Acta Scientiarum

http://periodicos.uem.br/ojs
ISSN on-line: 1807-8664
Doi: 10.4025/actascitechnol.v47i1.71378

COMPUTER SCIENCE

Bi-STAB: Enhanced abstractive text summarization model using hybridized deep learning network

Vanathi Dhinakaran[1], Abdullah Baihan[2], Ahmed Alhussen[3] and Radhakrishnan Paramasivan[4*]

[1]Department of Computer Science and Engineering, Nandha Engineering College, Vaikkaalmedu, Erode, Tamil Nadu, India. [2]Department Computer Science Engineering, Community College, King Saud University, Riyadh, Saudi Arabia. [3]Department of Computer Engineering, College of Computer and Information Sciences, Majmaah University, Al-Majmaah, Saudi Arabia. [4]School of Computer Science & Artificial Intelligence, SR University, Warangal, Telangana, 506371, India. *Author for correspondence. E-mail: rksiva13@gmail.com

ABSTRACT. Abstractive text summarization is based on Sequence-to-Sequence architecture which is developed to capture syntactic and contextual information between words. Abstractive summarization is needed to address information overload and enhance summary generation. However, it faces challenges such as maintaining consistency, capturing subtle nuances, and striking a balance between conciseness and comprehensiveness. To overcome these challenges, a novel Bi-directional Stacked GRU-LSTM for ABstractive text summarization (Bi-STAB) framework is proposed to generate an effective text summarization system using a hybridized deep learning network. Initially, the original document is pre-processed and fed to the hybridized stacked deep learning network for relevant feature extraction. The hybridized deep learning network extracts features from the text by capturing both forward and backward sequential dependencies by ensuring contextual information from both directions. An Attention-based Neural Network is used to select the most relevant parts of the text to form a coherent summary from these extracted features. Finally, the summary is then post-processed to enhance readability and provide a concise and accurate representation of the original text. A simulation of Bi-STAB is performed using MATLAB, and validation is performed using WCEP and Giga Word data. TBy comparing the accuracy of Bi-STAB with existing techniques such as DeepSumm, MOOTweetSumm, and WL-AttenSumm the proposed framework outperforms these techniques by 84.80, 88.35, 92.17, and 97.64%, respectively.

Keywords: Abstractive text summarization; stacked bidirectional gated recurrent unit; long short-term memory; attention-based convolutional neural network.

Received on Februaryy 21, 2024.
Accepted on September 3, 2024.

Introduction

Text summarization is the process of reducing a text's content without changing its originality or key points. An abbreviated form of the text is produced as part of the summary, and it frequently includes significant details from the original document (Raza & Shahzad, 2024; Jang & Kim, 2021). Humans can quickly analyze and comprehend lengthy materials when they are given a clear and high-quality summary (Shin, 2023). An efficient way to extract relevant information from lengthy texts is through Automatic Text Summarization (ATS) (Ulker & Ozer, 2024; Moratanch & Chitrakala, 2018).

ATS is a technology that was created to change with the information era. The amount of text that is available has increased rapidly due to the explosion of information (Alomari et al., 2023). The users' attention is to condense, summarize, and generalize big data material so they can peruse and comprehend it more rapidly (Cheng et al., 2020; Shin, Park & Song, 2023). At the moment, extractive text summarizing and abstract text summarization are the two categories into which ATS research directions can be separated (Zhuang & Zhang, 2019). The abstraction approach provides a summary by rewriting the material, while the extraction method extracts key information from the document (Movassagh et al., 2023).

ATS technology is used to understand all relevant information from the summary to the readers quickly and efficiently without having to read the entire document. Recently, text summarization has been popular and is being used in numerous industries, including news headline generation and summary report generation (Guo et al., 2019). Computers generally have a hard time comprehending textual content and producing automatic summaries. It must comprehend the text's content information to assess each word's significance, choose words wisely, and then duplicate, rearrange, and merge those choices to satisfy text automation specifications (Gnanamalar et al., 2023).

Text summaries produced by supervised statistical machine learning are frequently inaccurate and inconsistent because of the model's strong reliance on text feature quality. After various stages, RNN models might lose some information, which could lead to erroneous text summaries. Furthermore, in the big data era, its computational efficiency and cost are not able to satisfy the demands of enormous texts (You et al., 2020). The evaluation index scores have significantly increased as a result of these upgrades. There is still an opportunity for considerable improvement in terms of accuracy and minimization of repetition in the summaries produced by these models (Rajendran Arulappan et al., 2024; Guetari & Kraiem, 2023). The collaborative Adversarial Networks for expression recognition captures salient aspects between two phrases and implementing community-oriented learning based on mismatch learning to extract common features across many domains (Alzubi et al., 2020b). However, it faces inconsistent handling of long-term dependencies. The ABLG-CNN approach classify the lengthy Chinese texts and the filtering stage evaluates the BiLSTM and CNN output features to generate text generation features for classification. The experiments were conducted on two long-text news datasets which demonstrates the superior performance of the ABLG-CNN methodology (Deng et al., 2021). The Automatic summary of the embedded text enhances the correlation between the source text and output text summary by utilizing bidirectional attention-based LSTM and four Seq2Seq structured ATS models. The Automatic summary of the embedded text performs better when tested on two publicly available datasets (Jiang et al., 2021). However, it contains Challenges with out-of-vocabulary words. A productive deep-learning strategy for legal text summarization offers a straightforward generating mechanism for summarizing Indian court judgment texts using neural networks (Anand & Wagh, 2022). However, it contains a Tendency to produce repeated statements. A WL-AttenSumm technique which is a deep learning-based method for word-by-word text summarization uses a Bi-GRU convolutional network to identify syntactic and semantic linkages in text. The WL-AttenSumm technique is validated using the CNN/Daily Mail, DUC 2002, and CNN/Daily Common datasets and achieves an overall ROUGE and F1 scores are 42.9%, 19.7%, and 39.3% respectively (Gambhir & Gupta, 2022). However, it contains Risk of including erroneous factual details. An abstractive text summarization method incorporates attention and long-term memory into the convolutional neural network's encoder enhances the final summary's coherence and applicability. The improved ROUGE measure for summary generation is achieved and validated with CNN/Daily Mail and DUC-2004 datasets (Aliakbarpour, Manzuri & Rahmani, 2022). A DeepSumm method that leverages latent document information inferred from the sequence network and topic vectors to improve the quality and accuracy of text summarization. After validating the DeepSumm technique using synthetic data from CNN/DailyMail and DUC 2002, ROUGE-1, ROUGE-2, and ROUGE-L scores are obtained by 43.3, 19.0, and 38.0%, respectively (Joshi et al., 2023). A MOOTweetSumm method that accounts for the ideas in the corpus while summarizing opinion articles using a multi-objective segmentation which enables unsupervised extraction and synthesis based on the concepts from a corpus by employing a multi-objective pruning strategy. The MOOTweetSumm approach results in improvements of 11% and 9% for the ROGUE-1 and ROGUE-L metrics respectively (Gudakahriz et al., 2023). A linguistic feature space-based neural attention model for summarizing abstract material uses word weights, type tags, named entity tags, and other attributes to convert the extracted summary into an informative summary. A ROUGE score of 37.76% is obtained by the suggested strategy when it is validated using the CNN/DailyMail dataset (Dilawari et al., 2023). A content will be condensed using a generative model, the discriminator's task is to estimate the probability which makes it challenging to discern between the summary and the extracted summary. The CNN/Daily Mail dataset was used to validate the technique and achieves Red-1 and Red-L ratings of 41.58 and 38.96, respectively (Tank & Thakkar, 2024) However, it faces difficulty in maintaining semantic relevance. To overcome these issues, a novel Bi-STAB framework is proposed to generate an effective text summarization system using a hybridized deep learning network. The major contributions of the proposed Bi-STAB framework are given as follows,

