



T-RFO-EADQG: A Taylor-Ruppell's Fox Optimized Encoder-Attention-Decoder Framework for Automatic Question Generation from Text Corpus

Amit Virmani ([0009-0004-8033-2538](tel:0009-0004-8033-2538))*¹
Alok Kumar ([0000-0003-1051-0672](tel:0000-0003-1051-0672))²
Vineeta Singh ([0000-0003-1084-279X](tel:0000-0003-1084-279X))³

^{1,2,3}Department of Computer Science and Engineering, School of Engineering and Technology (Formerly known as UIET), Chhatrapati Shahuji Maharaj University, Kalyanpur, Kanpur, 208024, Uttar Pradesh, India

*Corresponding Author mail id: amitvirmani@csjmu.ac.in

DOI: <https://doi.org/10.47760/ijcsmc.2026.v15i06.001>

Abstract: Question generation plays a crucial role in the area of Natural Language Processing (NLP), and it aims to generate questions automatically from the given passage. Question generation has broad applications in intelligent education scenarios, like knowledge assessment systems, writing support systems, and intelligent tutoring systems. Over the past decades, questions have been generated manually by domain experts and educators, but the process is time-consuming, labor-intensive, and difficult to scale from large text collections. Nowadays, various automated approaches are designed for question generation by minimizing human effort. Meanwhile, these approaches mainly rely on predefined templates and rules that limit their ability to generate contextual, natural, and diverse questions. This paper presents Taylor-Ruppell's Fox Optimized Encoder Attention Decoder Question generation (T-RFO-EADQG) for question generation from a text corpus. Here, a round prompting method is used to increase the contextual understanding and question generation quality. The question generation is performed using T-RFO-EADQG from the processed text, where the T-RFO-EADQG is designed by fine-tuning Encoder-Attention-Decoder-Bidirectional Long-Short-Term Memory-Bidirectional Encoder Representations from Transformers (EAD-BiLSTM-BERT) using Taylor Ruppell's Fox Optimizer (T-RFO) algorithm. Moreover, the T-RFO-EADQG attained F1-score of 98.335%, recall of 98.548%, and precision of 98.123%.

Keywords: Deep Learning, Question generation, Ruppell's Fox Optimizer, Text corpus, Natural Language Processing

I. INTRODUCTION

For centuries, teaching through questions has been regarded as a well-established practice that has been a fundamental component of education. Further, good questions facilitate meaningful dialogues and support the students in constructing their knowledge and understanding about the subject, thereby leading to long-term retention and deep comprehension rather than rote memorization. Hence, the quality of questions plays a significant role in enhancing the efficacy of teaching. Effectively designed questions facilitate meaningful discussions, offer students new perspectives, and promote a thorough exploration of the reading topic. Meanwhile, inefficiently designed questions negatively impact learning by limiting their creative thinking, reducing their confidence, and creating confusion [1]. Question generation has been developed as a promising application of modern Large Language Model (LLM), which benefits from generative abilities, few-shot learning strategies, and contextual understanding [2]. Automatic question generation approaches are developed for addressing the challenges that are faced by test developers while generating a huge number of high-quality questions. This technology is employed for decreasing the effort and cost of question construction, thereby enabling teachers to focus on other significant learning and teaching tasks [3] [4]. Besides, this technology effectually addresses the challenges encountered by conventional manual question design, such as suboptimal outcomes and slow speed. Moreover, it also aids in facilitating deeper learning and promoting student autonomy in reading comprehension. Since the incorporation of Artificial Intelligence (AI) technology in the educational field, automatic question generation has been developed as a promising technology [1].

In the NLP domain, question generation for reading comprehension is regarded as a difficult task, and it aims to create questions automatically based on the specified reading passage. Moreover, this technology has been widely exploited in numerous significant educational applications, namely learning assessment systems, writing support systems, intellectual tutoring systems, and so on [5]. Nevertheless, providing timely feedback and high-quality question generation consumes more time and is a demanding task for educators, especially in distinct and large-scale environments [2]. In addition, substantial outcomes and advances have been attained in automatic question generation research because of the fast improvement of deep learning and NLP technologies [5]. Furthermore, recent advancements in deep learning schemes have strengthened several fields of AI. In question generation and NLP fields, numerous advancements have been made due to the development of neural transformer-based techniques, like deep bidirectional transformers (BERT). Moreover, these techniques are tuned employing next sentence prediction and masked language modelling, which enables them to learn high text representations by contemplating both the right and left sides of the sentences [6]. Even though question answering supports advanced intelligent assistants and has been regarded as a significant research area in NLP, question generation has become more popular and gathered more attention in recent days [18]. Further, this increased attention is mainly due to a large number of BERT-based techniques that are tuned on a huge number of texts, thereby generating high-quality questions on diverse topics [7].

In this paper, the T-RFO-EADQG is developed for the generation of questions from the text corpus. At first, the input text document is accumulated, and is subjected to a round prompting technique for increasing the contextual understanding and quality of question generation. Then, the processed text is applied to T-RFO-EADQG for the generation of questions. Here, the T-RFO-EADQG is developed by training the parameters of EAD-BiLSTM-BERT by utilizing T-RFO to improve the generation performance.

The primary research contribution is summarized below,

- **Designed T-RFO-EADQG for question generation:** The T-RFO-EADQG is designed for generating texts from the text corpus. The T-RFO-EADQG is developed by training EAD-BiLSTM-BERT by utilizing the T-RFO algorithm, where the T-RFO is the incorporation of Ruppel's Fox Optimizer (RFO) and Taylor series.

The paper is arranged as: a comprehensive survey of prevailing schemes is explicated in Section 2. In Sections 3 and 4, an overview of the question answering system and T-RFO-EADQG, respectively, is demonstrated. Section 5 details the experimental results and discussions performed, and Section 6 portrays conclusion of paper with outlines for future work.

II. MOTIVATION

Question generation is the automatic generation of meaningful, relevant, and reasonable questions from the text document. Different techniques have been utilized for automatically generating questions for reducing human efforts, but these methods rely on predefined templates and rules, which limit the ability to generate relevant questions. Hence, it encourages to develop a hybrid model for question generation.

