IJICTR
International Journal of Information & Communication Technology Research

Volume 17- Number 1 – 2025 (27-35)

Extractive Text Summarization in Persian Language through Integrative Approaches

Mohammad Rabiei*
Department of the Electrical and Computer Engineering, University of Eyvanekey, Semnan, Iran
Mohammad.rabiei@uniud.it

Received: 22 May 2024 – Revised: 6 June 2024 - Accepted: 17 August 2024

Abstract—**Text summarization is the process of condensing a source text while retaining its key points, tailored to a specific audience or task. The research extractive summarization, where each news article was segmented into individual sentences. Each sentence underwent processing through the ParsBERT algorithm. Subsequently, an attention layer combined the sentence weights with the Bidirectional GRU algorithm's output to extract summarized sentences for labeling. The dataset comprised over 175,000 articles sourced from reputable Persian news agencies (ISNA-TASNIM), covering various topics such as science, politics, and sports. Evaluation of the summarization techniques was conducted using Rouge metrics. The results of the investigation revealed precision values of 0.7923 (Rouge-1), 0.7613 (Rouge-2), and 0.8582 (Rouge-L). The study also evaluated the effectiveness of Gated Recurrent Unit (GRU) algorithms in extractive summarization by integrating its architecture with the attention network. The results demonstrated an improvement in news text summarization compared to other deep learning hybrid algorithms.**

Keywords: text summarization, Persian news, deep learning, attention network, extractive summarization

Article type: Research Article

I. INTRODUCTION

Summarization is the process of distilling the essence of a document, a task that, when performed manually, can be both laborious and overwhelming.

Dealing with vast volumes of unstructured or semi-structured data poses a significant challenge in the realm of text data [1]. The exponential growth of textual data in Persian news and news agencies, coupled with the diminishing time people allocate for reading, has intensified the need for Automatic Text Summarization (ATS) techniques. Automatic text summarization is an approach that condenses lengthy documents into a few sentences or words, encapsulating the essence and essential information of the document [2]. Condensing lengthy news texts into concise summaries facilitates rapid comprehension of crucial information on news websites. Summarizing text can be accomplished through extractive or abstractive methods [3]. Abstractive summarization, which involves rephrasing the text using one's own words, presents a more significant challenge for machines as it requires careful consideration of user language in each instance [4].

Moreover, abstractive text summarization may lead to potentially misleading and biased outcomes. In response, industries are increasingly favoring extractive methods, wherein essential sentences from the input document are chosen and combined to create a

* Corresponding Author

summary [5]. Unfortunately, most pretrained models are currently only available for English. Summarizing Persian language text poses challenges due to sentence structure nuances and the absence of optimized models for this language. The distinct linguistic features of Persian necessitate tailored algorithms. Web documents vary in text structure, adding complexity to the algorithm's adaptability. While this approach relies on extractive text summarization, recent advancements have introduced neural network-based techniques such as ' BERT ', which have gained popularity in this field.

However, BERT encounters limitations in summarizing lengthy documents due to input length constraints and the increase in the number of inputs [6]. In pursuit of a more efficient solution, we propose a novel approach. This approach harnesses the capabilities of BERT in collaboration with bidirectional GRUs (bidirectional gated recurrent units), an attentional network adept at capturing sequential dependencies in text to extract salient information.

Currently, over 110 million people in various countries such as Iran, Afghanistan, and Tajikistan speak Persian, and this number continues to rise steadily.

Consequently, in recent years, the proliferation of news websites offering Persian content has surged dramatically, with Persian now being used to create content on 3.3% of all known websites globally. This surge in content creation has elevated the status of the Persian language on the internet, ranking 8th in 2022 and climbing to 5th in 2023 and 2024 among the most popular content languages. Moreover, in 2021 and 2022, Persian secured the 3rd and 2nd spots respectively among the fastest-growing content languages.

This remarkable growth underscores the increasing global popularity and expansion of the Persian language, emphasizing the importance of studying and analyzing Persian language trends [7].

The effectiveness of this approach is assessed through the analysis of proposed techniques using the datasets of two Persian news agencies (ISNA-TASNIM-FARS NEWS-…), containing covering various topics such as science, politics, and sports articles [8]. This comprehensive strategy enables us to attain a profound understanding of the entire document's context, and we assess the effectiveness of our proposed approach using standard evaluation metrics such as ROUGE-score. The remainder of the paper is organized as follows: Section 2 provides an overview of previous work related to automatic text summarization. In Section 3, we detail the proposed model and its architecture. Section 4 covers the experimental setup, encompassing datasets, baseline models, hyperparameters, and evaluation metrics employed in the study. Experimental results and discussions are presented in Section 5. Finally, Section 6 summarizes the main findings of the paper and suggests potential future research directions.