- Initially, the original document is pre-processed using Text Stemming (TS), Stop Word Removal (SWR), Text Tokenization (TT), and Text Lemmatization (TL).
- After pre-processing, the Stacked BiGRU-LSTM extracts feature from the text by capturing both forward and backward sequential dependencies by ensuring contextual information from both directions.
- An Attention-based Convolutional Neural Network (CNN) analyzes these extracted features to produce a coherent summary by focusing on the most relevant parts of the text.

- After the summary is generated, it undergoes post-processing for enhanced readability, so that the final summarized text provides an accurate representation of the original content.

The rest of the document is organized as follows: Section 2 holds a summary of earlier research on text summarization. In Section 3, the proposed Bi-STAB model and its architecture are explained. Section 4 presents the findings of the experiment and a discussion. The research is finally concluded in Section 5 with future scope.

Materials and methods

Summarization is the process of concisely conveying the main idea of the source text in a brief summary. Abstractive summarization paraphrase using advanced and nearer-to human explanation by adding novel words or phrases. Existing studies use a variety of DL-based neural methods to address a range of issues, including long-term dependencies, words not in the vocabulary, repeated statements, erroneous factual details, and difficulties maintaining semantic relevance which are critical for information, fidelity, and output control to improve abstract summarization results. However, in this research, a Bi-STAB framework is proposed which is DL-based model tends to be abstractive rather than exploitative and produces acceptable results with a very high novelty.

Bi-directional stacked GRU-LSTM for ABstractive text summarization

In this research, a novel Bi-directional Stacked GRU-LSTM for ABstractive text summarization (Bi-STAB) framework is proposed to generate an effective text summarization system using a hybridized deep learning network.

Initially, the original document is pre-processed using Text Stemming (TS), Stop Word Removal (SWR), Text Tokenization (TT), and Text Lemmatization (TL). In preprocessing, the words in the input text are reduced to their root forms via TS, and the common non-essential words are eliminated through SWR. The TT technique splits the input text into individual words or sentences and the words are reduced to their base form using TL. After pre-processing, the Stacked BiGRU-LSTM extracts features from the text by capturing both forward and backward sequential dependencies by ensuring contextual information from both directions. An attention-based convolutional neural network (CNN) that focuses on the most pertinent textual segments is fed the collected features to provide a coherent summary. After the generated summary is post-processed to enhance readability, the final summary text offers a concise and accurate summary of the original content. The overall structure of the proposed Bi-STAB framework is illustrated in Figure 1.

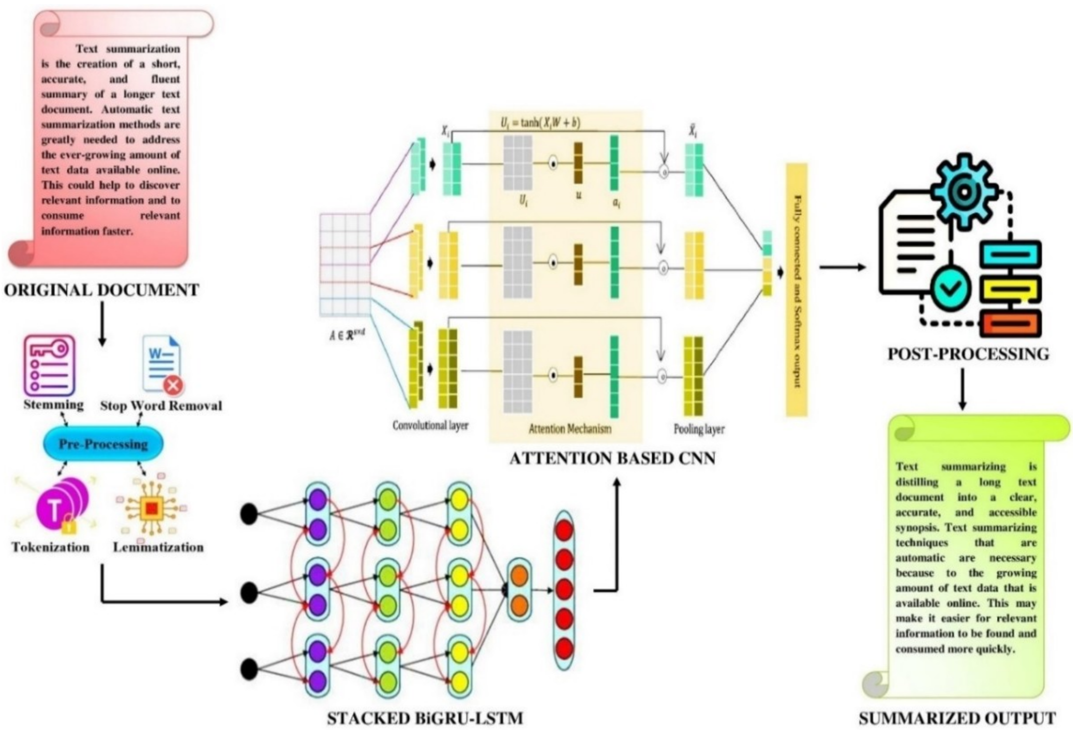

Figure 1. Overall structure of the proposed Bi-STAB framework.