A. Literature Review

Wang, L., *et al.* [1] established a prompt based Automatic Question Generation technology to generate questions. This approach consumed less time, particularly for larger databases, and decreased human bias in question framing. Still, this approach did not expand the database size for validating the prompt pattern's efficacy and generalization across diverse qualitative assessments. A method using Knowledge Augmented Question Generation (KAQG) was introduced by Chen, C.H. and Shiu, M.F., [8] for generating questions. This

model effectively enhanced the reliability, decreased manual workload, and redundancy. However, this technique did not attain better generalization, and also failed to incorporate error patterns and response times for deepening the insights into learner behaviour and item quality. A Hierarchical Question Generation and Learned Importance Integration (HILI) was developed by Zhang, X., *et al.* [9] for question generation. This scheme attained high robustness and maximized the system's capacity in handling intricate real-time misinformation effectively. Nevertheless, this model did not integrate additional modalities or knowledge graphs for enriching the verification process. Badenes, C. and Eyzaguirre, P., [2] designed QuerIA for adaptive question generation. This technique was cost-effective, attained high scalability in different academic contexts, and ensured data privacy. Nonetheless, this scheme had imbalance issues, which limited the generalization, and also demanded more resources. Difficulty Labeling and Controllable Question Generation QG (DL-CQG) was devised by Dong, X. and Li, Z., [5] to generate questions. This approach addressed the lack of annotated databases and unified quantification standards, which enhanced the applicability and quality of generated questions even under low-resource settings. Meanwhile, this method mostly relied on complex knowledge graphs, thereby made the system difficult to maintain and apply in real-time environments.

B. Challenges

The limitations of baseline techniques are enumerated as follows,

- The DL-CQG model designed in [5] utilized a low-cost and unified approach to generate questions with explicitly controllable difficulty levels. However, this model failed to enable more mathematically guided and precise optimization that limited the convergence behavior and overall solution quality in complex optimization issues.
- In [8], the KAQG was designed to ensure that generated items are statistically aligned with difficulty and discrimination parameters. Meanwhile, this technique failed to consistently produce diverse, accurate, as well as contextually relevant questions with strong performance.
- The HILI framework [9] utilized Natural Language Inference (NLI) for improving decision quality and importance scores, but this approach failed to increase training efficiency, specifically in complex models.
- Automatic Question Generation encountered various challenges in maintaining factual accuracy, diversity control, content selection, reasoning ability, and semantic understanding from the source text. It is further complicated by problems, like domain adaptation, coreference resolution, ambiguity, and generating fluent, non-redundant questions.

III. SYSTEM MODEL

Question generation is the task performed for the creation of meaningful questions automatically from the collected text document using NLP methods. Question generation mainly involves analyzing a text document, identifying important information, and generating relevant questions concerning entities, concepts, or key facts. The input text document is applied to the question generation system for the generation of questions. The generated questions are further applied to the answer generation system to generate accurate answers. Fig. 1 depicts the overview of question generation model.

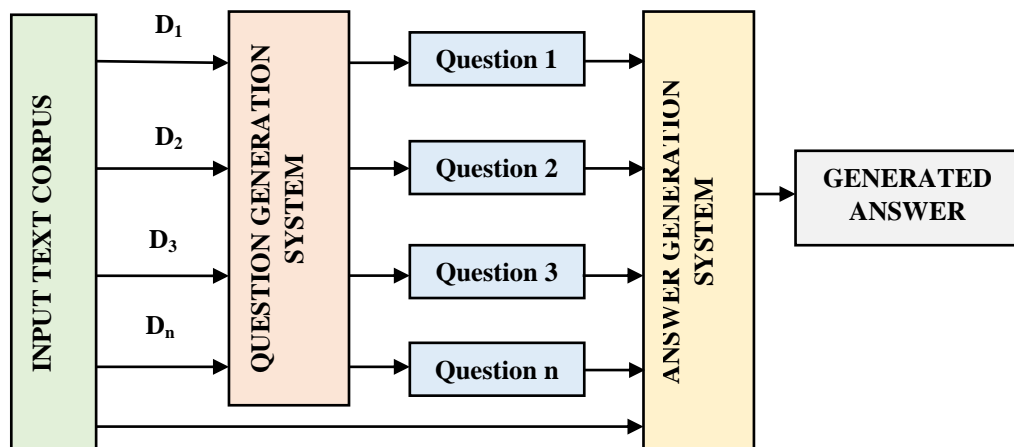


Fig. 1. Overview of question generation model

IV. DESIGNED TAYLOR-RUPPELL'S FOX OPTIMIZED ENCODER ATTENTION DECODER QUESTION GENERATION FOR QUESTION GENERATION

This paper presents T-RFO-EADQG for question generation from a text document. The input text document is initially acquired from the dataset [10] [11]. The input text document is further applied for enhancing contextual understanding and improving question generation quality using a round prompting technique [12]. The processed text is applied to T-RFO-EADQG for the generation of questions. Here, the T-RFO-EADQG is designed by training the parameters of EAD-BiLSTM-BERT [13] by utilizing the T-RFO algorithm. The T-RFO is the incorporation of RFO [14] and Taylor series [15]. The T-RFO-EADQG effectively analyzes the contextual and semantic information available in the text corpus and generates coherent, relevant, and context-aware questions. In fig. 2, the workflow of T-RFO-EADQG designed for question generation is depicted.

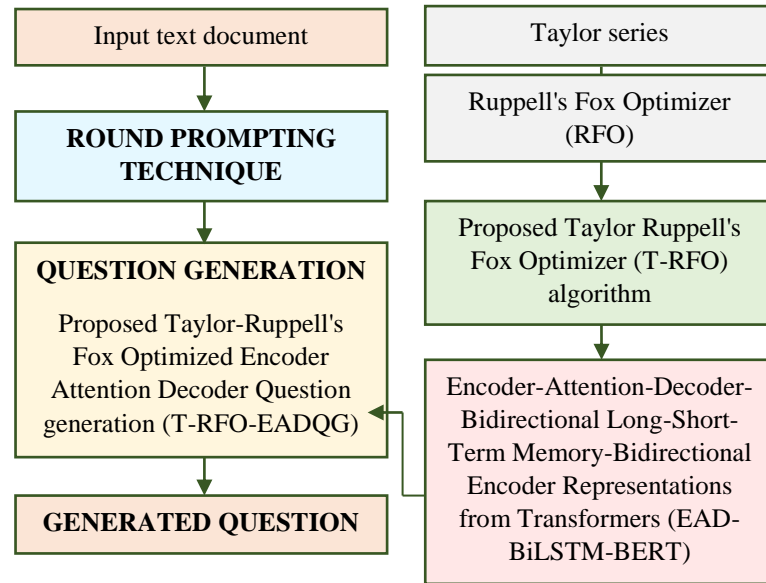


Fig. 2. Workflow of T-RFO-EADQG designed for question generation

A. Text document acquisition

The input text document is accumulated initially from the XQuAD database [10] and the MultiLingual Question Answering (MLQA) database [11], and the database is articulated as,

$$K = [K_1, K_2, K_3, \dots, K_p, \dots, K_w] \quad (1)$$

In the expression, W specifies the total text in the database, and the P^{th} accumulated text document is given as K_p .

