Intelligent Automation & Soft Computing

Tech Science Press

DOI: 10.32604/iasc.2023.034718

ARTICLE

Abstractive Arabic Text Summarization Using Hyperparameter Tuned Denoising Deep Neural Network

Ibrahim M. Alwayle[1], Hala J. Alshahrani[2], Saud S. Alotaibi[3], Khaled M. Alalayah[1], Amira Sayed A. Aziz[4], Khadija M. Alaidarous[1], Ibrahim Abdulrab Ahmed[5] and Manar Ahmed Hamza[6,*]

[1]Department of Computer Science, College of Science and Arts, Sharurah, Najran University, Sharurah, Saudi Arabia

[2]Department of Applied Linguistics, College of Languages, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia

[3]Department of Information Systems, College of Computing and Information System, Umm Al-Qura University, Makkah, Saudi Arabia

[4]Department of Digital Media, Faculty of Computers and Information Technology, Future University in Egypt, New Cairo, 11835, Egypt

[5]Computer Department, Applied College, Najran University, Najran, 66462, Saudi Arabia

[6]Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia

*Corresponding Author: Manar Ahmed Hamza. Email: ma.hamza@psau.edu.sa

Received: 25 July 2022 Accepted: 11 November 2022 Published: 05 February 2024

ABSTRACT

Abstractive text summarization is crucial to produce summaries of natural language with basic concepts from large text documents. Despite the achievement of English language-related abstractive text summarization models, the models that support Arabic language text summarization are fewer in number. Recent abstractive Arabic summarization models encounter different issues that need to be resolved. Syntax inconsistency is a crucial issue resulting in the low-accuracy summary. A new technique has achieved remarkable outcomes by adding topic awareness in the text summarization process that guides the module by imitating human awareness. The current research article presents Abstractive Arabic Text Summarization using Hyperparameter Tuned Denoising Deep Neural Network (AATS-HTDDNN) technique. The presented AATS-HTDDNN technique aims to generate summaries of Arabic text. In the presented AATS-HTDDNN technique, the DDNN model is utilized to generate the summary. This study exploits the Chameleon Swarm Optimization (CSO) algorithm to fine-tune the hyperparameters relevant to the DDNN model since it considerably affects the summarization efficiency. This phase shows the novelty of the current study. To validate the enhanced summarization performance of the proposed AATS-HTDDNN model, a comprehensive experimental analysis was conducted. The comparison study outcomes confirmed the better performance of the AATS-HTDDNN model over other approaches.

KEYWORDS

Text summarization; deep learning; denoising deep neural networks; hyperparameter tuning; Arabic language

1 Introduction

The text summarization process can be classified into three types based on the angle of observation. The initial angle is related to the input type in which the summarization process is classified into a single- or multi-document summarization [1]. At the time of document summarization, the input becomes merely a single document, and the text from those documents is summarized. In the case of multi-document summarization, the input can be of a multi-document type, whereas the summary must contain data from every single document [2]. The second angle is related to the context-based text summarization process in which the process is classified into domain-specific, generic and query-driven summaries. Among these, generic summaries utilize only the original documents, while query-driven summaries concentrate on returning a significant amount of data based on the query raised by an end user. Domain-specific summary utilizes certain areas of concern to generate the summaries [3,4]. The final and the most significant angle of the text summarization process depends upon the types of output, which in turn have two kinds, such as abstractive and extractive summarization. In extractive summarization, the summary is generated based on the phrases or sentences in the source document by relying upon linguistic and statistical features. On the other hand, the abstractive summarization process expresses the ideologies from the source document using different words related to the real semantics of the text [5,6]. The abstractive summarization process is generally more complex than the extractive one since the former involves semantic text analysis using advanced Natural Language Processing (NLP) and ML approaches. But abstractive summarization is better in terms of quality since it produces a summary of the text, which is almost identical to a human-written summary and is more meaningful [7].

In recent times, DL approaches improved and are widely applied in significant tasks, namely, text summarization, text translation, Sentiment Analysis (SA) and other sectors. Similarly, one of the significant features of utilizing DNNs is that it takes full advantage of big datasets to improve their outcomes. Novel text summarization techniques are related to the sequence-to-sequence structure of encoder-decoder modes [8,9]. Arabic is one of the ancient languages widely spoken across the globe. In contrast, much information is yet to be revealed since archaeologists are still attempting to discover much about this language. Arabic has an official language status in 26 countries, with over 280 million speakers across the globe [10]. However, text summarization processes face difficulties, as far as the Arabic language is concerned, owing to the difficulties with morphological and syntactic structures. Further, the compression ratio is also observed when summarizing many texts instead of a single document [11]. No standard summary exists for the Arabic language, machine-readable dictionaries, or Arabic benchmark corpora or lexicons [12]. Arabic Text Summarization (ATS) has multiple challenges to deal with for research communities, such as identification of the most useful text segment in the product, summarization of numerous documents; assessment of computer-made summaries without comparing them with the summaries generated by humans [13]; and production of abstractive summaries compared with human-made ones. Across the globe, researchers are still searching for an effective ATS mechanism that can precisely review the major topics of the Arabic language [14].

In this background, the current research article presents Abstractive Arabic Text Summarization using Hyperparameter Tuned Denoising Deep Neural Network (AATS-HTDDNN) technique. The presented AATS-HTDDNN technique aims to generate summaries of Arabic text. In the presented AATS-HTDDNN technique, the DDNN model is utilised to generate summaries. Chameleon Swarm Optimization (CSO) algorithm is exploited in this study, which is also a novel contribution of the current study, to fine-tune the hyperparameters relevant to the DDNN model since it considerably affects its summarization efficiency. To validate the enhanced summarization performance of the proposed AATS-HTDDNN model, a comprehensive experimental analysis was conducted.