Dataset description

The significance of the proposed Bi-STAB approach is examined in this study using the Giga Word and Wikipedia News Portal (WCEP) dataset. The WCEP dataset is composed of compilations of English-language articles that are summarized by hand using a technique called Multi-Document Summarization (MDS). The collection, which is organized into 10,200 categories, has 10.2 million documents (2.39 million in JSON format). A collection of Giga Word article pairs is used to create headlines for the almost 4 million English articles that comprise the Giga Word dataset. The corpus, which consists of documents published between the 1950s and the present, is intended to be representative of the materials included in it. The most recent revision to abstract writing gradually does away with subject-verb agreement.

Pre-processing

To ensure or improve efficiency, pre-processing is the process of modifying or removing data before use. Pre-processing is an integral part of the data mining process. The pre-processing methods that are most frequently employed are Text Stemming (TS), Text Lemmatization (TL), Text Tokenization (TT), and Stop Word Removal (SWR). ATS pre-processing stages frequently involve the application of these pre-processing techniques.

i) Text Stemming: The technique of restoring words with varied tenses to their original form is used by TS.

ii) Text Tokenization: The input document is separated into individual words by the TT.

iii) Text Lemmatization: It is superior to TS because TL examines word morphology.

iv) Stop Word Removal: The word "SWR" is frequently utilized in written correspondence. It's unclear because these two lines contradict one another. After pre-processing, the document's output such as a list of tokens with stop words removed, and words reduced is fed to the Stacked BiGRU-LSTM model for feature extraction.

Stacked BiGRU-LSTM Network

Stacked BiGRU-LSTM networks extract text features while capturing forward and backward sequential dependencies to guarantee that contextual information is recorded in both directions. Stacked BiGRU-LSTM, a type of Recurrent Neural Network (RNN) layer, processes sequence data. The approach leverages the context of previous executions and extracts features from historical data to determine the current result. The main goal of the stacked BiGRU-LSTM layer is data extraction from the input matrix elements. Both forward and inverted GRU models are supplied into the BiGRU-LSTM classifier. The forward and inverted GRU models are applied to the data processing. When the input text, "X," is evaluated by the GRU model at a specific point in time known as the "hidden state," an "inverse GRU" model is developed.

$$H_t' = (x_t, H_{t-1'}) \tag{1}$$

$$H_t^{"} = \left(x_t, H_{t-1^{"}}\right) \tag{2}$$

The front and back GRUs are combined to develop the Attention layer. The phrase might also contain terms that are just tangentially related to the main summary or that are completely unrelated. The Attention layer thus assigns different weights to different words in the phrase. The input feature matrix is categorized using the Fully Connected layer. An overview of this layer's output may be found below.

$$P = f(M * X) + D \tag{3}$$

This layer, whose values range from 0 to 1, represents the input element. When a text's value is close to zero, it has an overall negative pole; when it is close to one, it has an overall positive pole. The f represents the activation function. M indicates the weight matrix. The offset is denoted by D. LSTM leverages the benefits of the RNN architecture by functioning as an improved RNN. The issues with vanishing and exploding gradients are effectively resolved by the unique gate structure of RNN. Applications including voice recognition, natural language processing, and image classification have benefited from LSTM models for handling long sequences. The time is shown by Zt, the output from the previous hour is represented by Yt−1, and the position of the neuron is indicated by Ct−1. The forget gate either retains or does not retain the Ct-1 unit from the previous hour to the current hour. The charge at the forget gate's input is the output charge from the preceding time, Ct-1. The last function that generates ft is the leftmost sigmoid activation function. A formula for calculating it can be found in equation (4).

$$f_t = \sigma\left(W_f \cdot [Y_{t-1}, Z_t] + b_f\right) \tag{4}$$

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{5}$$

where Wf is the forgetting gate weight matrix, bf is the bias vector, and σ is the Sigmod activation function. The variable ft, whose value ranges from 0 to 1, represents the probability that the last neuron layer is skipped. Equations (6) to (8) are used to calculate the input gate, which selects the new input Zt that can be stored in the neurons. ZT and VT are the two main types of neurons.

$$i_t = \sigma(W_i \cdot [Y_{t-1}, Z_t] + b_i) \tag{6}$$

$$V_t = tanh(W_c \cdot [Y_{t-1}, Z_t] + b_c) \tag{7}$$

$$tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{8}$$

The weight matrix includes Wi and Wc. Bi and bc are the polarization vectors. The percentage of current data that must be retained is expressed by it and vt. To create updated neuron state information Ct, it is added to the neuron state that was recorded at the previous time point. Equation (9) illustrates the computation.

$$C_t = C_{t-1} \cdot f_t + i_t v_t \tag{9}$$

The output of the neuron state is processed by the output gate before being sent to the next neuron. Equation (10), which defines the OT output gate.

$$O_t = \sigma(W_o[Y_{t-1}, Z_t] + b_o) \tag{10}$$

where Yt is the neuron's output at time T, bo is the error vector and wo is the weight matrix of the output gate. The final summarized value from the last layer computation can be found using the formula in equation (11).

$$Y_t = O_t \cdot tanh(C_t) \tag{11}$$

Several variables are fed into the combined summarization model. First, the training data for the GRU layer, has a more straightforward network structure, fewer parameters, and a low summarization time. Compared to LSTM, GRU trains more quickly and needs less training time, but it is less extractive. Lastly, the model extracts a keyword by considering several variables. To concentrate on the most pertinent text segments and produce a cogent summary, the attention-based CNN is fed with extracted features, such as hierarchical representations, sequential patterns, long-term dependencies, bidirectional context, and contextual word embeddings.