II. BACKGROUND AND RELATED WORKS

Numerous studies have addressed the text summarization problem, employing extractive, abstractive, and hybrid methods [9]. Recent literature on extractive methods[5] categorizes them into statistical-based [10], concept-based [11], graph-based [12], clustering-based [13], topic-based [14], semantic-based [15], etc.

To tackle the challenges of text summarization, researchers have explored various architectures such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Recurrent Neural Network (RNN), Transformer, and attentional encoder-decoder models [16]. LSTM and GRU, both recurrent neural networks, excel at learning long-term dependencies in text. However, GRU is generally considered more efficient than LSTM, making it a preferred choice for large-scale text summarization tasks [3]. Recurrent Neural Network (RNN), a simpler architecture, is less efficient at learning long-term dependencies [17]. The Transformer, a newer architecture based on attention mechanisms, allows the model to focus on specific parts of the text when generating a summary, making it more effective than RNN-based models [18].

The choice of architecture is task and dataset-dependent. For instance, GRU-based models may be more suitable for tasks requiring short summaries, while Transformer-based models may excel in tasks demanding accurate summaries. The selection of architecture is a crucial consideration, aligning with the specific requirements of the task and dataset.

In a study conducted by Adelia in 2019, Recurrent Neural Network (RNN) demonstrated success in summarizing abstractive texts in English and Chinese. The Bidirectional Gated Recurrent Unit (Bi_GRU) RNN architecture was employed to ensure that the resulting summaries were influenced by the surrounding words. This research extends the application of such a method to Bahasa Indonesia, aiming to enhance text summarizations commonly developed using extractive methods with low inter-sentence cohesion [19].

The GRU-RNN based sequence model integrates absolute and relative positions of phrases, along with previously chosen summaries, to mitigate issues related to duplication. Its versatility allows training for both extractive and abstractive summarization tasks. Neural extractive summarization models commonly utilize a hierarchical encoder for document encoding, and their training often involves sentence-level labels created through rule-based heuristic methods. In a study conducted by Zhang et al., they propose HBERT (Hierarchical Bidirectional Encoder Representations from Transformers) for document encoding. Additionally, they introduce a pre-training method using unlabeled data. The randomly initialized counterpart achieved an improvement of 1.25 ROUGE on the CNN/Dailymail dataset and 2.0 ROUGE on a version of the New York Times dataset [20].

The performance of extractive summarization can be improved by enhancing a transformer architecture based on a pre-trained BERT encoder-decoder model. Recent research underscores that Transformers have become the preferred choice for model architectures. The conclusions drawn from the Tay et al. research suggest that combining pre-training and architectural

advances may be misguided, emphasizing the importance of considering both advances independently [21].

In the research conducted by Abdel-Salam and Ahmed Rafea, text summarization methods have embraced the integration of BERT with an encoder-decoder architecture for encoding both sentences and documents. The proposed encoder-decoder framework synthesizes information from both sentence embeddings and document representations [22].

BERT is employed for sentence encoding, and a Bidirectional GRU-GRU network generates scores for each sentence. Labels for each sentence are created using sentence embeddings from BERT and document representations from Bidirectional GRU, facilitating the classification of sentences as summary or non-summary.

In the realm of text summarization, research has been conducted across various languages. For instance, in the Arabic language, Abu Nada and et al. (2020) proposed an extractive Arabic text summarizer based on a general-purpose architecture for Natural Language Generation (NLG) and Natural Language Understanding (NLU), such as AraBERT, BERT, XLNet, XLM, etc. The aim was to summarize Arabic documents by evaluating and extracting the most important sentences using the Rouge measure [23].

In the research study by Alcantara et al. (2023), the focus was primarily on abstractive text summarization, which involves extracting the most important contents from a text in a rephrased form. The main objective of the project was to summarize texts in German. They concentrated on the GermanBERT multilingual model and the BART monolingual model for English, taking translation possibilities into consideration. For the experiment setup, they utilized the German Wikipedia article dataset and compared the performance of the multilingual model for German text summarization with machine-translated text summaries from monolingual English language models. The quality of text summarization was analyzed using the ROUGE-1 metric. The experimental results indicate that the monolingual BART model would be a better approach compared to the GermanBERT model for abstractive text summarization, especially when using a large dataset such as German Wikipedia [24].

