS method and other existing methods is shown in Figure 9 and Table 5. The graphic establishes that the MRNN-Efficient MHS's Text Summarization technique has improved performance while maintaining readability. For example, the readability value for the MRNN-MHS model with the data set

DUC05 on ROUGE-1 is 98.43%. In contrast, the readability values for the ESDS-MCSO, MCRMR-SSO, MCRMR-PSO, and MWi-Sum models are, respectively, 79.32%, 87.65%, 95.32%, and 93.21%.

## 5. CONCLUSION

Based on the multi-hidden Recurrent Neural Network (MRNN) study, the proposed hybrid approach effectively generates accurate and concise summaries from lengthy texts. The hybrid model combines the strengths of the Multi-hidden Layer Recurrent Neural Network (MHL-RNN) and the metaheuristic algorithm to progress the superiority of abstractive manuscript summarization. By taking diverse Mayfly Algorithm explanations from different regions of the search space and processing them with Harmony Search, the suggested hybrid of the Mayfly Algorithm and Harmony Search was employed to produce superior results. The text summary task was described as an optimization problem to ensure that the system-generated summaries are relevant, with numerous text qualities with optimal weights used to score the condemnations to discover the applicable ones. The proposed model outperformed competing approaches such as extractive single document summarization using multi-objective modified cat swarm optimization (ESDS-MCSO), maximum coverage and relevancy with minimal redundancy-shark small optimization (MCRMR-SSO), whole scope, and applicability with minimal redundancy-particle swarm optimization (MCRMR-PSO), and a multilingual summarizer based on frequent weighted item set (MWi-Sum).

## REFERENCES

- [1] Debnath, D., Das, R., Pakray, P. (2021). Extractive single document summarization using multi-objective modified cat swarm optimization approach: ESDS-MCSO. *Neural Computing and Applications*, 1-16. <http://dx.doi.org/10.1007/s00521-021-06337-4>
- [2] Gambhir, M., Gupta, V. (2017). Recent automatic text summarization techniques: A survey. *Artificial Intelligence Review*, 47(1): 1-66. <http://dx.doi.org/10.1007/s10462-016-9475-9>
- [3] Verma, P., Om, H. (2019). MCRM: Maximum coverage and relevancy with minimal redundancy based multi-document summarization. *Expert Systems with Applications*, 120: 43-56. <http://dx.doi.org/10.1016/j.eswa.2018.11.022>
- [4] Rautray, R., Balabantaray, R.C. (2018). An evolutionary framework for multi document summarization using cuckoo search approach: MDSCSA. *Applied Computing and Informatics*, 14(2): 134-144. <http://dx.doi.org/10.1016/j.aci.2017.05.003>
- [5] Peng, L., Zhu, Q., Lv, S.X., Wang, L. (2020). Effective long short-term memory with fruit fly optimization algorithm for time series forecasting. *Soft Computing*, 24: 15059-15079. <http://dx.doi.org/10.1007/s00500-020-04855-2>
- [6] Wang, L., Peng, L., Wang, S., Liu, S. (2020). Advanced backtracking search optimization algorithm for a new joint replenishment problem under trade credit with grouping constraint. *Applied Soft Computing*, 86: 105953. <http://dx.doi.org/10.1016/j.asoc.2019.105953>
- [7] Civicioglu, P., Besdok, E. (2021). Bezier search differential evolution algorithm for numerical function optimization: A comparative study with CRMLSP, MVO, WA, shade and LSHADE. *Expert Systems with Applications*, 165: 113875. <https://doi.org/10.1016/j.eswa.2020.113875>
- [8] Chaudhari, M., Mattukoyya, A.N. (2018). Tone biased MMR text summarization. arXiv:1802.09426. <https://doi.org/10.48550/arXiv.1802.09426>
- [9] Nambiar, S.K., Peter S, D., Idicula, S.M. (2021). Attention based abstractive summarization of Malayalam document. *Procedia Computer Science*, 189: 250-257. <http://dx.doi.org/10.1016/j.procs.2021.05.088>
- [10] Karmakar, R., Nirantar, K., Kurunkar, P., Hiremath, P., Chaudhari, D. (2021). Indian regional language abstractive text summarization using attention-based LSTM neural network. 2021 International Conference on Intelligent Technologies (CONIT), Hubli, India, pp. 1-8. <http://dx.doi.org/10.1109/CONIT51480.2021.9498309>
- [11] Dou, Z.Y., Liu, P., Hayashi, H., Jiang, Z., Neubig, G. (2021). GSum: A general framework for guided neural abstractive summarization. arXiv Preprint arXiv:2010.080142020. <http://dx.doi.org/10.18653/v1/2021.naacl-main.384>
- [12] Liu, Y., Dou, Z.Y., Liu, P. (2021). Refsum: Refactoring neural summarization. arXiv preprint arXiv:2104.07210. <http://dx.doi.org/10.18653/v1/2021.naacl-main.113>
- [13] Verma, P., Om, H. (2019). MCRM: Maximum coverage and relevancy with minimal redundancy based multi-document summarization. *Expert Systems with Applications*, 120: 43-56. <http://dx.doi.org/10.1016/j.eswa.2018.11.022>
- [14] Rashid, T.A., Abbas, D.K., Turel, Y.K. (2019). A multi hidden recurrent neural network with a modified grey wolf optimizer. *PloS One*, 14(3): e0213237. <http://dx.doi.org/10.1371/journal.pone.0213237>
- [15] Bhattacharyya, T., Chatterjee, B., Singh, P.K., Yoon, J.H., Geem, Z.W., Sarkar, R. (2020). Mayfly in harmony: A new hybrid meta-heuristic feature selection algorithm. *IEEE Access*, 8: 195929-195945. <http://dx.doi.org/10.1109/ACCESS.2020.3031718>