Attention based CNN approach

CNN uses the features that are extracted from the content to generate a satisfactory synopsis. On CNN, each word is associated with a certain target section. Each word in the series provides a distinct element that improves the text's topic classification. For subject classification, keywords are especially crucial when compared to other words. Using a word vector attention mechanism, this is achieved by comparing the correlation coefficients of each word vector in the sequence. Thus, more significant phrases will be more noticeable. Consider a text string of K sentences, each with a length of m. Relevant features of keywords are highlighted by calculating the word context variables.

$$g_{i,j} = \sum_{k=1,k\neq j}^{m} \alpha_{ij,k} X_{i,k} \tag{12}$$

The attention weight of formula 12, $\alpha_{ij,k}$, is referred to as "attention weight". In equation 13, the correlation coefficient between terms in the i-th and j-th sentences is denoted by $\alpha_{ij,k}$.

$$\alpha_{ij,k} = \frac{\exp\left(score(X_{i,j}, X_{i,k})\right)}{\sum_{j'=1,j'\neq j}^{m} \exp\left(score(X_{i,j}, X_{i,j'})\right)} \tag{13}$$

A Rating System Calculate the correlation coefficient between a word and other sentences using Equation 13 to rank the terms relevant to the current phrase. A higher score in the context vector denotes a higher weight for the present word. The score value is calculated using the formula in equation (14).

$$score\left(X_{i,j}, X_{i,k}\right) = v_a^T \tanh\left(w_a X_{i,j} + w_u X_{i,k}\right) \tag{14}$$

The training parameters are represented by the symbols v_a, w_a, and w_u in Equation 14. Using w_a, and w_u, respectively, $X_{i,j}$ and $X_{i,k}$ are mapped and transformed into matrix mappings. The outcomes for $X_{i,j}$ and $X_{i,k}$ are then produced by v_a. The word vector $X'_{i,j} = [X_{i,j}, g_{i,j}]$ will be represented by concatenating the word vector with its context vector. An n-length text string is produced by concatenating all of the sentences. The Stacked BiGRU-LSTM entry is represented by the string $D = \{X'_1, X'_2, X'_3, \dots \dots . X'_n\}$. To make the document easier to read, post-processing will be applied to the extracted summary at the conclusion.

Post-processing

The post-processing module makes use of the following strategies to enhance textual appearance and make data interpretation and analysis easier by completing summary boxes, generalizing phrases, fixing clusters, and cutting down on redundancy.

Redundancy elimination

These two sentences might signify the same thing even though they appear in different locations in the document, therefore it might be appropriate to include both of them in the summary. Using a threshold computation to ascertain the degree of similarity between each sentence is the solution to this issue. When two phrases are too similar to be used separately, they are considered duplicates. The summary ignores all other sentences and only presents the sentence with the highest similarity weight by using the formula below to find similarity:

$$sim(\vec{x}, \vec{y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}} \quad (15)$$

Among it, $\vec{x}, \vec{y}$ are sentence vectors using VSM, and n is the dimension of the vector.

ii) Adjustment of clustering based on paragraph

Summaries frequently omit the important point-highlighting sentences because they don't make much sense. The summary is nevertheless supported even with the inclusion of these sentences. This enables the generation of a succinct summary by automatically extracting key sentences from paragraphs. A sentence can be added to the summary or deleted if it is missing from the paragraph after it has been grouped. This procedure can be recognized from the following algorithm.

iii) Sentence Generalization

Due to their extensive use of technical terminology, business documents can be challenging to read. The best approach to this problem is to make coarse generalizations. The class tree of a text shows that the probability of a term gaining a broader understanding increases with its node class. Therefore, when using generalizations in a text, the technical terminology must serve as a tree structure.

iv) Infilling of summary frame

The method begins with a highly effective Bayesian text classification system. The categories, content, and keywords form the foundation of the summary structure which contains six subjects and three categories that are used by CTS to classify documents. Based on the information at hand, the links between domains are then chosen as a table to be incorporated into the framework. Individuals, groups, and connections to other professions develop the framework. To sum up, the results of the post-processing phase provide an accurate and cohesive representation of the original data.

Result and discussion

In this section, to develop the proposed Bi-STAB model, an exploratory analysis was conducted using multiple datasets for Automatic Text Summarization (ATS). This section describes the datasets that are utilized for training and testing the proposed Bi-STAB model. The performance data and underlying DL models that were used for comparison are included in this section.

Performance analysis

A number of statistical measures are used to evaluate the efficacy of the classification method, including the F1 score, accuracy, precision, and recall.

As Figure 2a demonstrates, there is a correlation between the number of epochs and the reliability of the Bi-STAB training and validation as assessed by this research. The accuracy of the model increases with the number of iterations, beginning modestly. The model overfits and the accuracy declines after 50 iterations of training and validation, respectively. Training and validation losses are essentially eliminated by the deep learning model. As shown in Figure 2b the training and validation losses may be calculated using the number of epochs. Up to the 50th epoch, the training and validation losses increase as expected. The performance of the model becomes unstable due to overfitting

