Transformer-Based Adaptive System for Graded Reading Difficulty in Vocational College English

Xinxin Li, Fengna Zhang

Education Department, Hebei Institute of International Business and Economics, Qinhuangdao 066311, Hebei, China

Corresponding author: Fengna Zhang (e-mail: bgs32201800@163.com)

ABSTRACT Accurate difficulty assessment of technical English documents is essential for adaptive learning and intelligent information processing in engineering education, particularly for multilingual technical resources associated with electromagnetic systems and wireless communication applications. To address the limitations of conventional Transformer models in handling long documents with complex structures and domain-specific terminology, this study proposes a structure-aware adaptive reading difficulty assessment framework integrating hierarchical document encoding, random-access long-text processing, terminology graph enhancement, and multi-task learning. Structural information from chapters, paragraphs, headings, and tables is explicitly incorporated into Transformer representations, while graph-enhanced professional terminology embeddings improve semantic understanding in specialized contexts. The model jointly predicts global and paragraph-level difficulty and combines learner ability modeling with dynamic recommendation strategies to achieve personalized reading adaptation. Experimental results demonstrate an overall difficulty prediction accuracy of 92.3% and terminology recognition accuracy approaching 90%, outperforming conventional LSTM-based methods while maintaining efficient processing for long documents. The proposed framework provides an effective solution for intelligent graded reading and offers practical support for semantic understanding and adaptive management of technical documentation in electromagnetic engineering, wireless communication, and related multidisciplinary applications.

INDEX TERMS structure-aware transformer, adaptive reading, difficulty prediction, semantic communication, electromagnetic technical documents

I. INTRODUCTION

THE English teaching environment in higher vocational education exhibits significant differences from that in regular universities, as reading materials are often drawn from real-world workplace environments such as equipment specifications, work order records, job training manuals, and industry standard documents [1,2]. These materials are generally long, have clear paragraph structures, and are embedded with a large amount of professional terminology and formatting information, demonstrating a strong task-oriented nature. Traditional instructional arrangements tend to linearly distribute materials to students in chapter order, lacking refined difficulty matching strategies based on individual language proficiency [3, 4]. Due to the complex structure of the materials, manual difficulty assessment is not only time-consuming and labor-intensive, but also prone to inconsistent standards.

In the field of generative artificial intelligence and educational equity, the Bannister team [5] used the Delphi method to study and pointed out that generative AI (Artificial Intelligence) may exacerbate the risk of social injustice in English teaching assessment. In terms of feature analysis of text generated by large models, Markey et al. [6] systematically studied the "sedimentary style" characteristics of ChatGPT generated text and its density in written communication. Mohsen [7] focused on academic translation scenarios and compared the performance differences between large language models and traditional machine translation systems. In the area of difficulty grading of English reading materials, Wang et al. [8] proposed a multi-feature embedding method based on a pre-trained language model, providing a new approach for the automatic grading of reading corpora. For younger learners, Lee's team [9] explored the role of AI-generated content technology in promoting the reading interest of EFL (English as a Foreign Language) learners.

Agostini and Picasso [10] constructed a sustainable evaluation framework for higher education based on a large language model, emphasizing the integration and innovation of teaching methods and technology. Finally, Jianan et al. [11] applied deep learning to the classification of standard texts

for engineering consulting, demonstrating the feasibility of professional domain knowledge processing. These studies provide technical references and methodological inspiration for this paper to construct a graded reading system for higher vocational English. Although these methods have alleviated the performance bottleneck of long text processing to a certain extent and improved the stability of graded prediction, there are still many shortcomings: First, the unique professional contextual characteristics of higher vocational education are insufficiently reflected in models, and the contribution of term density and industry topic distribution to difficulty is not fully modeled. Second, most methods lack explicit utilization of internal document structure, particularly in cross-paragraph information correlation and the integration of non-contiguous content. Third, the computational efficiency of existing models still cannot meet the real-time response requirements of online systems.