One of the primary applications of the text summarization system is to access the core content of a text without the necessity of reading the entire document. This reduction in text volume leads to an increase in reading and content analysis speed, along with a decrease in storage space requirements. Based on the investigations conducted in this research, it is evident that the BERT algorithm and model exhibit superior accuracy and performance compared to other competitors, employing a more modern structure. The BERT algorithm, developed using deep learning techniques, enhances its capabilities.

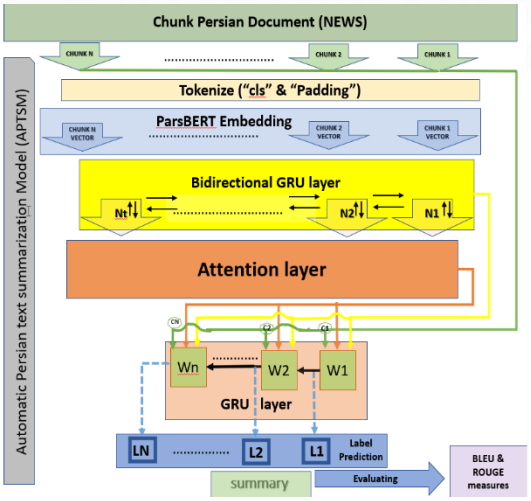

Figure 1. Architecture of proposed model.

With its two-way understanding of the text and the generation of appropriate text representations, BERT demonstrates enhanced proficiency in natural language processing. However, as mentioned earlier, a limitation of the BERT algorithm is the maximum number of input tokens, set at 512. This constraint poses challenges when dealing with larger texts. The integration of Bi_GRU with the BERT algorithm, coupled with the use of an attention layer, effectively addresses this limitation and results in improved outcomes.

III. Materials and Methods

The architectural enhancements implemented in the model aimed to address the limitations of the BERT algorithm, specifically its constraint on the number of input tokens. The improvements included the development of the ParsBERT algorithm in conjunction with Bi_GRU and the addition of an attention layer, effectively resolving the input token limitation. This not only mitigated the problem but also resulted in an overall enhancement of the ParsBERT algorithm's performance. At the outset of this section, a detailed analysis of the developed model for the Persian language and news text was conducted, focusing on each sentence in the news text. The proposed text summarization solution extends previous work that utilized Bidirectional GRU on top of ParsBERT for tasks like entity recognition, text classification, and question answering.

Our architecture integrates ParsBERT with Bi_GRU, an attention layer, and an additional GRU framework to capture the global context of the entire document, as depicted in Fig. 1. Subsequently, the steps of the model were presented, offering a sequential walkthrough of the algorithm's operations.

Each part of the model was explained in detail, elucidating its role and contribution to the overall process. This presentation aimed to provide transparency and clarity regarding the workflow of the model.

A. Entering Persian Text Initially

The Persian text is input into the model, and subsequently, the sentences are segmented. These sentences proceed to the embedding stage in the

ParsBERT algorithm, being treated as individual sentences from sentence 1 to sentence N.

For Persian text tokenization in this study, the "Hazm" library was employed. "Hazm" is a natural language processing library specifically crafted for the Persian language. This library offers diverse tools and functionalities for tasks including tokenization, word root extraction, sentence role analysis, and Persian text normalization.

B. ParsBERT Embedding

Pars BERT embeddings are representations of words generated by BERT, differing from traditional word embeddings like Word2Vec. These embeddings capture the meaning and context of words in a sentence by considering the surrounding words. ParsBERT embeddings essentially serve as vector representations of words or tokens in a given context, reflecting the ParsBERT model's understanding of language. The process of generating vectors using ParsBERT embeddings involves the following steps:

1) Tokenization: Convert the input text into tokens using the WordPiece tokenization method employed in the ParsBERT tutorial.

2) Input Formatting: Format the marked text into the input format required by ParsBERT. This typically includes adding special markers such as [CLS] (for classification tasks) and [SEP] (for separating sentences) and adjusting sequences to a fixed length.

3) Extract Embeddings: Retrieve the embeddings for the tokens of interest. These embeddings can be obtained from the final hidden layers or specific intermediate layers of the ParsBERT model.

4) Vector Representation: The obtained embeddings for each token constitute the vector representation for those tokens in the input text. Creating a single vector representation for the entire sentence can be achieved by averaging these embedded tokens.

5) Further Processing: Depending on the specific task or application, additional operations such as integration, normalization, or dimensionality reduction may be applied to the generated vectors.