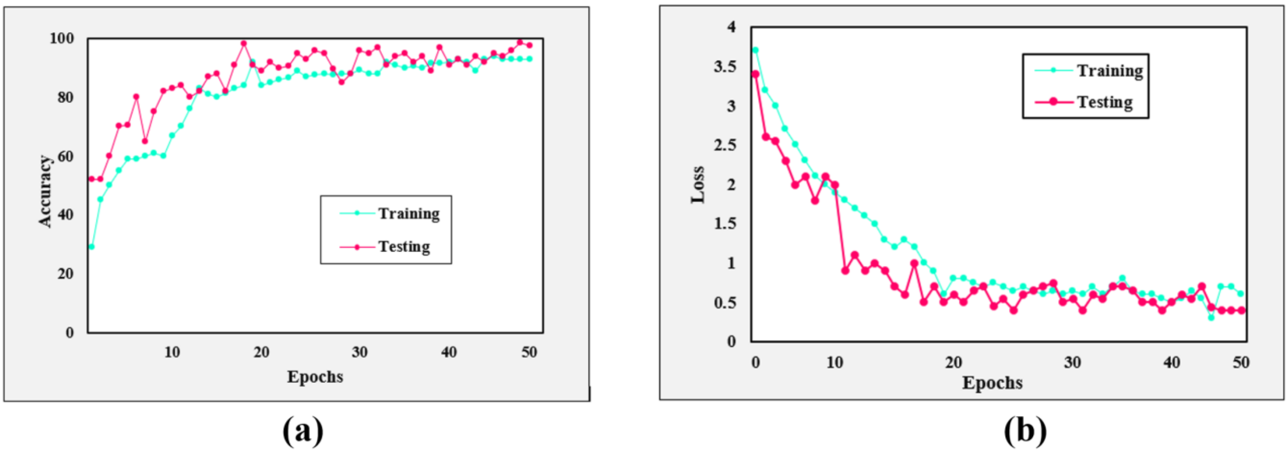

Figure 2. (a,b) accuracy and loss of Bi-STAB Framework.

Table 1 presents an outcome of the Abstractive Summarization Output which offers accurate, concise text summaries that are thorough enough to be efficiently readable. Some of the accurately summarized outputs for the users are depicted below in Table 1.

Table 1. Abstractive Summarization Output.

Input	Summarized Output
Roger Federer defeated Marin Cilic in three sets to win his seventh Wimbledon men's singles title. Federer won the match after an hour and forty-five minutes due to a foot issue that prevented him from participating in the game. Roger Federer has won 19 Grand Slam events since taking home the trophy at the Australian Open earlier this year.	In the Wimbledon final, Roger Federer triumphed over Marin Cilic to win a record-tying eighth championship.
Andy Murray, the reigning Open champion, defeated American Donald Young 6-0, 6-2 on Monday to win the second round of the competition.	Andy Murray's reputation grew dramatically during the 2010 FIFA World Cup.
Law enforcement activities have long been associated with the terrorist group known as ISIS. America launched airstrikes against its fighters in retaliation for its actions in Iraq, and it may consider additional strikes against Syria.	One aspect of the US military's program is considered to be airstrikes.
After an interview on CNN and an incredible amount of donations from people worldwide, Dr. Kekula's medical coat from Emory University was acquired.	Kekula decided to wear a hospital gown when the CNN story broke and donations began rolling in from all around the world.
Vietnam's agricultural industry is in decline as a result of insufficient assistance from the government and foreign investors, and local farmers are now obliged to pay to take part in programs designed to build rural roads.	Vietnam's agricultural output is insufficient.

Comparative analysis

The pre-processing data analysis is assisted by CoreNLP through the use of Natural Language Translators (NLTs) and other NLP technologies.

However, each NLP technique has advantages of its own, and accuracy in word segmentation is likely to be higher when using CoreNLP than when using other tools. By comparing the average repeat rate per word in the pre-text and post-text, Figure 3 shows the pre-processing reduced the word redundancy to 1,512 and 1,798 respectively.

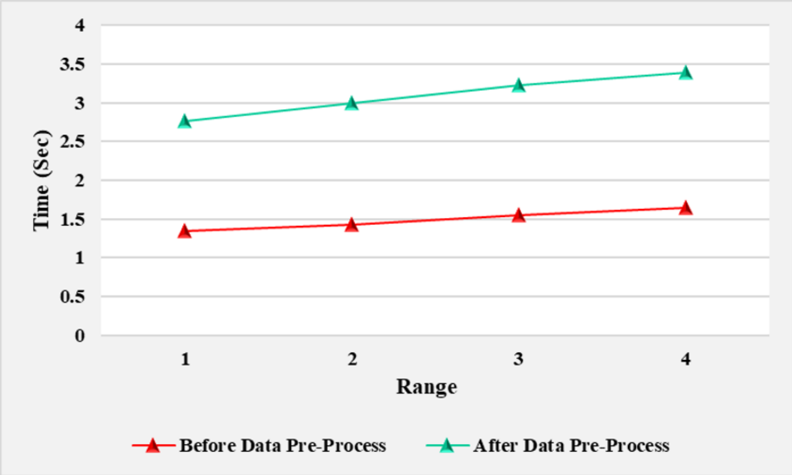

Figure 3. Time Vs Data pre-processing.

The evaluation of ROUGE Score is presented in Table 2. The automated evaluation methods for comparison using RED. The RED Number (RED N) can be found by comparing the n-gram total with the reference sum or the candidate summary with the reference summary. Comparison of ROUGE Score shown in Figure 4.

Table 2. Evaluation of ROUGE Score.

Model	ROUGE-N (%)	ROUGE-L (%)
DeepSumm	43.18	40.35
MOOTweetSumm	38.74	33.43
WL-AttenSumm	36.23	32.21
Bi-STAB	44.17	29.62

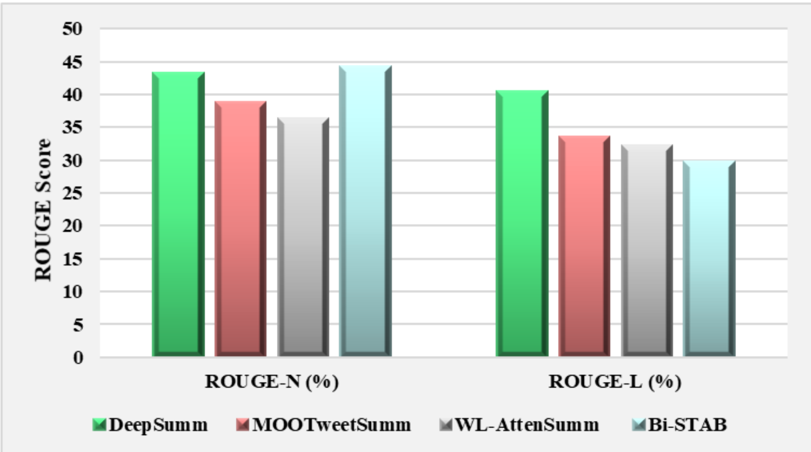

Figure 4. Comparison of ROUGE Score.