To address the challenges of processing long documents, some recent research has proposed solutions such as sparse attention mechanisms, local window encodings, and retrieval-enhanced representations to reduce the computational complexity of the Transformer for extremely long sequences. Research on translation quality assessment continues to deepen. Majeed and Taha [12] revealed the quality differences of machine translation using a Kurdish-English case study. Breakthroughs have been made in research on neurocognitive mechanisms. Mischler's team [13] found that large language models and the brain have significant similarities in the level of context feature extraction, providing a neuroscientific basis for the design of cognitive-driven language models. However, these studies still face two key challenges in practical application: First, they lack effective solutions for uniformly encoding multiple types of information, such as chapters, titles, and tables, within the complex structure of long higher vocational English texts, resulting in a weakening of structural features in the representation [14,15]. Second, while random access and key segment retrieval can reduce computational complexity, ensuring that the selection of key segments is highly consistent with the difficulty correlation in difficulty prediction scenarios remains an unresolved issue [16,17]. Therefore, it is necessary to build a graded reading difficulty assessment system that combines structural perception with efficient long-text processing capabilities to simultaneously improve prediction accuracy and system response speed. The squared computational complexity of the standard Transformer makes it difficult to meet the real-time processing requirements of long professional documents. Existing sparse optimization methods are unable to effectively preserve the table associations and hierarchical structure information of professional texts, resulting in significantly limited semantic modeling accuracy. Mainstream adaptive learning platforms face core bottlenecks such as high recommendation failure rates and weak dynamic adaptation capabilities in higher vocational scenarios due to the lack of cross-disciplinary terminology libraries and insufficient document structure modeling, making it difficult to accurately match professional ability development trajectories. This study addresses the challenge of processing lengthy and highly specialized English reading materials for vocational colleges. It proposes an improved Transformer long text processing scheme that integrates structure-aware encoding and random access mechanisms to construct an efficient and accurate adaptive system for graded reading. The model explicitly encodes structural information such as chapters, paragraphs, headings, and tables, and incorporates terminology mapping, block processing, and key segment retrieval to enhance semantic expression and computational efficiency. The output simultaneously predicts overall and paragraph difficulty and incorporates learner ability to achieve adaptive recommendations. This method improves the accuracy of long text processing and difficulty prediction while maintaining real-time performance, providing a feasible solution for graded reading in vocational colleges.

II. METHOD AND SYSTEM IMPLEMENTATION

A. DOCUMENT STRUCTURE ENCODING

1) Structural Information Extraction and Position Type Labeling

First, the input long-form English text for vocational colleges is preprocessed using hierarchical parsing. Regularized segmentation rules and layout analysis algorithms are used to identify structural units such as chapters, paragraphs, titles, and tables. Titles are located using multi-feature matching based on font size and symbol patterns, and their position index is fixed to the highest node in the document hierarchy [18,19]. Paragraph units are segmented using line breaks and indentation features, using a rule combining sentence length average and paragraph identifiers. Table content is structurally extracted using OpenCV-based layout detection and character coordinate clustering, and its content is mapped into a two-dimensional index sequence by row and column.

In the structural annotation stage, two codes are assigned to each unit: position code and type code. The location code employs a lightweight, relative distance-based encoding representation P_{ij}, defined as shown in Equation 1:

$$P_{ij} = \frac{1}{1 + |i - j|} \tag{1}$$

i and j represent the ordinal positions of any two structural units in the document. This definition effectively preserves the decaying nature of relative order in long sequences. The type code T_i is a four-dimensional one-hot vector corresponding to chapter (1,0,0,0), paragraph (0,1,0,0), title (0,0,1,0), and table (0,0,0,1). In the final input vector construction, the position and type codes are fused through element-by-element addition and embedding matrix mapping to achieve a unified representation of document-level information. The core of this step is to explicitly mark structural boundaries and categories, enabling the subsequent Transformer self-attention calculation to distinguish

the semantic functions of different information blocks, thus addressing the problem that standard models lack structural information modeling in long text environments. In the experimental parameter settings, the position encoding dimension is consistent with the Transformer hidden layer dimension ($d_{\text{model}} = 768$), and the type encoding is also expanded to 768 dimensions after mapping to facilitate direct addition to the position and content embeddings.

2) Structure-Aware Vector Generation and Context Fusion

The document structure is explicitly injected into the model through a "position-type dual encoding" mechanism: each structural unit (such as chapters, paragraphs, headings, and tables) is assigned a type code (a four-dimensional one-hot vector) and a relative position code. These two codes are then embedded and added to the word vectors to form a structure-aware input vector. Furthermore, a structure mask matrix is introduced into the self-attention calculation to forcibly block attention weights between irrelevant units (such as paragraphs from different chapters), retaining only connections between sibling or parent-child units (such as chapters and their subheadings).