B. Round prompting

Prompting is the commonly utilized model for the generation of targeted questions automatically from the source text. Here, the round prompting technique [12] is used for increasing the contextual understanding and improving the quality of question generation by meeting the specific knowledge points and target cognitive level. Round prompting is a type of prompting that provides multiple iterative rounds of interaction rather than providing single prompts. Initially, the method is promoted in the first round for generating questions that focus on the desired cognitive level. The specific knowledge point is focused on by the model in the second round. Following this, the outputs from the specific knowledge points and target cognitive level are combined by utilizing the context from earlier rounds. Finally, a few-shot prompting is performed in the final round of the prompt to increase the accuracy of generated questions. Thus, the generated prompt P_c is obtained from the input text document K_p from the final round.

C. Question generation using T-RFO-EADQG

The process of generating meaningful questions automatically from the text document is termed question generation. The relevant questions are generated, the important information is identified, and the text document is analyzed to generate contextual and natural questions accurately. In this research, the question is accurately generated from the generated prompt P_c using T-RFO-EADQG. Here, the T-RFO-EADQG is designed by utilizing the T-RFO algorithm to fine-tune the parameters of EAD-BiLSTM-BERT [13]. The T-RFO is designed

by integrating RFO [14] and the Taylor series [15]. The question generation process performed is explicated below,

1) *Architecture of EAD-BiLSTM-BERT*: EAD-BiLSTM-BERT [13] is a novel deep learning approach designed by integrating the architecture of Encoder-Attention-Decoder (EAD), Bidirectional Encoder Representations from Transformers (BERT), and Bidirectional Long Short-Term Memory (BiLSTM) to increase text understanding from the generated prompt P_G . Here, the BiLSTM is integrated into both the decoder and encoder along with an attention mechanism. The EAD-BiLSTM-BERT learns essential word weights and captures the discriminative features more effectively for the generation of accurate questions. The EAD-BiLSTM-BERT mainly comprises four layers, such as the Bi-LSTM decoder layer, the Bi-LSTM encoder layer, the attention layer, and the embedding layer. The input text is converted into a dense vector representation in the embedding layer. The embedding layer facilitates learning by leveraging knowledge from the unlabelled database, which also supports generalization during question generation. The Bi-LSTM encoder layer processes these embeddings in both backward and forward directions and records the contextual information. The Bi-LSTM encoder layer process the output from embedding layer for the extraction of meaningful representations, and this representation is further fed into the attention layer. The attended representation is taken by the Bi-LSTM in the decoder layer, and is further processed for the generation of the question. The contextual information is captured from the input sequence by the encoder within the Bi-LSTM layer. Each sequence is presented from left to right, and a backward layer is used to show the representation of the sequence in reverse order. Further, the outputs of the backward and forward layers are merged to form an ultimate hidden state and record the contextual information. Further, the encoder output is applied to the attention layer by assigning weights to the hidden states of the encoder. A scoring function is used for computing an attention weight for each representation in the attention mechanism. Here, a softmax function is used to get attention weights, and the context vector is obtained by taking the weighted sum of the encoded representations. The relevant information is captured by the context vector from the encoded representations. In the decoder, a Bi-LSTM layer is used for processing the output of attention mechanism. Following this, the hidden states are generated in the decoder Bi-LSTM, and the forward as well as backward passes are determined. Further, the previously generated tokens and the context vector are applied to the Bi-LSTM decoder layer for producing one token at a time. The probability of the whole sequence is estimated by multiplying the conditional probabilities of each word from the context vector and previous words. Thus, the generated question G_q is obtained by the model from the generated prompt P_G , and in fig. 3, the core design of EAD-BiLSTM-BERT is displayed.

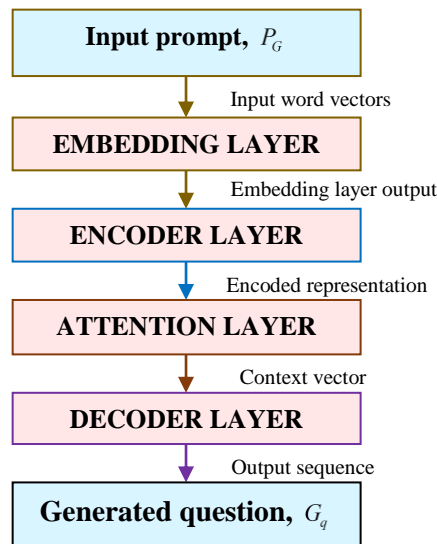


Fig. 3. Core design of EAD-BiLSTM-BERT

2) *Hybrid T-RFO for training EAD-BiLSTM-BERT*: The performance of EAD-BiLSTM-BERT in question generation is enhanced by fine-tuning the parameters by utilizing the T-RFO algorithm. Here, the T-RFO is designed by incorporating RFO [14] and the Taylor series [15]. RFO is a metaheuristic algorithm designed concerning the collective and adaptive characteristics of Ruppell's foxes. The Ruppell's fox adjusts its diet based on the seasonal food availability, and also thrives a wide range of environments, and exhibits remarkable flexibility. The algorithm is created by mimicking the social nature, opportunistic foraging strategies, as well as the ability of Ruppell's fox to work together in challenging situations in solving various issues. Similarly, the Taylor series is used for the simple local approximation of complex functions. The

incorporation of Taylor series with RFO significantly increases the search efficiency and stabilizes the search process. The T-RFO is mathematically formulated as follows,

a) Initialization: The solutions are initially generated randomly in the search space, and these solutions are expressed by,

$$H = \begin{bmatrix} H_1^1 & H_2^1 & \dots & H_l^1 \\ H_1^2 & H_2^2 & \dots & H_l^2 \\ \vdots & \vdots & \ddots & \vdots \\ H_1^u & H_2^u & \dots & H_l^u \end{bmatrix} \quad (2)$$

where, the position of k^{th} Ruppell's foxes is signified as H_k^i , the number of decision variables is indicated as l , and the population size is indicated by u . Further, the initial population is given by,

$$H_k^t = \delta_t + Rand(\eta_t - \delta_t), \quad k=1,2,3,\dots,W \quad t=1,2,3,\dots,V \quad (3)$$

In the expression, a uniformly distributed random number in the range $[0,1]$ is specified as $Rand$, the upper as well as lower limits of the decision variable are denoted by δ_t and η_t , and H_k^t symbolizes the value of the k^{th} decision variable.

b) Fitness computation: The best candidate solution is determined using the Mean Square Error (MSE), where the MSE is designated by,

$$Fitness\ function = \frac{1}{I} \sum_{d=1}^I \left[(G_q^*) - (G_q)_d \right]^2 \quad (4)$$

wherein, the number of training samples is given by I , and G_q^* is the targeted output of EAD-BiLSTM-BERT.