2 Related Works

In literature [15], a hybrid, single-document text summarization technique (ASDKGA) was proposed integrating Genetic Algorithm (GA), domain knowledge, and statistical features for the extraction of significant points from Arabic political files. The proposed ASDKGA technique was tested on two corpora such as Essex Arabic Summaries Corpus (EASC) and KALIMAT corpus. Recall-based Understudy for Gisting Evaluation (ROUGE) structure was utilized to compare the ASDKGA-based mechanically-produced summaries with that of the summaries produced by human beings. Qaroush et al. [16] devised an automated, generic, and extractive Arabic single document summarization methodology. This method focused on generating summaries with adequate information. The formulated extractive algorithm evaluated every sentence related to the amalgamation of semantic and statistical attributes, in which a new formulation was utilized considering diversity, sentence importance, and coverage. Moreover, two summarization techniques, supervised ML and score-related, were used to generate the summaries, after which the devised features were utilized. Alami et al. [17] presented a new graph-based Arabic summarization mechanism integrating semantic and statistical analyses. The presented technique used ontology-based hierarchical structures and relationships to produce highly-precise similarity measurements among the terms to improve the summary's quality. The presented approach relied upon a 2-dimensional graph method in which semantic and statistical similarities were deployed.

In the study conducted earlier [18], the authors attempted to overcome the existing limitations through an innovative technique utilizing unsupervised NNs, document clustering, and topic modelling to frame a potential file representation method. At first, a novel document clustering approach, using Extreme Learning Machine, was executed on a large text corpus. Secondly, topic modelling was implemented for document collection so as to identify the topics in every cluster. Thirdly, a matrix gave every document a topic space in which the columns represented cluster topics and rows denoted document sentences. In literature [19], the authors tried to overcome such limitations by implementing linear algebraic and statistic techniques integrated with the text's semantic and syntactic processing. Then, the parts of the speech tagger were utilized to reduce LSA dimensions. Then, the term weight in four adjacent sentences was included in the weighting methods, whereas the input matrix was computed to consider syntactic relations and word orders.

Alami et al. [20] introduced an innovative technique for ATS with the help of the Variational Auto-Encoder (VAE) method to study a feature space using a higher-dimension input dataset. The authors explored input representations like Term Frequency (TF) and TF-IDF along with global and local glossaries. Every sentence was ranked under the hidden presentation generated by VAE. In literature [21], the authors accepted a pre-processing approach to resolve the noise issues and utilized the word2vec approach for two purposes; to map the words with a fixed-length vector and to obtain the semantic relations among every vector related to the dimension. Likewise, the author employed a K-means technique for two purposes: selecting distinctive files and tokenising the document into sentences. In this study, additional iterations of the k-means technique were employed to select the key sentences related to the similarity metric to overcome redundancy and produce the summary early.

3 The Proposed Model

The current study developed the projected AATS-HTDDNN technique for Arabic text summarization. The presented AATS-HTDDNN technique aims to generate summaries of Arabic text. In the presented AATS-HTDDNN technique, the DDNN model generates the summaries. Since the hyperparameters relevant to the DDNN model considerably affect the efficiency of the summarization process, the CSO algorithm is exploited. Fig. 1 showcases the block diagram of the proposed AATS-HTDDNN approach.

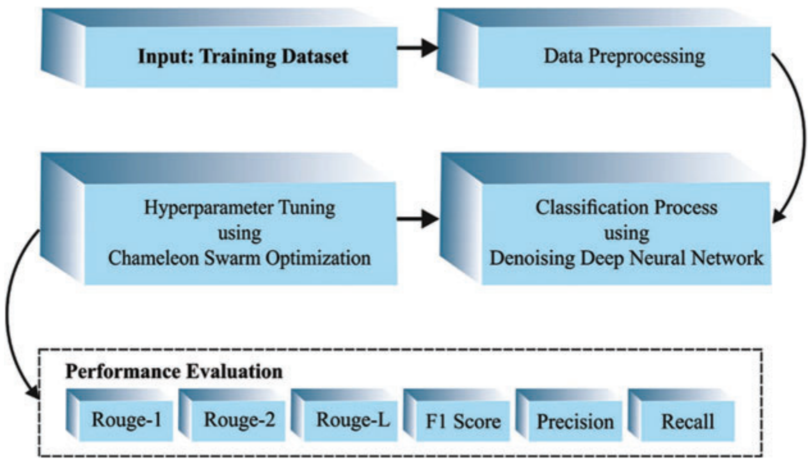

Figure 1: Block diagram of the AATS-HTDDNN approach

3.1 Text Summarization Using DDNN Model

The DDNN model is utilized in this study for the text summarization process. DDNN is created by integrating DAE and RBM and efficiently reduces the noise without removing the features. For the DDNN technique, a vector with set dimensions is considered the input value. Primarily, the technique is applied by denoising element, which has two layers named DAE1 and DAE2, with the help of unsupervised training approaches [22]. At this point, the other instance is trained all the time. Training is generally provided to minimize the reconstruction error in input data result from the preceding layer. Since the encoded or potential expression can be computed based on the preceding layer k, the $(k + 1)^{th}$ layer is directly managed using the results of the k^{th} layer. Still, every denoising layer is trained. After being managed with the Denoising layer, the data enters the RBM phase, which extracts the distinct features from the DAE layer. The following feature extraction is highly important and represents the model.

This phase is developed by stacking two RBM layers. The model is trained by training the RBM from low to high values as given below:

(1) An input of the bottom RBM denotes the resultant of the Denoising layer
(2) The features extracted from the bottom RBM are input values for the topmost RBM.

The rationale behind selecting DAE during the text classification procedure is that the data certainly exists in the form of a blend of distinct kinds. In contrast, the intensity of the noise affects the trained model and deteriorates the final classification performance. DAE performs the initial extraction of new features, whereas the learning condition is noise reduction. During pre-training, various unique strengths and distinct noise signals are added to the new input signals so that the encoded model attains the optimum stability and robustness.