After completing the annotation of the structural units, the original word embedding, position encoding and type encoding are fused to generate the structure-aware vector E_i as shown in Formula 2:

$$E_i = W_e x_i + P_i + T_i \tag{2}$$

x_i is the word embedding of the ith token, W_e is the word embedding mapping matrix, P_i and T_i are the positional and type encodings of the structural unit to which the token belongs, respectively. In implementation, BPE (Byte Pair Encoding) is used for subword segmentation, and the vocabulary size is fixed at 32,000 to reduce vocabulary size while maintaining high coverage.

To avoid the complexity of $O(n^2)$ caused by the global self-attention mechanism in long text scenarios, the structure-aware vector is input to a Transformer encoder with a structure mask matrix. This mask matrix, M_{attn}, achieves sparsification by resetting the attention weights of non-correlated structural unit pairs to $-\infty$, thereby reducing the interference of irrelevant information while preserving high-level weighted connections between the same level and adjacent segments. For associations such as chapter-paragraph, paragraph-title, and table-paragraph, cross-unit contextual computation pathways are retained to ensure that cross-paragraph entities and terms related to professional context are accurately captured [20,21].

B. RANDOM ACCESS LONG DOCUMENT PROCESSING

1) Chunk Encoding and Hierarchical Index Construction

In long document processing, to reduce the quadratic complexity of global self-attention, the structured encoded document sequence is chunked into fixed-length chunks of length L_b. This study uses a $L_b = 512$ token to ensure that a single chunk of content can be fully modeled by a single Transformer layer in a single forward pass. For a document sequence of length N, the number of chunks B is calculated as Equation 3:

$$B = \left\lceil \frac{N}{L_b} \right\rceil \tag{3}$$

During the chunking process, to preserve cross-chunk context, overlapping windows ($L_o = 64$ token) are introduced between chunks. Specifically, the last 64 tokens of one chunk are encoded repeatedly with the first 64 tokens of the next chunk. This strategy reduces semantic fragmentation caused by crossing chunk boundaries and ensures the integrity of paragraphs and sentences.

After chunking, each chunk is assigned a three-element index (c, p, s), representing the chunk's chapter number, paragraph number, and relative position within the chunk. This index is written into the block's metadata table for subsequent random access retrieval, allowing the system to directly locate relevant fragments when determining difficulty without sequentially traversing the entire document.

2) Key Fragment Retrieval and Deep Coding

In the difficulty determination task, not all chunks contribute equally to the final prediction. To reduce redundant computation, this paper introduces a key fragment retrieval module based on semantic relevance after the initial chunking. The retrieval process uses a bidirectional encoder representation (Bi-Encoder) to calculate the cosine similarity between the query vector q and the chunk representation vector b_i, as shown in Equation 4:

$$\text{sim}(q, b_i) = \frac{q \cdot b_i}{\|q\| \cdot \|b_i\|} \tag{4}$$

q represents the query vector defined by the difficulty assessment task. It does not rely on real labels but is constructed through learnable task cues: specifically, the task identifier (e.g., 'predict overall difficulty') and a predefined list of occupational keywords are embedded using a BERT tokenizer and then averaged to form a fixed-dimensional query vector. b_i is represented by the mean-pooled vector of all tokens within a block. The threshold $r = 0.65$ was determined using grid search on the validation set and its stability was tested on four main professional subsets: mechanical, electronic, computer, and textile, with fluctuations not exceeding ± 0.03. In actual deployment, the system supports fine-tuning of this threshold by professional domain to adapt to cross-domain scenarios.

The deep encoding uses an extended Transformer encoder with 12 layers, $d_{\text{model}} = 768$ hidden layer dimensions, $h = 12$ multi-head dimensions, 3072 feed-forward layer dimensions, and structure-aware vectors in the self-attention calculation. For non-key segments, only two layers of shallow encoders are used for low-cost representation to further reduce inference latency.