c) Prey searching in daylight: The Ruppell's foxes perform local exploitation and exploration for updating their position by utilizing ear and eye rotation scanning, hearing-based navigation, and sighting-based navigation. The Ruppell's foxes search for prey based on sight under the condition $R \geq 0.25, (U - 0.5) \geq 0$, and $J \geq g$, where the updated Ruppell's fox position is expressed as,

$$H_k(S+1) = H_k(S) + w_{index} (H_{Rand} - H_k(S)) + w_{index} (F_{best_k}(S) - H_k(S)) \quad (5)$$

Here, the random number distributed uniformly in the interval $[0,1]$ is given by R , a random variable set to $[0,1]$ is denoted by U , the hearing and sight sense of Ruppell's fox are given by g and J , S denotes iteration, the best global position is symbolized as $F_{best_k}(S)$, the randomly selected particle position is indicated as H_{Rand} , and w_{index} is the random coefficient. Moreover, the food sources are accurately detected by the Ruppell's fox when the view is 260° by utilizing its rational mobility. When $R < 0.25, (U - 0.5) \geq 0$, and $J \geq g$, the updated position of Ruppell's fox is designated as,

$$H_k(S+1) = H_{Rotate_k}(S) + \alpha \times Rn(1,l) \times \beta \quad (6)$$

where, β specifies flag, the step size of boundaries is given by α , the Ruppell's fox position after rotation is symbolized as $H_{Rotate_k}(S)$, and a l -dimensional vector with uniformly distributed random values among $[0,1]$ is indicated as $Rn(1,l)$. In order to locate the prey in daylight, both visual capabilities, turning feature of sight, and strength are utilized by Ruppell's fox. Also, it utilizes the hearing sense for location prey. When $R \geq 0.75, (U - 0.5) \geq 0$, and $J \leq g$, the updated Ruppell's fox position is articulated in equation (5). The rotational mobility of the ears are utilized by Ruppell's fox for increasing the localization of the prey. This helps to identify prey within 150° along the circular trajectory centered near the best candidate position. If $R < 0.75, (U - 0.5) \geq 0$, and $J < g$, the modified Ruppell's fox position is given in equation (6).

d) Prey searching in night: The Ruppell's fox mainly rely on their hearing sense other than vision for locating prey. Thus, the modified position of Ruppell's fox, if $R \geq 0.75, (U - 0.5) < 0$, and $J \geq g$ is given as,

$$H_k(S+1) = H_k(S) + w_{index} (C_{Rand} - H_k(S)) + w_{index} (F_{best_k}(S) - H_k(S)) \quad (7)$$

Here, C_{Rand} specifies a randomly selected particle position. Due to reduced visual ability, the Ruppell's fox relies mainly on hearing sense. The Ruppell's fox may miss the prey even with strong sensory skills, which leads then to perform random foraging in the search space. When $R < 0.25, (U - 0.5) < 0$, and $J \leq g$, the process is given in equation (6). Moreover, the ears are rotated by Ruppell's fox upto 150° for detecting the sounds from potential prey. Thus, the chances of sensing prey are increased by the unique ability in multiple directions near the best position. The prey search based on the sighting sense at night when $R \geq 0.75, (U - 0.5) < 0$, and $J > g$ is articulated in equation (5). Likewise, the Ruppell's fox rotates its eyes upto 260° for improving its ability in

detecting prey from different locations and directions. This behaviour is modelled mathematically $R < 0.75$, $(U - 0.5) < 0$, and $J > g$ when is given in equation (6).

e) Locating prey based on smell feature: The Ruppell’s fox relies on exceptional smelling sense for locating prey on both night and day, where the expression is mathematically modelled as,

$$H_k(S+1) = \begin{cases} H_k(S) + U \times (H_{Rand(1)} - H_k(S)) \times q_2 + U \times (F_{best_k}(S) - H_k(S)) \times q_3 & q_1 \geq smell \\ H_{Rand(1)} + \alpha \times Rn(1,l) & q_1 < smell \end{cases} \quad (8)$$

Here, H_{Rand} specifies a randomly selected particle position, the random values fixed to $[0,1]$ is denoted by q_1 , q_2 , and q_3 .

Let us consider $q_1 \geq smell$,

$$H_k(S+1) = H_k(S) + U \times (H_{Rand(1)} - H_k(S)) \times q_2 + U \times (F_{best_k}(S) - H_k(S)) \times q_3 \quad (9)$$

$$H_k(S+1) = H_k(S) + U \times H_{Rand(1)} q_2 - U \times H_k(S) q_2 + U \times F_{best_k}(S) q_3 - U \times H_k(S) q_3 \quad (10)$$

$$H_k(S+1) = H_k(S) [1 - Uq_2 - Uq_3] + U \times H_{Rand(1)} q_2 + U \times F_{best_k}(S) q_3 \quad (11)$$

The Taylor series [15] is integrated with RFO [14] to stabilize the search process, and increase the search efficiency. From Taylor series,

$$H(S+1) = H(S) + \frac{H'(S)}{1!} + \frac{H''(S)}{2!} \quad (12)$$

$$H'(S) = \frac{H(S) - H(S-z)}{z} \quad (13)$$

$$H''(S) = \frac{H(S) - 2H(S-z) + H(S-2z)}{z^2} \quad (14)$$

Substituting $z = 1$ in equation (14),

$$H(S+1) = H(S) + H(S) - H(S-1) + \frac{H(S) - 2H(S-1) + H(S-2)}{2} \quad (15)$$

$$H(S+1) = H(S) \left[1 + 1 + \frac{1}{2} \right] - H(S-1) - \frac{2H(S-1)}{2} + \frac{H(S-2)}{2} \quad (16)$$

$$H_k(S) = \frac{2}{5} \left[H_k(S+1) + 2H_k(S-1) - \frac{H_k(S-2)}{2} \right] \quad (17)$$

Substituting equation (17) in equation (11),

$$H_k(S+1) = \frac{2}{5} \left[H_k(S+1) + 2H_k(S-1) - \frac{H_k(S-2)}{2} \right] [1 - Uq_2 - Uq_3] + U \times H_{Rand(1)} q_2 + U \times F_{best_k}(S) q_3 \quad (18)$$

$$5H_k(S+1) - 2H_k(S+1)(1 - Uq_2 - Uq_3) = \frac{2}{5} \left[2H_k(S-1) - \frac{H_k(S-2)}{2} \right] [1 - Uq_2 - Uq_3] + \quad (19)$$

$$U \times H_{Rand(1)} q_2 + U \times F_{best_k}(S) q_3$$

Thus, the updated expression of T-RFO is given as,

$$H_k(S+1) = \frac{5}{3 + 2(Uq_2 + Uq_3)} \left[\frac{2}{5} \left(2H_k(S-1) - \frac{H_k(S-2)}{2} \right) [1 - Uq_2 - Uq_3] + U \times H_{Rand(1)} q_2 + U \times F_{best_k}(S) q_3 \right] \quad (20)$$

f) Movement towards optimal Ruppell’s Fox: The positions are adjusted constantly by Ruppell’s fox during the searching for potential prey. The Ruppell’s fox move towards detected prey in respond to scent, visual, or auditory based cues. Here, retaining of optimal position is performed and the others modify the positions. Under certain cases, if the prey move, the Ruppell’s fox continue wandering and reach the location of prey. Thus, the movement of Ruppell’s fox towards optimal solution is designated as,

$$H_k(S+1) = \begin{cases} H_k(S) + M_0 \times (H_{Rand} - H_k(S)) \times q_1 + M_1 \times (F_{best_k}(S) - H_k(S)) \times q_2 & R \geq 0.1 \\ H_{k_{op}}(S) + H_k(S) & R < 0.1 \end{cases} \quad (21)$$

where, $H_{k_{op}}(S)$ indicates modified Ruppell’s fox position based on the position of prey, the positive constants are indicates as M_0 and M_1 .