The graph of confidence is displayed in Figure 5. It demonstrates the model's confidence in its predictions and their accuracy related to each other. The abstract models frequently overstate their estimates since they are calibrated to Giga Words rather than WCEPs. This might be due to the shortened summaries provided by Giga Word, which reduce the possibility of visible bias.

A comparison of every line extracted from the original text forms the basis of the summary presented in Figure 6. A summary should contain between 30 to 45% of the original information, however, some contain as much as 85% respectively. There are roughly 45 lines that are divided into subsections, as seen in Figure 6.

Figure 7, provides a comparison of the accuracy achieved by three different models used for abstractive text summarization which is the Stacked BiGRU-LSTM, Attention CNN, and Bi-STAB approach which is a combined Stacked BiGRU-LSTM with Attention CNN model. The proposed Bi-STAB hybrid model combines the strengths of both the Stacked BiGRU-LSTM and Attention CNN architectures. The BiGRU-LSTM part is responsible for extracting deep contextual features from the text, while the attention-based CNN efficiently generates summaries by focusing on the most important information. The accuracy of the proposed Bi-STAB framework is 85.3, 83.7, and 87.8% better than that of existing methods such as stacked BiGRU-LSTM and attention CNN.

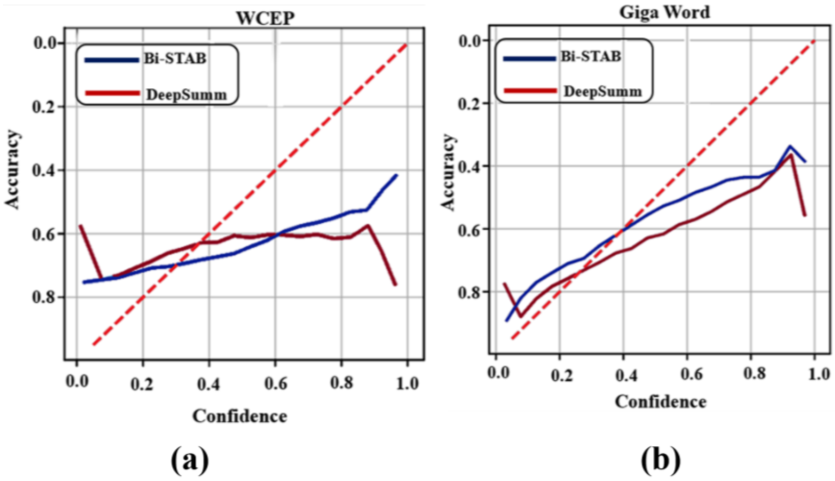

Figure 5. Accuracy Vs Confidence.

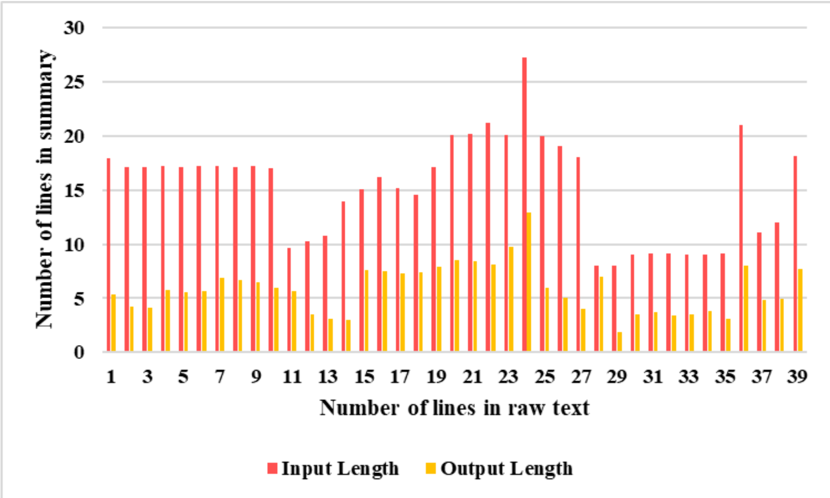

Figure 6. Total number of lines in summary Vs raw text.

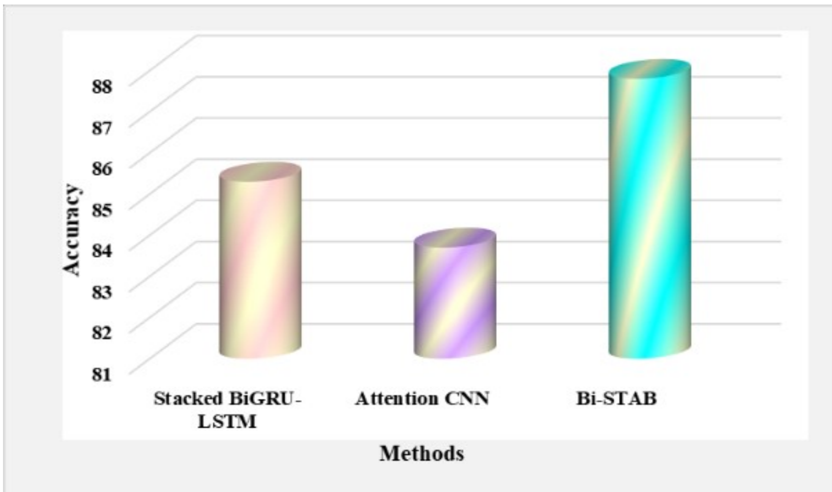

Figure 7. Accuracy Comparison of ATS Models.

The comparison of generated summaries with and without post-processing is depicted in Figure 8. The post-processing comparison is done between four aspects such as grammar and syntax, redundancy removal, content refinement, and overall readability which is used for post-processing in abstractive text summarization. Without post-processing the summary, the proposed Bi-STAB approach achieves an accuracy of 75% for grammar and syntax, 65% for redundancy removal, 68% for content refinement, and 70% for overall readability. Conversely, in the post-processed summary, the proposed Bi-STAB approach achieves an effectiveness of 92% for grammar and syntax, 88% for redundancy removal, 91% for content refinement, and 93% for overall readability respectively.