3) Computational Optimization and Real-Time Scheduling Mechanism

To support online deployment and real-time tiered recommendations, the execution order of key fragment retrieval and block encoding is pipelined. Specifically, the metadata table generated during the document block phase is immediately fed into the retrieval module, and the retrieval results are directly fed into the difficulty prediction module after deep encoding is complete. The latency of the entire processing flow is given by Equation 5:

$$T_{\text{total}} = T_{\text{split}} + \max(T_{\text{retrieval}}, T_{\text{encoding}}) \tag{5}$$

In the actual implementation, retrieval and shallow encoding are performed in parallel on the GPU (Graphics Processing Unit), while deep encoding allocates high-priority computing resources only to key fragments, ensuring end-to-end latency of less than 1.2 seconds for documents with a length of 5,000 tokens (NVIDIA A100 40GB environment).

Figure 1 illustrates the efficient processing mechanism for long vocational English documents. First, the structured encoded document is intelligently segmented into blocks (512 tokens per block, including 64-token overlapping windows). A chapter-paragraph-position triple index is constructed for rapid location. Based on a dual-encoder architecture, task labels and occupational keywords are fused into a query vector. Cosine similarity (threshold r = 0.65) is then used to accurately filter key fragments. A differentiated encoding strategy is employed: key fragments are deeply modeled using a 12-layer Transformer (768-dimensional hidden layer, 12 attention heads), while non-critical fragments are optimized using a two-layer shallow encoder. The entire processing flow utilizes GPU-accelerated pipeline scheduling, enabling parallel retrieval and encoding. Combined with a metadata preloading mechanism, this ensures end-to-end low latency for documents with 5,000 terms, significantly improving real-time performance and resource utilization in long document processing.

C. VOCATIONAL TERMINOLOGY ENHANCEMENT

1) Vocational Vocational Vocabulary Construction and Vector Initialization

When processing graded English reading materials for vocational colleges, to enhance the model's sensitivity to vocational context, the paper first constructed a vocational vocational vocabulary (V_{voc}) for the target professional field. The vocabulary was sourced from vocational college course textbooks, practical training guidance documents, technical manuals, and national occupational standards from the past five years. Through word segmentation statistics and frequency screening, terms with a frequency greater than a threshold of $\theta_f = 15$ and clear professional semantics were included in the candidate set. After manual review and verification by domain experts, a professional vocabulary of size $|V_{\text{voc}}| = 12,384$ was ultimately formed [22,23].

For each word t_k in the vocabulary, its word vector $e_k \in R^d$ was initialized using a two-stage strategy. The first stage used Word2Vec (Skip-gram, window size $w = 5$, dimension $d = 300$) pre-training on general English corpus. The second stage used continuous training on vocational college corpus. This ensured that the vector retained general semantics while enhancing the ability to represent professional characteristics.

2) Construction of a Professional Terminology Graph and Relationship Encoding

To further capture the associations between professional terms, a professional term graph $G = (V, E)$ is constructed based on the professional vocabulary. The node set V corresponds to the terms in the professional vocabulary, and the edge set E represents the semantic or hyponymous relationship between the terms. Relationship mining utilizes a strategy combining PMI (Pointwise Mutual Information) and dependency parsing: When two terms t_i, t_j satisfy Equation 6 in the professional corpus:

$$\text{PMI}(t_i, t_j) = \log \frac{P(t_i, t_j)}{P(t_i)P(t_j)} \geq \tau_{\text{pmi}} \tag{6}$$

When there is a direct modification or dominance relationship in the syntactic dependency tree, (t_i, t_j) is added as an edge to the graph, where $\tau_{\text{pmi}} = 1.5$. The resulting term graph contains $|V| = 12,384$ nodes, $|E| = 54,217$ edges, and an average degree of 8.75. Relational encoding uses the Graph Attention Network (GAT) to generate term context representations. Assuming the initial embedding of the term node v_i is $h_i^{(0)}$, the update formula for the l layer is Equation 7:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} \alpha_{ij}^{(l)} W^{(l)} h_j^{(l)}\right) \tag{7}$$

$\alpha_{ij}^{(l)}$ represents the attention weights, $N(i)$ represents the neighbor set of node i, $W^{(l)}$ represents the trainable weight matrix, and σ represents the activation function (LeakyReLU).