C. Teaching the ParsBERT Model for Persian Language

The accuracy of the trained model significantly influences output performance. In this research, an optimal model was created and trained using the mentioned dataset for the Persian language. The stages of model training are outlined as follows:

Data Collection and Preparation: The dataset for this research consists of Farsi news articles, totaling more than 175,000, sourced from reputable news agencies such as TASNIM, ISNA, Fars, …. News Agency, and others. The inclusion of articles from diverse sources ensures a comprehensive and representative dataset for the study. This large collection is essential for robust model training and evaluation, providing ample variation in writing styles, topics, and linguistic nuances found in Farsi news. The use of established news agencies contributes to the reliability and credibility of the dataset, aligning with standard practices in text summarization research. Texts and summaries are formatted appropriately, and the dataset is divided into training and test sets. To ensure a comprehensive evaluation, we partitioned the dataset into distinct training and testing sets, allocating 80% for training and 20% for testing while maintaining a proportional distribution.

Preprocessing of Dataset Data: The dataset undergoes processing to eliminate extra characters and standardize Persian language characters. This preprocessing enhances accuracy and speeds up model training. Dealing with non-standard characters or variations in writing styles is crucial for model precision.

Tokenizer Tutorial: To train the model, text must be converted into a tokenized format. While most Transformer models come with a pre-trained Tokenizer, training a specific Tokenizer on the data is necessary when building a model from scratch. The ParsBERT Tokenizer Fast class from transformers is utilized for training a tokenizer on the data. Padding was applied to maintain consistent sequence lengths, set to a maximum of 500. The sliding window approach overcomes limitations in handling longer content, though it lacks connectivity between windows, limiting global context capture.

Model Pre-training: In the final step, the model undergoes pre-training. The pre-training phase is essential for the model to learn and adapt to the patterns and features present in the Persian language dataset. This phase is crucial for the model's ability to generate accurate and contextually relevant embeddings during the later stages of the summarization process.

The Bi_GRU layer, when applied after ParsBERT embeddings, is designed to augment the comprehension of sequential information within the rich textual representations produced by ParsBERT. This enhancement enables the model to grasp nuances and deeper dependencies present in the text.

Following the Bi_GRU layer, additional layers, such as extra GRU or LSTM layers, attention mechanisms, or dense layers, may typically be incorporated based on specific task requirements. This sequential processing empowers the model to learn intricate patterns and relationships in textual data. Such capabilities are valuable for diverse natural language processing tasks, including sentiment analysis, named entity recognition, or machine translation. In the context of the architecture described, an attention layer, positioned after a Bi_GRU layer, incorporates an attention mechanism into the neural network. This mechanism enables the model to concentrate on specific segments of the input sequence when making predictions or forming representations.

Following the bidirectional contextual information provided by the Bi_GRU layer, the attention layer assists the model in selectively attending to various segments of the sequence. It assigns varying degrees of importance or relevance to individual cues or elements within the sequence. This attention mechanism proves particularly beneficial for handling long sequences or scenarios where certain elements in the sequence are interrelated. Using attention scores, the attention layer

generates a weighted representation of the sequence by multiplying the representation of each cue by its corresponding attention score and summing these weighted representations.

D. Contextual representation

The resulting sum of weighted cue representations forms a context-weighted representation that accentuates cues considered more important or relevant by the attentional mechanism. This dynamic attention mechanism enables the model to adaptively concentrate on various segments of the input sequence, facilitating the learning process to emphasize pertinent information while disregarding noise or less crucial elements.

E. Output label prediction layers

The output label prediction layer plays a crucial role in transforming the learned representations from the GRU layer into the suitable output format for the specific task at hand. This layer is designed to convert the encoded information captured by the GRU into actionable predictions aligned with the task's objective. The architecture and configuration of this layer are adapted for sequence labeling, considering the specific nature of the task. In more intricate architectures or specialized frameworks, a summarization layer is a component or mechanism employed to consolidate or summarize the output generated by the label prediction layer or any preceding layers.

This summarization involves compressing information or forming a concise representation of the model's output. The role of the summary layer is to diminish the dimensions of the output representations or compress the information for further processing or interpretation. When working with a label prediction layer or the output of a machine learning model, it is common to evaluate the model's performance using various metrics to comprehend how well it performs on a specific task. BLEU, ROUGE-N (ROUGE-1 and ROUGE-2) and ROUGEL were employed as primary metrics for evaluating our proposed model.