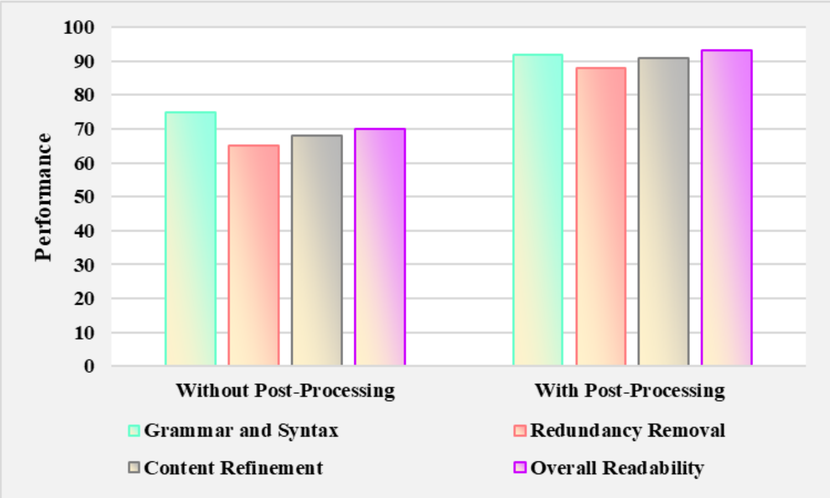

Figure 8. Comparison of Generated Summary With and Without Post-Processing.

The accuracy, sensitivity, recall, precision, and recall of the proposed strategy are compared with existing methods in Figure 9. With an accuracy of 96.74% for both datasets, the Bi-STAB method is significantly more accurate than existing models like DeepSumm, MOOTweetSumm, and WL-AttenSumm respectively.

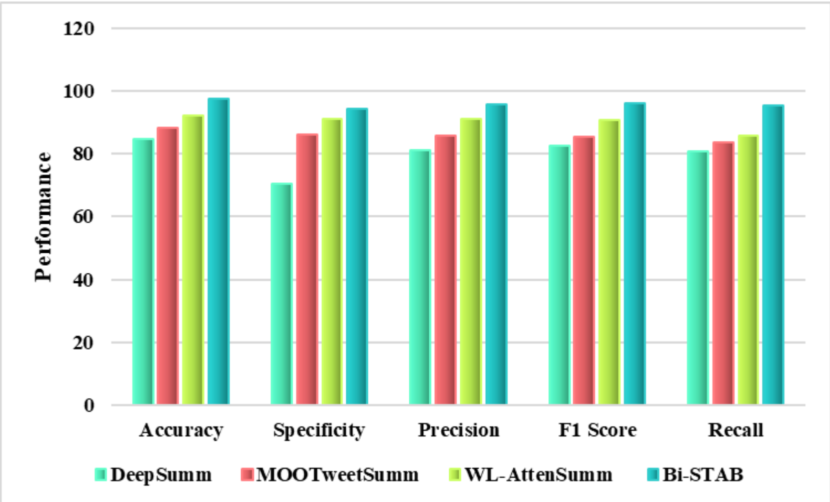

Figure 9. Performance comparison of Bi-STAB framework.

Discussion

In this research, a novel Bi-directional Stacked GRU-LSTM for ABstractive text summarization (Bi-STAB) framework is proposed to generate an effective text summarization system using a hybridized deep learning network. The proposed Bi-STAB framework is assessed and compared with a broad range of metrics including data pre-processing time, confidence accuracy, rouge score, number of lines in the summary, accuracy of ATS Models, post-processing performance, accuracy, specificity, precision, recall, and F1 score respectively. In data pre-processing time, the pre-processing data analysis is assisted by CoreNLP through the use of NLTs and other NLP technologies. Figure 3 shows that the pre-processing time reduced the word redundancy to 1,512 and 1,798 respectively. In the proposed Bi-STAB framework, the ROUGE score comparison is depicted in Figure 4 and the ROUGE score evaluation is given in Table 2. In the ROUGE score comparison, the proposed Bi-STAB framework achieves 44.17% for ROUGE-N and 29.62% for ROUGE-L respectively. The graph of confidence is displayed in Figure 5, which demonstrates the proposed Bi-STAB model's confidence in its predictions and their accuracy. A comparison of every line extracted from the original text forms the basis of the summary presented in Figure 6. In this comparison, a summary should contain between 30% to 45% of the original information and there are roughly 45 lines that are divided into subsections. A comparison of the accuracy achieved by three different models used for ATS which is the Stacked BiGRU-LSTM, Attention CNN, and Bi-STAB approach is illustrated in Figure 7. In comparison, the proposed Bi-STAB framework which is the combination of stacked BiGRU-LSTM and attention CNN network achieves 85.3%, 83.7%, and 87.8% of accuracy which is better than that of existing methods. In the proposed ATS approach, a comparison of generated summaries with and without post-processing is depicted in Figure 8. Finally, the performance comparison of the proposed Bi-STAB framework is depicted in Figure 9. The Bi-STAB method is significantly more accurate by achieving 96.74% of accuracy than existing models like DeepSumm, MOOTweetSumm, and WL-AttenSumm respectively.

Conclusion

This paper proposed a novel Bi-directional Stacked GRU-LSTM for ABstractive text summarization (Bi-STAB) framework to generate an effective text summarization system using a hybridized deep learning network. Comparison and validation of the proposed model are conducted on the Giga Word and WCEP datasets using the CNN attention network and Bi-Gru Stack model. The Bi-STAB framework is compared to DeepSumm, MOOTweetSumm, and WL-AttenSumm Based on the accuracy, specificity, precision, F1 score, recall, and ROUGE score. The accuracy of the proposed Bi-STAB framework is 97.64% higher than that of the DeepSumm method which is 84.80%, MOOTweetSumm method which is 88.35%, and WL-AttenSumm method, which is 92.17% respectively. In the future, this research could involve testing the Bi-STAB framework on additional, more diverse datasets, including multilingual datasets, to assess its generalizability and robustness across different languages and text genres. In addition, the proposed framework can be adapted for real-time text summarization applications. Investigating its performance in real-time scenarios and optimizing the model for faster inference without compromising accuracy could make the Bi-STAB system more practical for deployment in real-world applications.