DDNN technique comprises four layers such as RBM1, RBM2, DAE1, and DAE2. Here, layer v acts as either a visual or input layer for the DDNN technique. All the documents in this work are demonstrated by a set of dimensional vectors, whereas W_3, W_1, W_2, and W_4 correspond to the connection weights amongst the layers. Further, h_1, h_2, h_3, and h_4 signify the hidden layers that are equivalent to the resultant layers correspondingly. The DAE2 layer defines the resultant layer of the denoising element along with the input layer of the 2-layer RBM modules. RBM2 represents the resultant layer of the DDNN technique that defines the feature of the documents and is related to the visual layer, v. This layer is a high-level feature depiction of the text dataset. The following text classification task is considered based on this vector. For every node, there is no link between similar layer nodes, whereas the nodes that exist between the two layers of an individual are Fully Connected (FC).

Especially the overview of the energy model is to capture the relationship amongst the variables and augment the model parameters too. So, it is essential to embed the optimum solution problems, such as the energy function, if the model parameters are to be trained. At this point, the RBM energy function is determined as follows:

$$E(\nu,\ h) = -\sum\nolimits_{i=1}^{n}\sum\nolimits_{j=1}^{m} w_{ij}h_i\nu_j - \sum\nolimits_{j=1}^{m} b_j\nu_j - \sum\nolimits_{i=1}^{n} c_ih_i. \tag{1}$$

Here, Eq. (1) implies the energy functions of all the visible and hidden nodes connected to the infrastructure. Particularly, n defines the count of hidden nodes, m represents the number of visible layer nodes, and b and c signify the bias values of visual and hidden layers correspondingly. The main function of the RBM technique is to accumulate the energy of every visible and hidden node. So, all the samples need to count the value of each hidden node equivalent to it so that the total energy is computed. The computation is difficult, whereas the joint likelihood of hidden and visible nodes is given below:

$$P(\nu, h) = \frac{e^{-E(\nu,h)}}{\sum\nolimits_{\nu,h} e^{-E(\nu,h)}}. \tag{2}$$

By establishing these probabilities, the energy function is simplified. In other words, the solution's objective is to mitigate the energy value. A statistical learning model exists in which the state of minimal energy has a superior probability to higher energy. Therefore, as shown below, it can maximize the probability and establish the free energy function.

$$Free\ Energy\ \ (\nu) = -\ \ln \sum\nolimits_{h} e^{-E(\nu,h)}. \tag{3}$$

So,

$$P(\nu) = \frac{e^{Free\ Energy(\nu)}}{Z},\ Z = \sum\nolimits_{\nu,h} e^{-E(\nu,h)}, \tag{4}$$

whereas Z demonstrates the normalization factor. Afterwards, the joint probability $P(\nu)$ is changed as given below:

$$ln\ p(\nu) = -Free\ Energy(\nu) -\ \ ln\ Z. \tag{5}$$

The 1st term on the right side of Eq. (5) represents the negative value of the free energy function for the entire network, whereas the left entity denotes the probability function. The model parameter is resolved by utilizing the maximal possibility function estimate.

At this point, it can act as a primary function to create a denoising function element for the novel feature and is mostly collected from DAE. The 2-layer DAE is located at the bottom of the models to ensure complete utilization of the denoising characters. Here, an input signal is denoised by restructuring the input signals with unsupervised learning. Afterwards, the influence of the noise upon succeeding construction of the classification dataset gets decreased.

The secondary element is established by employing DBN, created using RBM. Then, feature extraction capability is increased in this method. In addition, this method also attains difficult principles from the data, whereas high-level feature extraction is representative. To achieve optimum sorting outcomes, the extraction representative feature is utilized as input to the final classification. Then, the extraction process is repeated again with the help of RBM. Assuming the complexity of the training process and the efficacy of models, a 2-layer DAE and a 2-layer RBM are utilized.

3.2 Parameter Tuning Using CSO Algorithm

In this study, the CSO algorithm is used to fine-tune the hyperparameters involved in the DDNN model. This algorithm considerably enhances the efficiency of the summarization process. The proposed CSA is a new metaheuristic algorithm simulated from the hunting behaviour and food searching method. Chameleons are much-specified classes of species which are capable of changing their body colour according to the environment so that it quickly blends with their environment [23]. Chameleons can live and survive in semi-desert areas, lowlands, mountains, and deserts, and usually prey upon insects. Its food hunting procedure includes the following phases namely, attacking the prey, tracking the prey, and pursuing the prey using their sight. The mathematical steps and models are described in the succeeding subsections. Fig. 2 illustrates the steps involved in the CSO technique.

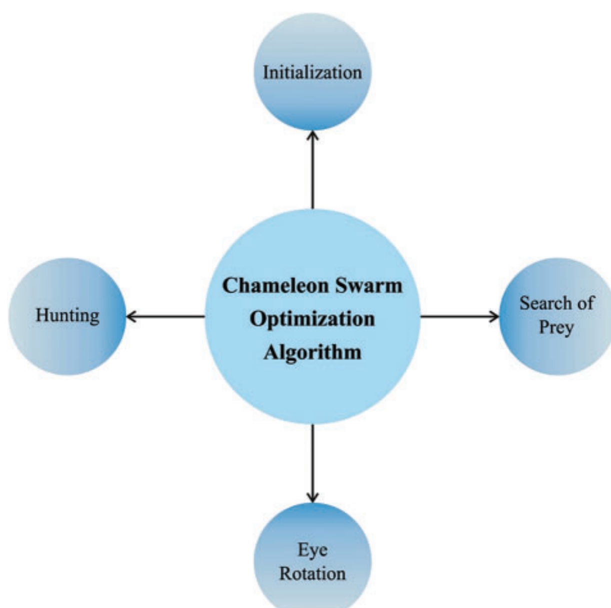

Figure 2: Steps involved in CSO technique

Initialization and Function Evaluation

CSA refers to a population-related metaheuristic algorithm that arbitrarily produces an initial population for optimization. The population of size n is produced in the d dimension search region, in which every individual in the population is a feasible solution to the optimization issue. The position of the chameleon, during iterations, in the searching region is calculated as follows:

$$y_t^i = \left[y_{t,1}^i, y_{t\,2}^{\,i}, y_{t,d}^i\right] \tag{6}$$

Here, $i = 1, 2 \ldots, t$ characterizes the iteration count and $y^i_{\tau,Error::0x0000}$ characterizes the location of the chameleon.