g) Animal characteristics in worst case: The Ruppell's fox increase their search to both faraway and near location when it is unable to identify the prey in the nearby regions to advance towards prospective regions. Thus, more profound investigation is performed, and is articulated as,

$$H_k^{worst}(S+1) = H_k^{worst}(S) + \alpha \times Rn(1,l), \quad \beta = 1 \quad (22)$$

h) Fitness re-estimation: The ideal solution is identified by utilizing equation (4), where if any solutions determined superior to existing solution, the determined solution can be replaced.

i) Termination: The procedure of T-RFO is followed continuously till obtaining the best candidate solution.

Thus, the T-RFO-EADQG analyzes the contextual and semantic information available in the text corpus more effectively and generates coherent, relevant, and context-aware questions.

V. RESULTS AND DISCUSSION

A detailed analysis of the efficiency of T-RFO-EADQG in question generation, in contrast to the various existing methods is presented in this section.

A. Experimental setup

The devised T-RFO-EADQG for question generation is implemented by utilizing Python 3.9.11. and in table 1, the parameters of T-RFO and EAD-BiLSTM-BERT are portrayed.

TABLE 1. EXPERIMENTAL PARAMETERS OF T-RFO AND EAD-BiLSTM-BERT

EAD-BiLSTM-BERT		T-RFO	
Parameter	Range	Parameter	Range
Epoch	150	Maximum Iteration	100
Optimizer	T-RFO	Decision variable	30
Dropout	0.2	Population size	50
Batch size	16	B (step size)	0.000001
Activation	softmax	r1, r2	random numbers between (0,1)
Loss	Categorical cross entropy		
kernel size	(3,3)		

B. Dataset description

The input text considered in this work is taken from MLQA [11] and XQuAD Dataset [10], and are described as follows

MLQA dataset: The dataset [11] comprises a total of 5,000 extractive question-answer pairs across seven languages, such as German, Hindi, Arabic, Spanish, English, Vietnamese, and Simplified Chinese.

XQuAD Dataset: This [10] dataset contains 240 paragraphs with 1,190 question-answer pairs attained from SQuAD v1.1 development set. This text data in the dataset is translated into multiple languages professionally, and the database provides fully parallel data in eleven languages.

C. Evaluation measures

The T-RFO-EADQG is analysed for its performance using three key metrics, which are detailed below.

i) F-score: The harmonic mean of recall and precision is measured using F1-score, and is computed as follows,

$$F1 \text{ score} = \frac{2.(X \times Y)}{X + Y} \quad (23)$$

where, X denotes precision and Y is recall.

ii) Recall: Recall is used to measure the proportion of accurately generated relevant questions out of the relevant references and is expressed as,

$$Y = \frac{Z_{tp}}{Z_{tp} + Z_{fn}} \quad (24)$$

where, Z_{fn} signifies false negative and Z_{fp} is false positive.

iii) Precision: This metric is utilized for identifying the proportion of accurately generated questions out of all the questions generated and is given by,

$$X = \frac{Z_{tp}}{Z_{tp} + Z_{fp}} \quad (25)$$

wherein, Z_{tp} denotes true positive.

D. Experimental results

The input text document and the corresponding questions generated by the T-RFO-EADQG, considering the two databases, are depicted in fig. 4. The input samples text from the MLQA dataset and the questions generated is given in Fig. 4a), while fig. 4b) depicts the corresponding results from the XQUAD dataset.

Passage	"Standard Oil's pre-history began in 1863 as an Ohio partnership formed by industrialist John D. Rockefeller, his brother William Rockefeller, Henry Flagler, chemist Samuel Andrews, silent partner Stephen V. Harkness, and Oliver Burr Jennings, who had married the sister of William Rockefeller's wife. In 1870, Rockefeller abolished the partnership and incorporated Standard Oil in Ohio. Of the initial 10,000 shares, John D. Rockefeller received 2,667; Harkness received 1,334; William Rockefeller, Flagler, and Andrews received 1,333 each; Jennings received 1,000, and the firm of Rockefeller, Andrews & Flagler received 1,000. Rockefeller chose the \"Standard Oil\" name as a symbol of the reliable <u>standards</u> of quality and service that he envisioned for the nascent oil industry."
Questions	"How did Standard Oil begin?" How many total shares were there initially? Who received the largest number of shares? Why was it called "Standard Oil"?"
a)	
Passage	Nearby, in Ogr\u00f3d Saski (the Saxon Garden), the Summer Theatre was in operation from 1870 to 1939, and in the inter-war period, the theatre complex also included Momus, Warsaw's first literary cabaret, and Leon Schiller's musical theatre Melodram. The Wojciech Bogus\u0142awski Theatre (1922\u20131932), was the best example of \"Polish monumental theatre\". From the mid-1930s, the Great Theatre building housed the Upati Institute of Dramatic Arts \u2013 the first state-run academy of dramatic art, with an acting department and a stage directing department."
Questions	"What is the Saxon Garden in Polish?" "Where was the Summer Theatre located?" "How long was the Summer Theatre in operation?" "What was Warsaw's first literary cabaret?" "What theatre was the best example of \"Polish monumental theatre\"?"
b)	

Fig. 4. Experimental results of T-RFO-EADQG, a) MLQA, and b) XQUAD

E. Comparative analysis

The proposed T-RFO-EADQG is compared for its efficiency by comparing it with various frameworks, such as DL-CQG [5], KAQG [8], HILI [9], QuerIA [2], Exponential-Driven Aphids Algorithm_ RoBERTa - BiLSTM (ExAoA_RoBERTa-BiLSTM), Caviar Addax Optimization Algorithm_BiLSTM (CAOA_BERT-BiLSTM), Hiking Wolf Bird Optimization with BERT-BiLSTM (HkWBO with BERT-BiLSTM), Taylor-aware Query-RAG Dual View Network with LLM (TQ-RAGNetLLM), and Autoregressive Secant Optimization-based Retrieval-Augmented Generation Dual View Network (AuRSO-RAGNet)sss. Further, the analysis is carried out considering the MLQA and XQUAD datasets.