Acknowledgments

The author extends the appreciation to the Deanship of Postgraduate Studies and Scientific Research at Majmaah University for funding this research work through the project number (R-2024-1295).

References

Aliakbarpour, H., Manzuri, M. T., & Rahmani, A. M. (2022). Improving the readability and saliency of abstractive text summarization using combination of deep neural networks equipped with auxiliary attention mechanism. *The Journal of Supercomputing, 78*(2), 2528–2555. https://doi.org/10.1007/s11227-021-03950-x

Alomari, A., Al-Shamayleh, A. S., Idris, N., Sabri, A. Q. M., Alsmadi, I., & Omary, D. (2023). Warm-starting for improving the novelty of abstractive summarization. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3322226

Alzubi, J. A., Jain, R., Kathuria, A., Khandelwal, A., Saxena, A., & Singh, A. (2020). Paraphrase identification using collaborative adversarial networks. *Journal of Intelligent & Fuzzy Systems, 39*(1), 1021–1032. https://doi.org/10.3233/JIFS-191933

Anand, D., & Wagh, R. (2022). Effective deep learning approaches for summarization of legal texts. *Journal of King Saud University - Computer and Information Sciences, 34*(5), 2141–2150. https://doi.org/10.1016/j.jksuci.2019.11.015

Cheng, J., Zhang, F., & Guo, X. (2020). A syntax-augmented and headline-aware neural text summarization method. *IEEE Access, 8*, 218360–218371. https://doi.org/10.1109/ACCESS.2020.3042886

Deng, J., Cheng, L., & Wang, Z. (2021). Attention-based BiLSTM fused CNN with gating mechanism model for Chinese long text classification. *Computer Speech & Language, 68*, 101182. https://doi.org/10.1016/j.csl.2020.101182

Dilawari, A., Khan, M. U. G., Saleem, S., & Shaikh, F. S. (2023). Neural attention model for abstractive text summarization using linguistic feature space. *IEEE Access, 11*, 23557–23564. https://doi.org/10.1109/ACCESS.2023.3249783

Gambhir, M., & Gupta, V. (2022). Deep learning-based extractive text summarization with word-level attention mechanism. *Multimedia Tools and Applications, 81*(15), 20829–20852. https://doi.org/10.1007/s11042-022-12729-y

Gnanamalar, A. J., Bhavani, R., & Arulini, A. S. (2023). CNN–SVM based fault detection, classification and location of multi-terminal VSC–HVDC system. *Journal of Electrical Engineering & Technology, 18*, 3335–3347. https://doi.org/10.1007/s42835-023-01391-5

Gudakahriz, S. J., Moghadam, A. M. E., & Mahmoudi, F. (2023). Opinion texts summarization based on texts concepts with multi-objective pruning approach. *The Journal of Supercomputing, 79*(5), 5013–5036. https://doi.org/10.1007/s11227-022-04842-4

Guetari, R., & Kraiem, N. (2023). CoMod: An abstractive approach to discourse context identification. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3302179

Guo, Q., Huang, J., Xiong, N., & Wang, P. (2019). MS-pointer network: Abstractive text summary based on multi-head self-attention. *IEEE Access, 7*, 138603–138613. https://doi.org/10.1109/ACCESS.2019.2941964

Jang, H., & Kim, W. (2021). Reinforced abstractive text summarization with semantic added reward. *IEEE Access, 9*, 103804–103810. https://doi.org/10.1109/ACCESS.2021.3097087

Jiang, J., Zhang, H., Dai, C., Zhao, Q., Feng, H., Ji, Z., & Ganchev, I. (2021). Enhancements of attention-based bidirectional LSTM for hybrid automatic text summarization. *IEEE Access, 9*, 123660–123671. https://doi.org/10.1109/ACCESS.2021.3110143

Joshi, A., Fidalgo, E., Alegre, E., & Fernández-Robles, L. (2023). DeepSumm: Exploiting topic models and sequence to sequence networks for extractive text summarization. *Expert Systems with Applications, 211*, 118442. https://doi.org/10.1016/j.eswa.2022.118442

Moratanch, N., & Chitrakala, S. (2018). A novel framework for semantic oriented abstractive text summarization. *Journal of Web Engineering, 17*(8), 675–715. https://doi.org/10.13052/jwe1540-9589.1784

Movassagh, A. A., Alzubi, J. A., Gheisari, M., Rahimi, M., Mohan, S., Abbasi, A. A., & Nabipour, N. (2023). Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. *Journal of Ambient Intelligence and Humanized Computing*. https://doi.org/10.1007/s12652-020-02623-6

Raza, H., & Shahzad, W. (2024). End to end Urdu abstractive text summarization with dataset and improvement in evaluation metric. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3377463

Shin, J., Park, S. B., & Song, H. J. (2023). Token-level fact correction in abstractive summarization. *IEEE Access, 11*, 1934–1943. https://doi.org/10.1109/ACCESS.2022.3233854

Shin, Y. (2023). Multi-encoder transformer for Korean abstractive text summarization. *IEEE Access, 11*, 48768–48782. https://doi.org/10.1109/ACCESS.2023.3277754

Tank, M., & Thakkar, P. (2024). Abstractive text summarization using adversarial learning and deep neural network. *Multimedia Tools and Applications, 83*(17), 50849–50870. https://doi.org/10.1007/s11042-023-17478-0

Ulker, M., & Ozer, A. B. (2024). Abstractive summarization model for summarizing scientific article. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3420163

You, F., Zhao, S., & Chen, J. (2020). A topic information fusion and semantic relevance for text summarization. *IEEE Access, 8*, 178946–178953. https://doi.org/10.1109/ACCESS.2020.2999665

Zhuang, H., & Zhang, W. (2019). Generating semantically similar and human-readable summaries with generative adversarial networks. *IEEE Access, 7*, 169426–169433. https://doi.org/10.1109/ACCESS.2019.2955087