The initial population is produced according to the problem dimension, whereas the chameleon count in the searching region is determined as follows:

$$y^i = l_j + r\left(u_j - l_j\right) \tag{7}$$

In Eq. (7), y^i represents the primary vector of *i-th* chameleon, u_j and l_j denote the upper and lower bounds of the searching region, correspondingly, and r shows a uniform distribution value within [0, 1]. During every step, the quality of the solution is determined for every novel location, according to the assessment of objective function.

Search of Prey

The movement behaviour, at the time of searching, is represented by the updated approach of the location, as follows:

$$y_{t+1}^{i+j} = \begin{cases} y_t^{i,j} + P_1\left(P_\tau^{i,j} - G_\tau^j\right) r_2 + P_2\left(G_\tau^j - y_\tau^{i,j}\right) r_1 \\ \\ y_t^{i,j} + \mu\left(u^j - l^j\right) r_3 + l_b^j sn\left(rand - 0.5\right) r_1 < P_p^{r_1 \geq P_p} \end{cases} \tag{8}$$

In Eq. (8), t and $(t+1)$ designate the t^{th} and $(t+1)^{th}$ iterative phases correspondingly. i and j characterize the i^{th} chameleon in j^{th} parameter. $y_t^{i,j}$ and y_{t+1}^{i+j} denote the existing and novel locations correspondingly. $P_t^{i,j}$ and G_t^j suggest the finest and the global finest locations, correspondingly.

Further, P_1 and P_2 refer to positive numbers that control the exploration capability. r_1, r_2, and r_3 indicate the uniformly-distributed arbitrary values in the range of [0, 1]. r_i denotes the uniform distribution value, generated at index i in the range of [0, 1]. P_p shows the likelihood of chameleons perceiving prey. Sgn (rand-0.5) denotes the effect on the direction of exploration and exploitation phases in the range of [−1, 1]. μ indicates a function of the iteration variable, which decreases with iteration count.

Chameleon's Eyes Rotation

Chameleons can recognize the location of the prey by rotating their eyes up to 360 degrees. This feature assists in spotting the target for which the steps are given below:

- The initial location is the focal point of gravity;
- The rotation matrix identifies the prey's situation;
- Type equation here. The location is restored through a rotation matrix at the focal point of gravity;
- Type equation here. At last, they are resumed to the initial location position.

Hunting Prey

Chameleon assaults its target, once it is extremely closer to the target. The chameleon adjacent to the target is the optimum chameleon which is regarded as the optimum outcome. Such chameleons assault the target through its tongue. The chameleon situation gets enhanced since it can prolong its tongue up to double its length. It assists the chameleon in exploiting the pursuit space and enables it to sufficiently catch the target. The speed of the tongue, once it is protracted toward the target, is arithmetically given herewith.

$$v_{t+1}^{i,j} = wv_t^{i,j} + c_1\left(G_t^j - y_t^{i,j}\right) + c_2\left(P_t^{i,j} - y_t^{i,j}\right)r_2 \tag{9}$$

In Eq. (9), $v_{t+1}^{i,j}$ denotes the novel velocity of i^{th} chameleon in j^{th} parameter during $t+1$ iteration, and $v_t^{i,j}$ shows the existing velocity of i^{th} chameleon in *the* j^{th} parameter.

4 Experimental Validation

The current section provides information on the performance validation of the proposed model under different measures. Table 1 offers the detailed summarization outcomes accomplished by the proposed AATS-HTDDNN model and other existing models in terms of ROUGE.

Table 1: Results of the analysis of the AATS-HTDDNN approach under distinct measures

Methods	ROUGE (%)		
	R1	R2	RL
BERT-fine tuning	12.93	9.44	13.24
Bi-LSTM	30.31	11.70	20.70
GRU model	55.77	13.91	38.92
Seq2seq model	62.13	34.46	44.23
Multilayer bi-seq2seq	60.79	41.28	50.08
D-RNN model	71.60	58.60	70.10
AATS-HTDDNN	78.35	65.98	77.31

Fig. 3 demonstrates the comparative R1 analysis results achieved by the proposed AATS-HTDDNN model and other existing models in terms of R1. The results imply that BERT-Fine Tuning and BiLSTM models reported the least R1 values such as 12.93% and 30.31%, respectively. At the same time, GRU, Seq2seq, and multilayer bi-seq2seq models achieved slightly higher R1 values, such as 55.77%, 62.13%, and 60.79%, respectively. Meanwhile, D-RNN model reached a reasonable R1 value of 71.60%. But, the proposed AATS-HTDDNN model achieved enhanced performance with a maximum R1 value of 78.35%.

Fig. 4 showcases the comparative R2 study outcomes of the proposed AATS-HTDDNN algorithm and other existing models in terms of R2. The results denote that BERT-Fine Tuning and BiLSTM approaches reported the least R2 values, such as 9.44% and 11.70% correspondingly. Meanwhile, GRU, Seq2seq, and multilayer bi-seq2seq approach achieved slightly higher R2 values such as 13.91%, 34.46%, and 41.28%, correspondingly. In the meantime, D-RNN approach attained a reasonable R2 value of 58.60%. But, the proposed AATS-HTDDNN technique exhibited enhanced performance with a maximum R2 value of 65.98%.