IV. Results and Discussion

The provided section furnishes a thorough overview of the experimental setup, elucidating the data to be processed and the methodologies applied for subsequent evaluation. The dataset utilized in this research comprises Farsi news articles, exceeding 175,000 in total, sourced from reputable news agencies like TASNIM and ISNA News Agency. The research database included news articles categorized into three groups: political and international affairs, cultural and social commentary, and the third category of technology and market analysis [8]. 50,000 data points have been allocated to both the political and international notes categories, as well as to the technology and economic market categories. Additionally, 75,000 data points have been assigned to the sports and social-cultural category. It introduces the Chunk Vector Generation process, wherein the input is a document, and the output consists of vectors for each sentence. This process entails dividing the document into chunks, generating ParsBERT embedding vectors for each chunk, and concatenating these vectors to obtain embeddings for individual sentences. All the phases in the proposed (ATS) algorithm were implemented using Python 3 Jupyter Notebook and NVIDIA V100 GPUs.In tokenization, the input sentence is divided into smaller tokens (lexical or sub-lexical tokens). Each token represents one unit of input. This process is performed with the help of a tokenizer. The selection of sentences in tokenization means that among the tokens obtained from the algorithm, the part related to the desired sentence or sentences is chosen. Here are some tips for sentence selection in tokenization:

The beginning and end of the sentence: Sentence start and end tokens are typically assigned specific tokens. Commonly, [CLS] serves as the sentence start token, while [SEP] is used as the sentence end token.

Coordinates of tokens: To select sentences, choose tokens relevant to the desired sentence based on their ID in the tokenizer.

Attention to the order of tokens: In tokenizers, the order of tokens is crucial. It is imperative to preserve the sequential arrangement of tokens within the sentence.How Padding works in summarizing the text:

Determine the maximum length: Set a maximum length for sentences, which may be determined by the tokenizer or other criteria.

Filling with Padding: If sentences are shorter than the maximum length, padding tokens (usually represented as [PAD]) are added to reach the maximum length.

Use in the model: Sentences filled with Padding are input into the model. The model extracts information related to automatic text summarization from the real tokens and Padding. In order to develop an algorithm that fits the training data well and generalizes effectively to new data. As shown in Table 1, the performance of the hybrid model was evaluated with allocated 80% of the database for training and for validation and testing the validation set (10% of the dataset).For training, the BIGRU was fine-tuned using, and an attention layer was employed to initialize the target embedding values.

TABLE I. The number of NEWS in the training, validation, and testing datasets.

Dataset	Train Set (80%)	Val Set (10%)	Test Set (10%)
Persian text of the NEWS (175000)	140000	17500	17500

The features extracted from ParsBERT are in the form of real-numbered vectors (embeddings). These vectors encapsulate the semantic information of the text, as extracted by the ParsBERT model. In many cases, the embedding vectors are employed as input to subsequent layers of the model or as output for diverse tasks when utilizing the ParsBERT model. Table 2 provides an overview of the met parameters for simulating the ParsBERT layer.

TABLE II. Meta-parameters of ParsBert layer simulation.

Epoch Number	200

Batch Size	50
Learning Rate	0.00004
Iteration Value	40
Optimization Algorithm	Adam

Dropout is a regularization technique that prevents overfitting and enhances model performance by randomly excluding some neurons from the network probabilistically. For instance, the optimal parameters for the NEWS SUMMARY dataset include a dropout rate of 0.4 on the BIGRU units and a last GRU dropout rate of 0.2. The BiGRU layer serves as the subsequent layer, and its output vector typically includes the following characteristics:

Dimensions of the Output Vector: The number of dimensions of the output vector depends on the architecture and model settings. These vectors may be three-dimensional, generating a specific-dimensional vector for each token and each direction (forward and backward).

Activation Function**:** Activation functions, such as tanh or ReLU, can be employed to activate GRU units. These functions play a crucial role in determining the characteristics of the output features. In this case, ReLU has been used as the activation function. Meta-parameters set in this layer in Table 3.

TABLE III. BIGRU LAYER SIMULATION PARAMETERS.

Epoch Number	200
Batch Size	50
Learning Rate	0.00001
Iteration Value	40
Activation Function	ReLU

The Attention output can serve as input to subsequent layers or as the final output of the model, depending on the overall model structure and specific tasks.

The Attention layer is commonly paired with a softmax function to normalize the weights (ensuring their sum equals 1) and a transformation function (like the tanh function) to adjust the weights of the inputs. Adding the Attention layer to the model enhances focus on crucial components of the output text.

TABLE IV. HYPERPARAMETERS OF THE ATTENTION LAYER.