1) *For XQUAD dataset:* Fig. 5 delineates the comparative assessment of T-RFO-EADQG considering XQUAD database, varying K-group. In fig. 5(a), the evaluation of T-RFO-EADQG by utilizing F1-score is depicted, where the T-RFO-EADQG observed an F1-score of 98.272% for K-group 8. Likewise, the F1-score measured by previous techniques is 85.045% by DL-CQG, 88.535% by KAQG, 90.988% by HILI, 92.445% by QuerIA, 97.313% by TQ-RAGNetLLM, 96.514% by ExAoA_RoBERTa-BiLSTM, 96.025% by CAOA_BERT-BiLSTM, 94.984% by HkWBO with BERT-BiLSTM, and 98.037% by AuRSO-RAGNet. Compared with HILI, the T-RFO-EADQG achieved a high performance of 7.41% in question generation. Fig. 5(b) illustrates the investigation of T-RFO-EADQG using recall. The recall of 98.547% is measured by T-RFO-EADQG for K-group 8, whereas the recall recorded by baseline models, namely DL-CQG is 85.717%, KAQG is 89.718%, HILI is 91.715%, QuerIA is 92.615%, TQ-RAGNetLLM is 97.508%, ExAoA_RoBERTa-BiLSTM is 97.018%, CAOA_BERT-BiLSTM is 96.516%, HkWBO with BERT-BiLSTM is 95.707%, and AuRSO-RAGNet is

98.348%. The T-RFO-EADQG observed a high performance of 6.02% comparing with QuerIA. Furthermore, the validation of T-RFO-EADQG utilizing precision is depicted in fig. 5(c). The precision observed by previous approaches, such as DL-CQG, KAQG, HILI, QuerIA, HkWBO with BERT-BiLSTM, CAO_A_BERT-BiLSTM, ExAoS_RoBERTa-BiLSTM, TQ-RAGNetLLM, and AuRSO-RAGNet, is 84.383%, 87.383%, 87.383%, 90.273%, 92.275%, 94.273%, 95.539%, 96.016%, 97.119%, and 97.729%. Meanwhile, the highest precision of 97.999% is yielded by T-RFO-EADQG for K-group 8. The T-RFO-EADQG achieved maximum performance of 2.51% comparing with CAO_A_BERT-BiLSTM.

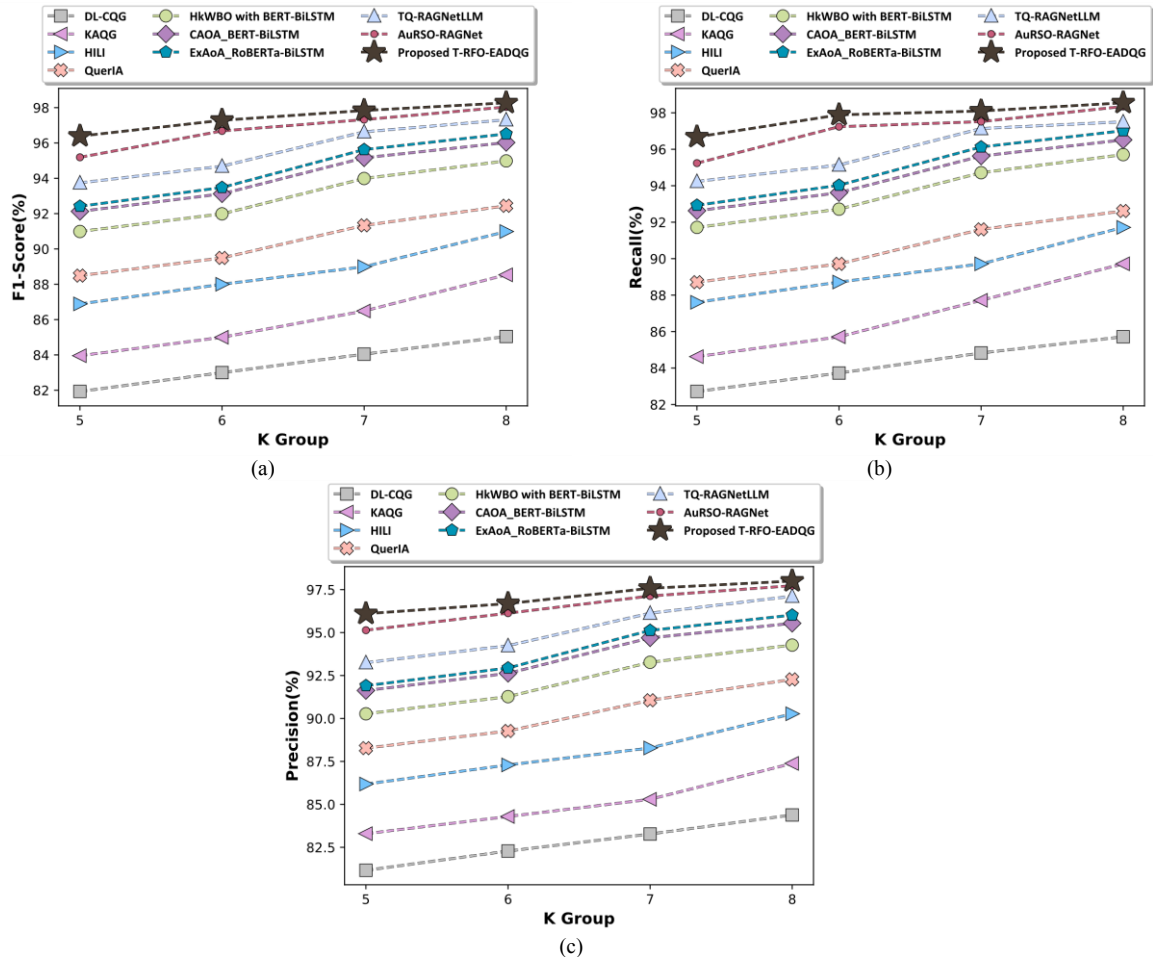


Fig. 5. Investigation of T-RFO-EADQG based on XQUAD database employing (a) F1-score, (b) Recall, (c) Precision