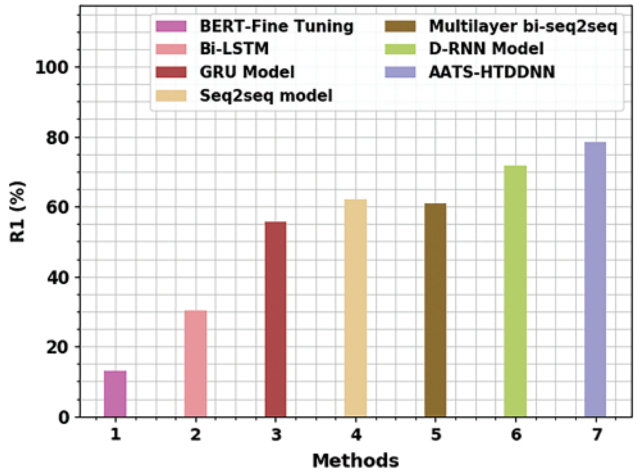

Figure 3: R1 analysis results of AATS-HTDDNN approach and other existing algorithms

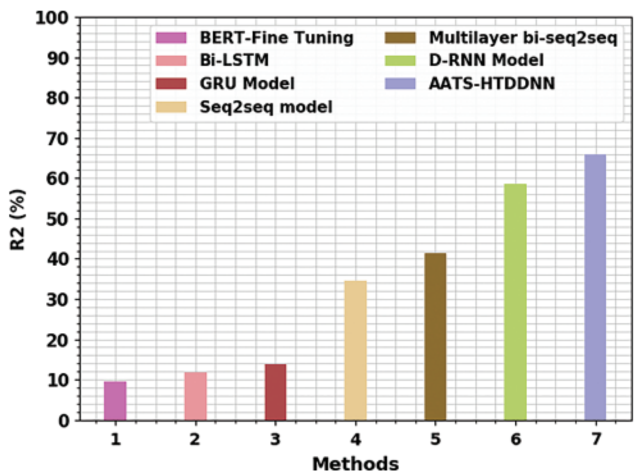

Figure 4: R2 analysis results of AATS-HTDDNN approach and other existing algorithms

Fig. 5 illustrates the detailed RL analysis results achieved by the proposed AATS-HTDDNN approach and other existing models in terms of RL. The results infer that BERT-Fine Tuning and BiLSTM techniques reported the least RL values, such as 13.24% and 20.70% correspondingly. In parallel, GRU, Seq2seq, and multilayer bi-seq2seq techniques displayed slightly increased RL values, such as 38.92%, 44.23%, and 50.08% correspondingly. Temporarily, D-RNN approach gained a reasonable RL value of 70.10%. But, the proposed AATS-HTDDNN technique achieved an enhanced performance with a maximal RL value of 77.31%.

Table 2 shows the detailed summarization outcomes achieved by the proposed AATS-HTDDNN and other existing models on 27-topic labelled dataset. Fig. 6 portrays the comparative $F1_{score}$ analysis results of the AATS-HTDDNN approach and other existing methodologies in terms of $F1_{score}$. The results imply that NB approaches and LR techniques achieved lower $F1_{score}$ values, such as 63.47 and 68.76, correspondingly. Simultaneously, SVM, SGD, and D-RNN approaches exhibited slightly increased $F1_{score}$ values such as 69.40, 69.65, and 83.52 correspondingly. But, the proposed AATS-HTDDNN approach accomplished the best performance with a maximum $F1_{score}$ of 89.87.

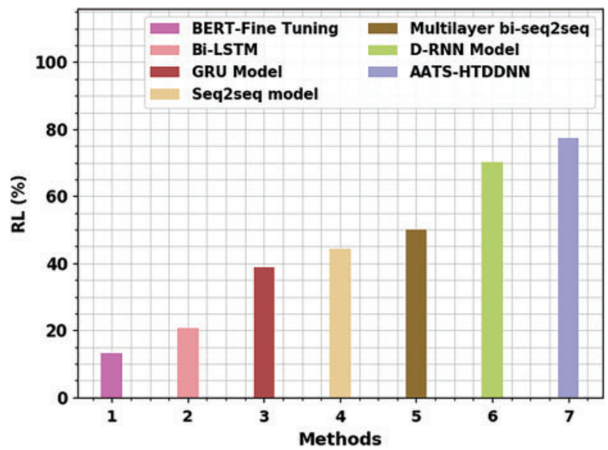

Figure 5: RL analysis results of AATS-HTDDNN approach and other existing algorithms

Table 2: Results of the analysis of AATS-HTDDNN approach and other existing algorithms on 27-topic labelled dataset

27-topic labelled dataset			
Methods	F1 score	Recall	Precision
NB model	63.47	65.76	64.36
SVM	69.40	70.04	71.76
LR	68.76	69.37	69.71
SGD	69.65	70.22	70.73
D-RNN	83.52	84.44	84.49
AATS-HTDDNN	89.87	89.04	90.29

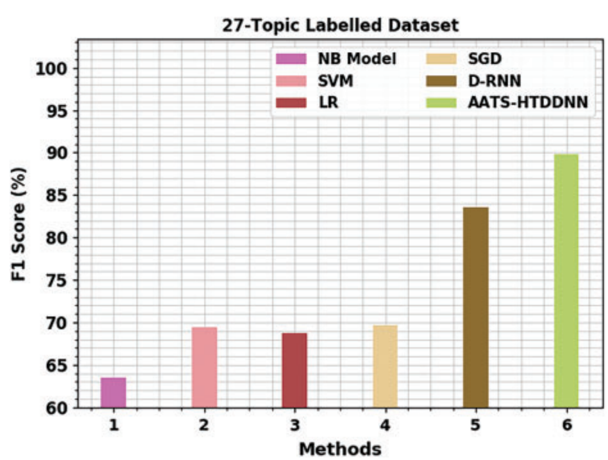

Figure 6: $F1_{score}$ analysis results of AATS-HTDDNN approach on 27-topic labelled dataset

Fig. 7 portrays the brief $reca_l$ investigation outcomes, achieved by the proposed AATS-HTDDNN algorithm and other existing models in terms of $reca_l$. The results imply that NB approaches and LR methodologies produced the least $reca_l$ values such as 65.76 and 69.37 correspondingly. In parallel, SVM, SGD, and D-RNN techniques achieved slightly increased $reca_l$ values such as 70.04, 70.22, and 84.44, correspondingly. But AATS-HTDDNN method exhibited an increased performance with a maximum $reca_l$ value of 89.04.