Number of Heads	4
Weight Dimensions	64
Dropout	0.2
Masking Strategy	padding

After implementing the attention layer in a text summarization model utilizing the GRU unit, it's essential to note that the hyperparameters of the GRU unit remain unaffected. The attention layer typically enhances the representation of the output sequence from the GRU but does not influence the internal parameters of the GRU unit. The met parameters for simulating the GRU layer are detailed in Table 5.

TABLE V. GRU LAYER SIMULATION PARAMETERS.

Epoch Number	200
Batch Size	40
Learning Rate	0.00002
Iteration Value	100
Activation Function	Adam

In Table 6 and Table 7, the evaluation results in Rouge-1, Rouge-2 are calculated and presented. This evaluation has been done on three Persian documents with the title of political, sports, and technology subject categories.

TABLE VI. COMPARISON OF THE LAST LAYER CHANGE RNN, LSTM AND GRU ON THE PROPOSED ALGORITHM(ROUGE-1)

Last layer algorithms	**Database**	**Rouge-1**		
		F1-Score	**Recall**	**Precision**
Results of scores Output by replacing RNN network instead of GRU	Persian political news document	0.79	0.91	0.75
	Persian sports news document	0.68	0.87	0.73
	Persian technology news document	0.72	0.89	0.86
Results of scores Output by replacing LSTM instead of GRU	Persian political news document	0.82	0.95	0.87
	Persian sports news document	0.71	0.88	0.83
	Persian technology news document	0.74	0.91	0.81
Results of scores Output of the developed algorithm	Persian political news document	0.81	0.96	0.88
	Persian sports news document	0.76	0.95	0.84
	Persian technology news document	0.79	0.94	0.86

TABLE VII. COMPARISON OF THE LAST LAYER CHANGE RNN, LSTM AND GRU ON THE PROPOSED ALGORITHM (ROUGE-2)

Last layer algorithms	**Database**	**Rouge-2**		
		F1-Score	**Recall**	**Precision**
Results of scores Output by replacing RNN network instead of GRU	Persian political news document	0.61	0.85	0.60
	Persian sports news document	0.61	0.84	0.74
	Persian technology	0.65	0.83	0.80

	news document			
Results of scores Output by replacing LSTM instead of GRU	Persian political news document	0.63	0.85	0.70
	Persian sports news document	0.63	0.84	0.77
	Persian technology news document	0.67	0.88	0.84
Results of scores Output of the developed algorithm	Persian political news document	0.79	0.94	0.84
	Persian sports news document	0.74	0.92	0.79
	Persian technology news document	0.76	0.93	0.82

In Table 8, the evaluation results in Rouge-L and Figure 2, the evaluation results in BLEU are calculated and presented.

TABLE VIII. COMPARISON OF THE LAST LAYER CHANGE BASED RNN, LSTM AND GRU BASED ON ROUGE-L.

Last layer algorithms	**Database**	**Rouge-L**		
		F1-Score	**Recall**	**Precision**
Results of scores Output by replacing RNN network instead of GRU	Persian political news document	0.77	0.74	0.81
	Persian sports news document	0.72	0.74	0.72
	Persian technology news document	0.79	0.82	۰,۷4
Results of scores Output by replacing LSTM instead of GRU	Persian political news document	0.80	0.89	0.85
	Persian sports news document	0.82	0.84	0.80
	Persian technology news document	0.84	0.89	0.87
Results of scores Output of the developed algorithm	Persian political news document	0.87	0.97	0.92
	Persian sports news document	0.83	0.95	0.88
	Persian technology news document	0.85	0.96	0.90

This evaluation has been done on three Persian documents with the title of political, sports, and technology subject categories.

To assess research outcomes effectively, defining criteria is crucial for comparing results across various studies, including prior research. Standardizing the output report from this platform is essential, enabling other researchers to utilize and evaluate the findings. Striking a balance in the number of evaluation criteria is crucial – while more criteria can enhance result accuracy, excessive complexity may challenge decision-making for optimal performance. In our developed model, the incorporation of the Attention network and BiGRU not only addresses Bert's limitations on the number of input tokens but also yields more favorable results in the Rouge evaluator criterion. The outcomes are presented in Table 9, showcasing a comparative analysis of scores with other research studies.

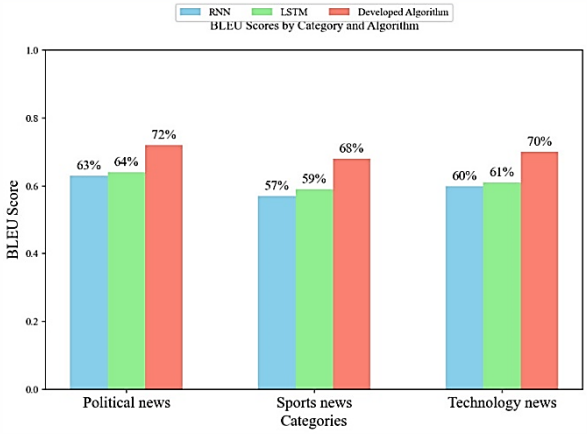

Figure 2. Comparison of the last layer change based RNN, LSTM and GRU based on BLEU.