2) *For MLQA database:* The validation of T-RFO-EADQG concerning the MLQA dataset varying K-group is displayed in fig. 6. The investigation of T-RFO-EADQG by utilizing F1-score is depicted in fig. 6(a). The F1-score of 98.335% is attained by existing schemes, like DL-CQG, KAQG, HILI, QuerIA, HkWBO with BERT-BiLSTM, CAO_A_BERT-BiLSTM, ExAoS_RoBERTa-BiLSTM, TQ-RAGNetLLM, and AuRSO-RAGNet is 86.631%, 88.639%, 90.705%, 92.349%, 94.475%, 95.434%, 96.468%, 96.938%, and 98.043%. The T-RFO-EADQG observed the highest F1-score of 98.335% for K-group 8. The superior performance of 3.93% is observed by T-RFO-EADQG compared to HkWBO with BERT-BiLSTM. In fig. 6(b), the analysis of T-RFO-EADQG by utilizing recall is presented, where the T-RFO-EADQG observed a recall of 98.548% for K-group 8. Likewise, the recall measured by previous techniques is 87.694% by DL-CQG, 89.594% by KAQG, 90.704% by HILI, 93.004% by QuerIA, 95.372% by HkWBO with BERT-BiLSTM, 96.405% by CAO_A_BERT-BiLSTM, 97.027% by ExAoS_RoBERTa-BiLSTM, 97.259% by TQ-RAGNetLLM, and 98.248% by AuRSO-RAGNet. The T-RFO-EADQG achieved a high performance of 7.96% as comparing with KAQG. Moreover, fig. 6(c) illustrates the investigation of T-RFO-EADQG employing precision. The precision of 98.123% is measured by T-RFO-EADQG for K-group 8, whereas the precision recorded by baseline models, namely DL-CQG is 85.595%, KAQG is 87.704%, HILI is 90.707%, QuerIA is 91.704%, TQ-RAGNetLLM is 96.619%, ExAoS_RoBERTa-BiLSTM is 95.916%, CAO_A_BERT-BiLSTM is 94.483%, HkWBO with BERT-BiLSTM is 93.595%, and AuRSO-RAGNet is 97.840%. The T-RFO-EADQG observed a high performance of 12.77% comparing with DL-CQG.

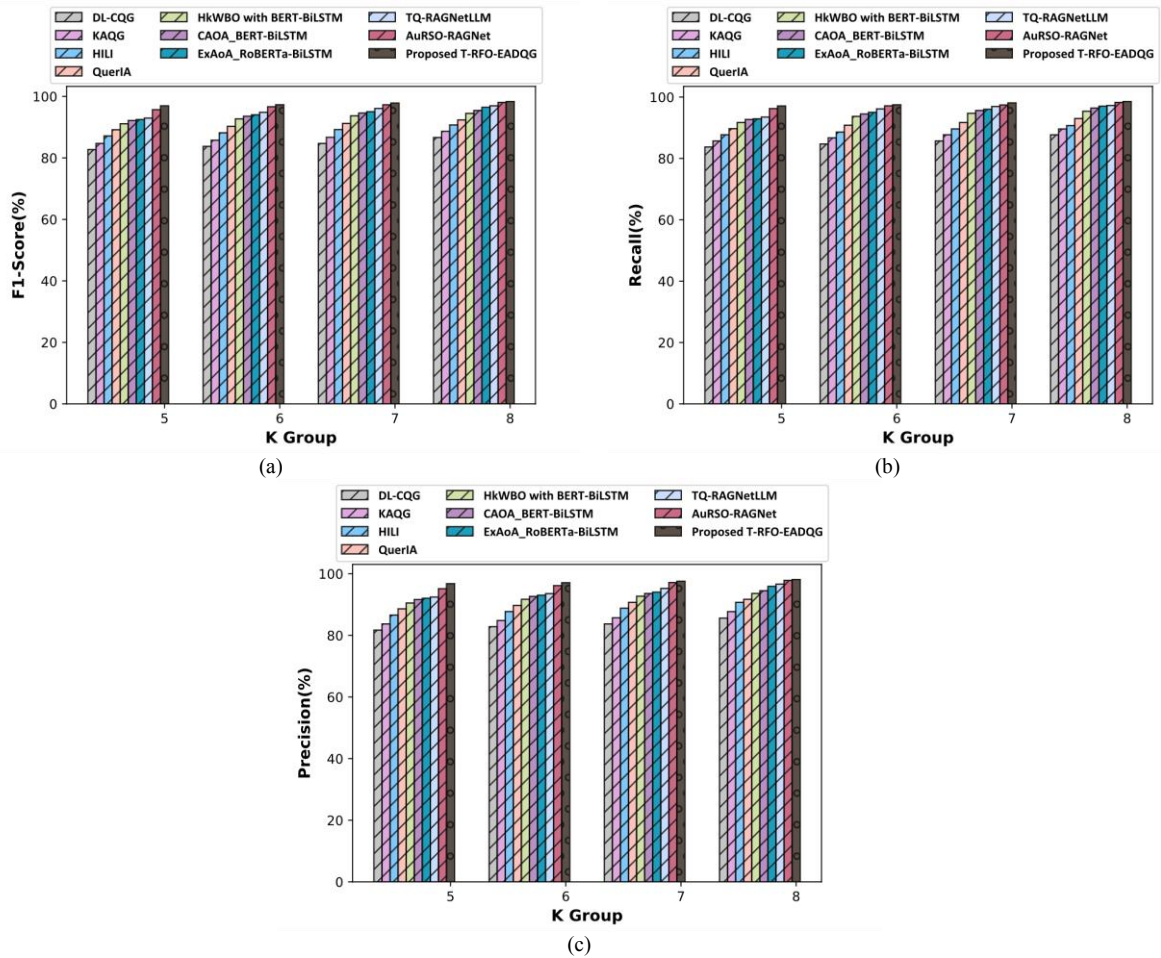


Fig. 6. Validation of T-RFO-EADQG (a) F1-score, (b) Recall, (c) Precision considering MLQA database

F. Comparative discussion

The outcomes obtained by T-RFO-EADQG and other previous approaches employed for question generation concerning MLQA and XQUAD databases are depicted in table 2. The T-RFO-EADQG recorded a high recall of 98.548%, F1-score of 98.335%, and precision of 98.123% for K-group 8 while using the MLQA database. The precision obtained by previous techniques is 85.595% by DL-CQG, 87.704% by KAQG, 90.707% by HILI, 91.704% by QuerIA, 94.483% by CAO_A_BERT-BiLSTM, 93.595% by HkWB0 with BERT-BiLSTM, 96.619% by TQ-RAGNetLLM, 95.916% by ExAoA_RoBERTa-BiLSTM, and 97.840% by AuRSO-RAGNet. Moreover, the recall recorded by baseline models, namely DL-CQG is 87.694%, KAQG is 89.594%, HILI is 90.704%, QuerIA is 93.004%, TQ-RAGNetLLM is 97.259%, ExAoA_RoBERTa-BiLSTM is 97.027%, CAO_A_BERT-BiLSTM is 96.405%, HkWB0 with BERT-BiLSTM is 95.372%, and AuRSO-RAGNet is 98.248%. Also, the F1-score observed by DL-CQG, KAQG, HILI, QuerIA, HkWB0 with BERT-BiLSTM [17], CAO_A_BERT-BiLSTM, ExAoA_RoBERTa-BiLSTM, TQ-RAGNetLLM, and AuRSO-RAGNet [16] is 86.631%, 88.639%, 90.705%, 92.349%, 94.475%, 95.434%, 96.468%, 96.938%, and 98.043%. The extensive experimental results shows that the T-RFO-EADQG attained high performance in generating questions from text document. The T-RFO-EADQG significantly understand the meaning of word based on the succeeding and preceding words to produce rich semantic representations for capturing contextual and syntactic information from the text document.