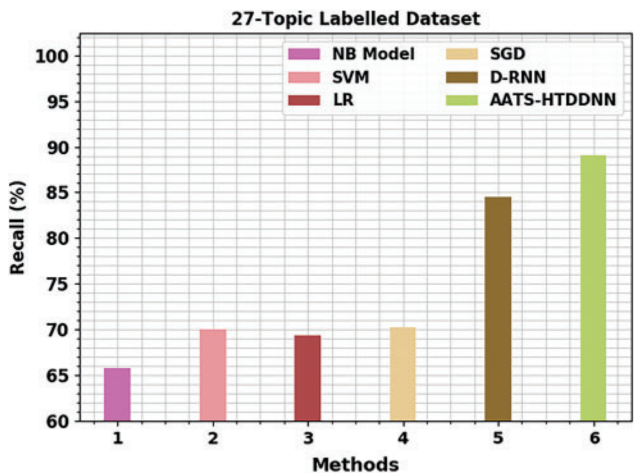

Figure 7: *Reca*$_l$ analysis results of AATS-HTDDNN approach on 27-topic labelled dataset

Fig. 8 validates the comprehensive $prec_n$ analysis results, attained by the proposed AATS-HTDDNN technique and other existing algorithms in terms of $prec_n$. The results denote that NB models and LR approaches reported the least $prec_n$ values, such as 64.36 and 69.71 correspondingly. Meanwhile, SVM, SGD, and D-RNN techniques displayed slightly increased $prec_n$ values such as 71.76, 70.73, and 84.49 correspondingly. But, the proposed AATS-HTDDNN approach achieved an enhanced performance with a maximum $prec_n$ of 90.29.

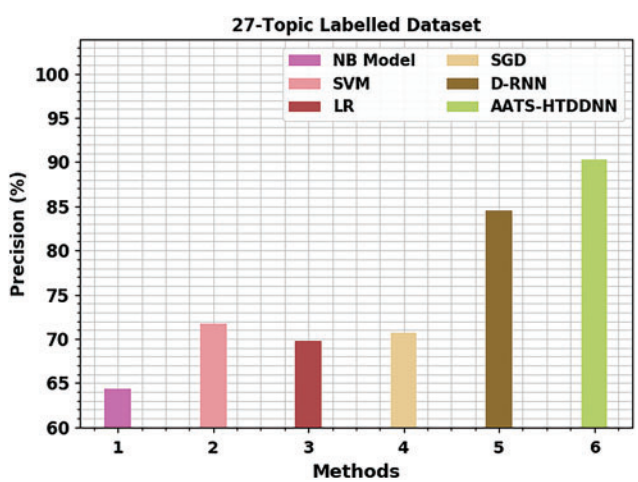

Figure 8: *Prec*$_n$ analysis results of AATS-HTDDNN approach on 27-topic labelled dataset

Both Training Loss (TRL) and Validation Loss (VLL) values, attained by the proposed AATS-HTDDNN algorithm on 27-topic labelled dataset, are shown in Fig. 9. The experimental outcomes imply that the AATS-HTDDNN methodology established the least TRL and VLL values while VLL values were lesser than TRL.

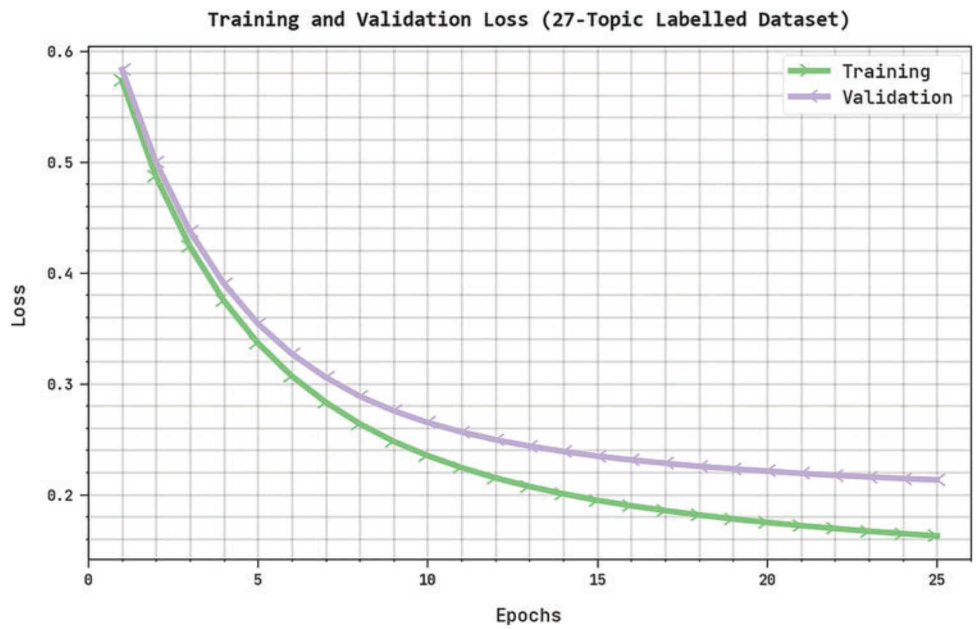

Figure 9: TRL and VLL analyses results of AATS-HTDDNN approach on 27-topic labelled dataset

Table 3 shows the thorough summarization outcomes, yielded by the proposed AATS-HTDDNN and other existing techniques in terms of ROUGE. Fig. 10 shows the brief $F1_{score}$ analysis results achieved by AATS-HTDDNN technique and other existing approaches in terms of $F1_{score}$. The results imply that SGD methods and NB algorithms reported the least $F1_{score}$ values such as 76.62 and 76.81 correspondingly. At the same time, LR, SVM, and D-RNN techniques achieved slightly increased $F1_{score}$ values, such as 76.81, 77.73, and 85.02, correspondingly. But, the proposed AATS-HTDDNN method accomplished an enhanced performance with a maximum $F1_{score}$ value of 88.32.