TABLE IX. COMPARING THE BEST RESULTS OF THE ALGORITHM WITH SIMILAR DEVELOPED ALGORITHMS.

Model	**Rouge-1**	**Rouge-2**	**Rouge-L**
ParsBERT+MT5 [25]	44.01	25.07	36.76
BERT + LSTM [7]	76.25	52.49	72.72
ParsBer t+ BiGRU	79.23	76.13	85.82

The process for summarizing Persian texts is depicted in the figure 2. On the right side, a Farsi news text can be inputted, ranging from a minimum of 300 words to a maximum of 3000 words. Following the completion of the summarization process and the specification of the desired number of sentences, which should be less than 30% of the sentences in the original text, the summarization process is executed. The output of this process is constrained to a maximum of 1000 words. The figure 3 illustrates an example of scientific news text.

Figure 3. The example of scientific news text from the ISNA for summarizing Persian texts.

V. CONCLUSION

The proliferation of text documents across various online platforms presents a significant challenge in text processing, particularly in text summarization. The aim of text summarization is to condense lengthy texts while retaining essential content, ensuring readability and coherence. While many methods score and select sentences based on importance, our research introduces enhancements to address limitations in the Bert algorithm for Persian language processing. By combining ParsBert with BiGRU and an Attention network, our study not only overcomes the token input restrictions of Bert but also demonstrates improved performance compared to existing benchmarks. Our evaluation, particularly against studies by Fitrianah and Farahani, showcases substantial performance enhancements, indicating the effectiveness of our approach. They achieving the best score of 36.76% in the Rouge-L evaluator criterion. Notably, the current research demonstrates a substantial 49% performance improvement over this benchmark.

Moreover, the integration of an Attention network and the addition of a GRU layer underscore significant performance improvements, suggesting avenues for further optimization, such as replacing the GRU layer with an LSTM layer. These findings highlight the importance of enhancing both the encoder and decoder components for better summarization results. Upon completion of the summarization process, adhering to predefined criteria such as limiting the output to less than 30% of the original text and constraining it to a maximum of 1000 words, further enhancements and dataset developments are proposed. Suggestions include incorporating Transformer networks and large language models into the summarization process to capitalize on their ability to capture contextual information effectively, thereby improving the quality and effectiveness of Persian text summarization systems.

REFERENCES

[1] G. Bharathi Mohan, R. Prasanna Kumar, S. Parathasarathy, S. Aravind, K. Hanish, and G. Pavithria, "Text Summarization for Big Data Analytics: A Comprehensive Review of GPT 2 and BERT Approaches," *Data Analytics for Internet of Things Infrastructure*, pp. 247-264, 2023.

[2] H. Aliakbarpour, M. T. Manzuri, and A. M. Rahmani, "Automatic text summarization using deep reinforced model coupling contextualized word representation and attention mechanism," *Multimedia Tools and Applications*, pp. 1-30, 2023.

[3] D. Fitrianah and R. N. Jauhari, "Extractive text summarization for scientific journal articles using long short-term memory and gated recurrent units," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 1, pp. 150-157, 2022.

[4] R. Tahseen, U. Omer, M. S. Farooq, and F. Adnan, "Text summarization techniques using natural language processing: A systematic literature Review," *VFAST Transactions on Software Engineering*, vol. 9, no. 4, pp. 102-108, 2021.

[5] P. J. Uppalapati, M. Dabbiru, and K. V. Rao, "A Comprehensive Survey on Summarization Techniques," *SN Computer Science*, vol. 4, no. 5, p. 560, 2023.

[6] S. Bano, S. Khalid, N. M. Tairan, H. Shah, and H. A. Khattak, "Summarization of scholarly articles using BERT and BiGRU: Deep learning-based extractive approach," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 9, p. 101739, 2023.

[7] E. Kebriaei *et al.*, "Persian offensive language detection," *Machine Learning*, pp. 1-21, 2023.

[8] M. Yousefi. "Persian News Dataset." https://github.com/milad-4274/persian_news (accessed 1 April, 2021).

[9] S. Relan and R. Rambola, "A review on abstractive text summarization Methods," in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2022: IEEE, pp. 1-7.