TABLE 2. COMPARATIVE DISCUSSION

Techniques	For XQUAD database			For MLQA database		
	Evaluation measures (%)			Evaluation measures (%)		
	F1-score	Recall	Precision	F1-score	Recall	Precision
DL-CQG	85.045	85.717	84.383	86.631	87.694	85.595
KAQG	88.535	89.718	87.383	88.639	89.594	87.704
HILI	90.988	91.715	90.273	90.705	90.704	90.707
QuerIA	92.445	92.615	92.275	92.349	93.004	91.704
HkWBO with BERT-BiLSTM	94.984	95.707	94.273	94.475	95.372	93.595
CAOA BERT-BiLSTM	96.025	96.516	95.539	95.434	96.405	94.483
ExAoA_RoBERTa-BiLSTM	96.514	97.018	96.016	96.468	97.027	95.916
TQ-RAGNetLLM	97.313	97.508	97.119	96.938	97.259	96.619
AuRSO-RAGNet	98.037	98.348	97.729	98.043	98.248	97.840
Proposed T-RFO-EADQG	98.272	98.547	97.999	98.335	98.548	98.123

VI. CONCLUSION

Question generation is the key research area in NLP, which helps to create natural language questions based on the text contents. This paper presents the T-RFO-EADQG for generating questions from text documents. Here, the input text document is acquired initially from the dataset, and then, the contextual understanding and quality of question generation are increased by applying the input text document to the round prompting technique. After that, the processed text is fed into T-RFO-EADQG for question generation. The T-RFO-EADQG is developed by training the parameters of EAD-BiLSTM-BERT by utilizing the T-RFO algorithm to improve the question generation performance, where the T-RFO is the incorporation of RFO with the Taylor series. In addition to this, the efficacy of T-RFO-EADQG in question generation is investigated by comparing it with prevailing approaches. The extensive experiments show that the T-RFO-EADQG observed maximum performance with the highest F1-score of 98.335%, recall of 98.548%, and precision of 98.123%. Future research endeavors will be on utilizing hybrid loss functions to modify the learning mechanism of the deep learning scheme for question generation.

REFERENCES

- [1]. L. Wang, R. Song, W. Guo and H. Yang, "Exploring prompt pattern for generative artificial intelligence in automatic question generation," *Interactive Learning Environments*, vol. 33, no. 3, pp. 2559-2584, 2025.
- [2]. C. Badenes-Olmedo and P. Eyzaguirre, "QuerIA: Adaptive Question Generation and Evaluation in Higher Education Using Large Language Models and Contextual Learning," *Expert Systems with Applications*, p. 130140, 2025.
- [3]. G. Kurdi, J. Leo, B. Parsia, U. Sattler and S. Al-Emari, "A systematic review of automatic question generation for educational purposes," *International journal of artificial intelligence in education*, vol. 30, no. 1, pp. 121-204, 2020.
- [4]. N. Mulla and P. Gharpure, "Automatic question generation: a review of methodologies, datasets, evaluation metrics, and applications," *Progress in Artificial Intelligence*, vol. 12, no. 1, pp. 1-32, 2023.
- [5]. X. Dong and Z. Li, "Efficient Unified Framework for Difficulty-Controllable Question Generation: Large Language Models Enabling Active Learning Labeling and Prompt-Based Low-Rank Adaptation," *IEEE Access*, 2026.
- [6]. Z. Wang, Y. Zhang, W. Min, Q. Guan and W. Yu, "GISedu-GPT: a large language model framework with prior knowledge for GIS education question bank generation," *Journal of Geography in Higher Education*, vol. 50, no. 1, pp. 72-99, 2026.
- [7]. H. A. Nguyen, S. Bhat, S. Moore, N. Bier and J. Stamper, "Towards generalized methods for automatic question generation in educational domains," in *European conference on technology enhanced learning*, 2022.
- [8]. C. H. Chen and M. F. Shiu, "Kaqg: A knowledge-graph-enhanced rag for difficulty-controlled question generation," *IEEE Access*, vol. 13, pp. 197234-197244, 2025.
- [9]. X. Zhang, W. Feng and H. Zhang, "From Coarse to Fine-Grained Decomposition: Hierarchical Question Generation and Learned Importance for Automated Fact-Checking," *Expert Systems with Applications*, p. 131215, 2026.
- [10]. "XQUAD dataset," June 2026. [Online]. Available: <https://github.com/google-deepmind/xquad>.

- [11]. "MLQA dataset," June 2026. [Online]. Available: <https://github.com/facebookresearch/MLQA?tab=readme-ov-file>.
- [12]. Q. Zhuge, H. Wang and X. Chen, "A novel design for enhanced test question generation using dual-llm engine," *Applied Sciences*, vol. 15, no. 6, p. 3055, 2025.
- [13]. Y. Saoudi and M. M. Gammoudi, "EAD-Bi-LSTM-BERT: a novel deep learning architecture for arabic question answering systems," *Multimedia Tools and Applications*, pp. 1-23, 2025.
- [14]. Y. Bouali and B. Alamri, "Enhancing radial distribution system performance through optimal allocation and sizing of photovoltaic and wind turbine distribution generation units with rüppell's fox optimizer," *Mathematics*, vol. 13, no. 15, p. 2399, 2025.
- [15]. R. Barrio, M. Rodriguez, A. Abad and F. Blesa, "Breaking the limits: the Taylor series method," *Applied mathematics and computation*, vol. 20, p. 217, 2021.
- [16]. Amit Virmani, Alok Kumar, Vineeta Singh, "An Autoregressive Secant Optimization-Based Dual-View Retrieval-Augmented LLM Framework for Question Answering," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 74, no. 5, pp. 20-35, 2026. Crossref, <https://doi.org/10.14445/22312803/IJCTT-V74I5P104>
- [17]. Virmani, A., Kumar, A., & Singh, V. (2026). A hybrid metaheuristic-optimized BERT-BiLSTM model for advanced question answering systems. *Journal of Integrated Science and Technology*, 14(7), 1602-1602. <https://doi.org/10.62110/sciencein.jist.2026.v14.1602>.
- [18]. Amit Virmani, Dr. Alok Kumar, Dr. Vineeta Singh, 2026, A Comprehensive Survey of Question Answering Systems: From Traditional Approaches to Large Language Models, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 15, Issue 05, May – 2026. DOI : 10.5281/zenodo.20459714.