Table 3: Results of the analysis of AATS-HTDDNN approach under distinct measures on 14-topic labelled dataset

14-topic labelled dataset			
Methods	F1 score	Recall	Precision
NB model	76.81	77.05	77.41
SVM	77.73	77.89	78.60
LR	76.81	77.05	77.41
SGD	76.62	76.81	77.50
D-RNN	85.02	87.13	87.72
AATS-HTDDNN	88.32	91.47	90.66

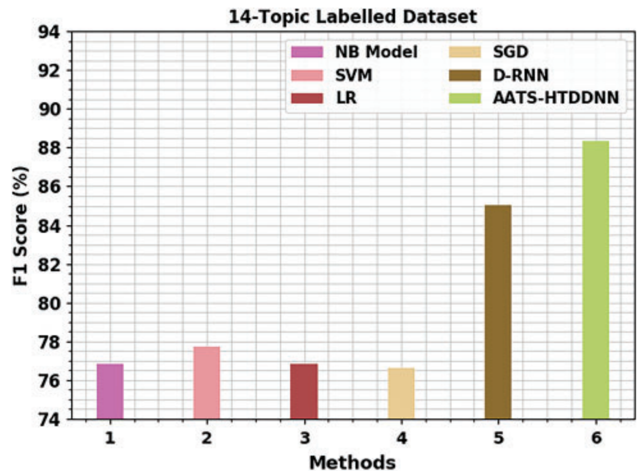

Figure 10: $F1_{score}$ analysis results of AATS-HTDDNN approach on 14-topic labelled dataset

Fig. 11 portrays the comparative $reca_l$ analysis results achieved by the proposed AATS-HTDDNN approach with other 14-topic labelled dataset in terms of $reca_l$. The results showcase that SGD methods and NB methods reported the least $reca_l$ values such as 76.81 and 77.05, correspondingly. Meanwhile, LR, SVM, and D-RNN algorithms achieved slightly higher $reca_l$ values such as 77.05, 77.89, and 87.13 correspondingly. But, the proposed AATS-HTDDNN approach achieved an enhanced performance with a maximum $reca_l$ of 91.47.

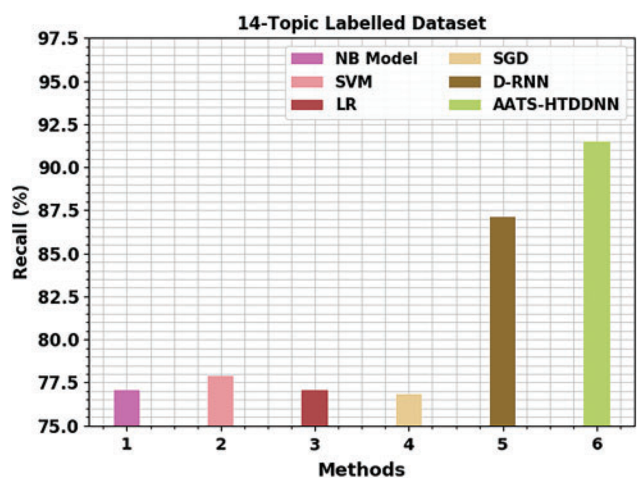

Figure 11: $Reca_l$ analysis results of AATS-HTDDNN approach on 14-topic labelled dataset

Fig. 12 demonstrates the comparative $prec_n$ analysis results, produced by the proposed AATS-HTDDNN algorithm with other 14-topic labelled dataset in terms of $prec_n$. The results denote that SGD methods and NB methods reported the least $prec_n$ values such as 77.50 and 77.41, correspondingly. Simultaneously, LR, SVM, and D-RNN methods achieved slightly higher $prec_n$ values such as 77.41, 78.60, and 87.72, correspondingly. But, the proposed AATS-HTDDNN method accomplished an enhanced performance with a maximum $prec_n$ of 90.66.

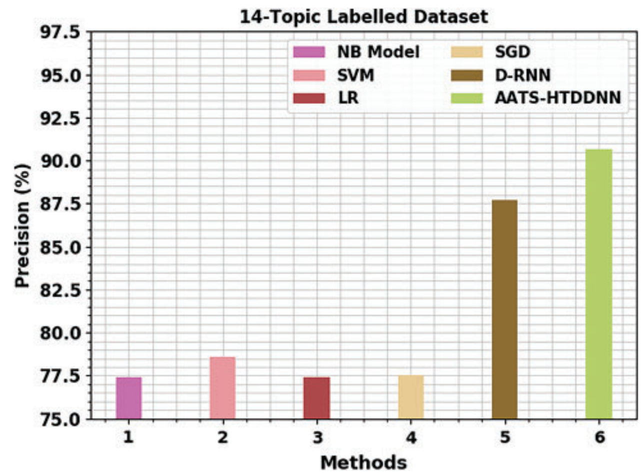

Figure 12: $Prec_n$ analysis results of AATS-HTDDNN approach on 14-topic labelled dataset

Both TRL and VLL values, attained by the proposed AATS-HTDDNN technique on 14-topic labelled dataset, are shown in Fig. 13. The experimental outcomes denote that AATS-HTDDNN method established the least TRL and VLL values while VLL values were lesser than TRL.

Figure 13: TRL and VLL analyses results of AATS-HTDDNN approach on 14-topic labelled dataset

The results and the discussion made above confirmed the enhanced performance of AATS-HTDDNN model over other existing models.

5 Conclusion

In the current study, the proposed AATS-HTDDNN technique has been developed for Arabic text summarization. The aim of the presented AATS-HTDDNN technique is to generate summaries of Arabic text. In the presented AATS-HTDDNN technique, the DDNN model is utilized for the generation of summaries. Since the hyperparameters, relevant to the DDNN model, considerably affect the efficiency of the summarization process, the CSO algorithm is exploited. To validate the enhanced summarization performance of the proposed AATS-HTDDNN model, a comprehensive experimental analysis was conducted. The comparison study outcomes established the superior performance of the AATS-HTDDNN model over other approaches. In the future, the performance of the AATS-HTDDNN algorithm can be boosted with the incorporation of hybrid metaheuristics-based hyperparameter tuning algorithms.

Acknowledgement: The authors express their gratitude to Princess Nourah bint Abdulrahman University researchers supporting the project.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R281), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: 22UQU4210118DSR33. The authors are thankful to the Deanship of Scientific Research at Najran University for funding this work under the Research Groups Funding Program Grant Code (NU/RG/SERC/11/7).