[10] N. Alami, M. E. Mallahi, H. Amakdouf, and H. Qjidaa, "Hybrid method for text summarization based on statistical and semantic treatment," *Multimedia Tools and Applications*, vol. 80, pp. 19567-19600, 2021.

[11] S. Chitrakala, N. Moratanch, B. Ramya, C. Revanth Raaj, and B. Divya, "Concept-based extractive text summarization using graph modelling and weighted iterative ranking," in *emerging research in computing, information, communication and applications: ERCICA 2016*, 2018: Springer, pp. 149-160.

[12] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "EdgeSumm: Graph-based framework for automatic text summarization," *Information Processing & Management*, vol. 57, no. 6, p. 102264, 2020.

[13] R. M. Alguliyev, R. M. Aliguliyev, N. R. Isazade, A. Abdi, and N. Idris, "COSUM: Text summarization based on clustering and optimization," *Expert Systems*, vol. 36, no. 1, p. e12340, 2019.

[14] R. C. Belwal, S. Rai, and A. Gupta, "Text summarization using topic-based vector space model and semantic measure," *Information Processing & Management*, vol. 58, no. 3, p. 102536, 2021.

[15] A. Ganesh, A. Jaya, and C. Sunitha, "An Overview of Semantic Based Document Summarization in Different Languages," *ECS Transactions*, vol. 107, no. 1, p. 6007, 2022.

[16] S. Jumpathong, T. Theeramunkong, T. Supnithi, and M. Okumura, "A Performance Analysis of Deep-Learning-Based Thai News Abstractive Summarization: Word Positions and Document Length," in *2022 7th International Conference on Business and Industrial Research (ICBIR)*, 2022: IEEE, pp. 279-284.

[17] R. S. Shini and V. A. Kumar, "Recurrent neural network based text summarization techniques by word sequence generation," in *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, 2021: IEEE, pp. 1224-1229.

[18] S. Abbes, S. B. Abbès, R. Hantach, and P. Calvez, "Automatic text summarization using transformers," in *Knowledge Graphs and Semantic Web: Third Iberoamerican Conference and Second Indo-American Conference, KGSWC 2021, Kingsville, Texas, USA, November 22–24, 2021, Proceedings 3*, 2021: Springer, pp. 308-320.

[19] R. Adelia, S. Suyanto, and U. N. Wisesty, "Indonesian abstractive text summarization using bidirectional gated recurrent unit," *Procedia Computer Science*, vol. 157, pp. 581-588, 2019.

[20] X. Zhang, F. Wei, and M. Zhou, "HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization," *arXiv preprint arXiv:1905.06566*, 2019.
[21] Y. Tay *et al.*, "Are pre-trained convolutions better than pre-trained transformers?," *arXiv preprint arXiv:2105.03322*, 2021.
[22] S. Abdel-Salam and A. Rafea, "Performance study on extractive text summarization using BERT models," *Information*, vol. 13, no. 2, p. 67, 2022.
[23] A. M. Abu Nada, E. Alajrami, A. A. Al-Saqqa, and S. S. Abu-Naser, "Arabic text summarization using arabert model using extractive text summarization approach," 2020.
[24] T. H. M. Alcantara, D. Krütli, R. Ravada, and T. Hanne, "Multilingual Text Summarization for German Texts Using Transformer Models," *Information*, vol. 14, no. 6, p. 303, 2023.
[25] M. Farahani, M. Gharachorloo, and M. Manthouri, "Leveraging ParsBERT and pretrained mT5 for Persian abstractive text summarization," in *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*, 2021: IEEE, pp. 1-6.

Mohammad Rabiei was born in 1983 in Tehran, Iran. He received his B.Sc. in Computer Engineering from YAZD University in 2006 and received the M.Sc. degree at the Industrial Engineering University of Science and Technology (IUST), Iran, in March 2009 with the highest mark, by discussing a thesis concerning the "Human information literacy and e-readiness". Also, he received his Ph.D. in Information Technology in Industrial Engineering (Robotics) from Udine University, Italy in 2015 and PhD specializing in ontology in robotics from the University of Leuven, Belgium in 2016. He has been an assistant professor and faculty member at department of Computer Engineering, University of Eyvanekey since the year 2016 to the present. His research interests include Image processing, machine vision, natural language processing, machine learning, deep learning, Implementing robotics projects and industrial robots using interdisciplinary science and Implementing new business intelligence techniques in public and private organizations, Managing customer relationship and loyalty and e-commerce, Implement social networks in e-marketing and replace this advertising strategy.