Author Contributions: Conceptualization, Hala J. Alshahrani; Methodology, Hala J. Alshahrani; Ibrahim M. Alwayle, Saud S. Alotaibi. Software, Khaled M. Alalayah; Validation, Amira Sayed A. Aziz; Investigation, Khadija M. Alaidarous; Data curation, Ibrahim Abdulrab Ahmed; Writing–original draft, Ibrahim M. Alwayle, Hala J. Alshahrani, Saud S. Alotaibi, Khaled M. Alalayah, Manar Ahmed Hamza; Writing–review & editing, Manar Ahmed Hamza, Khadija M. Alaidarous, Ibrahim Abdulrab Ahmed; Visualization, Amira Sayed A. Azi; Project administration, Manar Ahmed Hamza; Funding acquisition, Hala J. Alshahrani. All authors have read and agreed to the published version of the manuscript.

Availability of Data and Materials: Data sharing not applicable to this article as no datasets were generated during the current study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. Al-Maleh and S. Desouki, "Correction to: Arabic text summarization using deep learning approach," *Journal of Big Data*, vol. 8, no. 1, pp. 56, 2021.

[2] B. Elayeb, A. Chouigui, M. Bounhas and O. B. Khiroun, "Automatic arabic text summarization using analogical proportions," *Cognitive Computation*, vol. 12, no. 5, pp. 1043–1069, 2020.

[3] R. Elbarougy, G. Behery and A. El Khatib, "Extractive arabic text summarization using modified pagerank algorithm," *Egyptian Informatics Journal*, vol. 21, no. 2, pp. 73–81, 2020.

[4] F. N. Al-Wesabi, "A smart English text zero-watermarking approach based on third-level order and word mechanism of markov model," *Computers, Materials & Continua*, vol. 65, no. 2, pp. 1137–1156, 2020.

[5] A. H. Ababneh, "Investigating the relevance of Arabic text classification datasets based on supervised learning," *Journal of Electronic Science and Technology*, vol. 20, no. 2, pp. 100160, 2022.

[6] F. N. Al-Wesabi, "Proposing high-smart approach for content authentication and tampering detection of arabic text transmitted via internet," *IEICE Transactions on Information and Systems*, vol. E103.D, no. 10, pp. 2104–2112, 2020.

[7] A. Chouigui, O. B. Khiroun and B. Elayeb, "An Arabic multi-source news corpus: Experimenting on single-document extractive summarization," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3925–3938, 2021.

[8] F. N. Al-Wesabi, "A hybrid intelligent approach for content authentication and tampering detection of Arabic text transmitted via internet," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 195–211, 2021.

[9] S. L. M. Sainte and N. Alalyani, "Firefly algorithm based feature selection for Arabic text classification," *Journal of King Saud University–Computer and Information Sciences*, vol. 32, no. 3, pp. 320–328, 2020.

[10] Q. A. Al-Radaideh and M. A. Al-Abrat, "An Arabic text categorization approach using term weighting and multiple reducts," *Soft Computing*, vol. 23, no. 14, pp. 5849–5863, 2019.

[11] F. N. Al-Wesabi, "Entropy-based watermarking approach for sensitive tamper detection of Arabic text," *Computers, Materials & Continua*, vol. 67, no. 3, pp. 3635–3648, 2021.

[12] K. A. Maria, K. M. Jaber and M. N. Ibrahim, "A new model for Arabic multi-document text summarization," *International Journal of Innovative Computing, Information and Control*, vol. 14, no. 4, pp. 1443–1452, 2018.

[13] S. S. Mohamed and S. Hariharan, "A performance study of text summarization model using heterogeneous data sources," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 16, pp. 2001–2007, 2018.

[14] Y. Kumar, K. Kaur and S. Kaur, "Study of automatic text summarization approaches in different languages," *Artificial Intelligence Review*, vol. 54, no. 8, pp. 5897–5929, 2021.

[15] Q. A. Al-Radaideh and D. Q. Bataineh, "A hybrid approach for Arabic text summarization using domain knowledge and genetic algorithms," *Cognitive Computation*, vol. 10, no. 4, pp. 651–669, 2018.

[16] A. Qaroush, I. A. Farha, W. Ghanem, M. Washaha and E. Maali, "An efficient single document Arabic text summarization using a combination of statistical and semantic features," *Journal of King Saud University–Computer and Information Sciences*, vol. 33, no. 6, pp. 677–692, 2019.

[17] N. Alami, M. Mallahi, H. Amakdouf and H. Qjidaa, "Hybrid method for text summarization based on statistical and semantic treatment," *Multimedia Tools and Applications*, vol. 80, no. 13, pp. 19567–19600, 2021.

[18] N. Alami, M. Meknassi, N. En-Nahnahi, Y. El Adlouni and O. Ammor, "Unsupervised neural networks for automatic Arabic text summarization using document clustering and topic modeling," *Expert Systems with Applications*, vol. 172, pp. 114652, 2021.

[19] K. Al-Sabahi, Z. Zhang, J. Long and K. Alwesabi, "An enhanced latent semantic analysis approach for Arabic document summarization," *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 8079–8094, 2018.

[20] N. Alami, N. En-Nahnahi, S. A. Ouatik and M. Meknassi, "Using unsupervised deep learning for automatic summarization of Arabic documents," *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 7803–7815, 2018.

[21] S. Abdulateef, N. A. Khan, B. Chen and X. Shang, "Multidocument Arabic text summarization based on clustering and word2vec to reduce redundancy," *Information*, vol. 11, no. 2, pp. 59, 2020.

[22] J. L. Vincent, R. Manzorro, S. Mohan, B. Tang, D. Y. Sheth *et al.,* "Developing and evaluating deep neural network-based denoising for nanoparticle tem images with ultra-low signal-to-noise," *Microscopy and Microanalysis*, vol. 27, no. 6, pp. 1431–1447, 2021.

[23] M. Said, A. M. El-Rifaie, M. A. Tolba, E. H. Houssein and S. Deb, "An efficient chameleon swarm algorithm for economic load dispatch problem," *Mathematics*, vol. 9, no. 21, pp. 2770, 2021.