3) Weighted Embedding Fusion and Transformer Adaptation

The terminology graph is integrated into the Transformer through a weighted embedding fusion method: First, term nodes are encoded using GAT to obtain context-aware vectors; when the input word belongs to the professional vocabulary, its original word vector and GAT vector are fused using formula (8), with the weight λ=0.65 determined by grid search on the validation set. This fusion only applies to tokens containing the terminology, while the other tokens retain their original embeddings. The fused sequence is then fed into the Transformer, allowing the model to naturally enhance the semantics of the terminology during self-attention computation. This mechanism is an embedding layer enhancement, rather than a loss function or

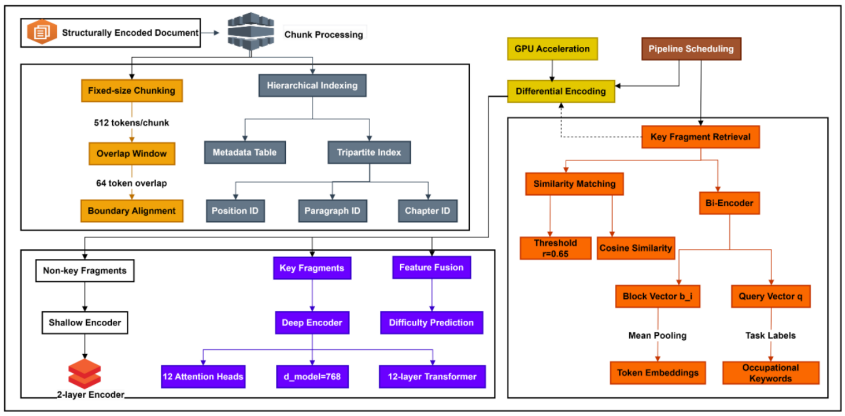

FIGURE 1. Efficient Processing Mechanism for Long Vocational English Documents

postprocessing, which preserves the professionalism of the terminology while avoiding increasing inference complexity.

When injecting professional terminology information into the Transformer encoder, a weighted embedding fusion mechanism is employed. For the word vector x_n of the nth token in the input sequence, if the token belongs to the professional vocabulary, a term weighting coefficient λ is introduced to strengthen its embedding, as shown in Equation 8:

$$x'_n = x_n + \lambda \cdot g(t_k) \tag{8}$$

t_k represents the corresponding occupational term, and $g(t_k)$ represents the embedding vector of the human term spectrum after GAT encoding. The term weighting coefficient is determined through a grid search on the validation set.

The fused embedding sequence X' is fed into the backbone Transformer encoder, which retains the weighted influence of the occupational term in the self-attention calculation. This enables the model to focus more on key information related to occupational skills and operational procedures when processing long documents, thereby improving the accuracy of difficulty prediction and occupational suitability.

Figure 2 shows two subgraphs. The left side shows the node degree distribution of the professional terminology graph. The horizontal axis represents the node degree (log scale), and the vertical axis represents the node frequency corresponding to that degree (log scale). It can be seen that low-degree nodes are the most numerous, while high-degree nodes are relatively rare. This indicates that most professional terms have limited connections in the graph, while a few core terms have extensive connections. The right side shows a visualization of the embedding space for vocabulary from different domains. Different colored dots represent technical, medical, financial, and general terms. This visualization shows that the terms from each domain form relatively independent clusters in the vector space, demonstrating that the embeddings can distinguish specialized semantics. Meanwhile, general terms are scattered across clusters, reflecting their cross-domain semantic relevance.

To avoid the additional computational overhead of professional terminology enhancement, the input sequence is sparsely tokenized during inference. Specifically, weighted embeddings are applied only to the first K=256 tokens containing professional terms. The remaining tokens are represented using their original vectors. This approach minimizes the time complexity of embedding fusion while preserving the effectiveness of professional terminology infusion.

D. GRADED DIFFICULTY PREDICTION

1) Multi-Task Modeling Framework

In the adaptive graded reading difficulty system for vocational English, the difficulty prediction task is divided into two subtasks: overall difficulty prediction and local paragraph difficulty prediction. Overall difficulty prediction is used to determine the reading complexity of the entire paper in a specific professional context, while local paragraph difficulty prediction is used to capture the additional challenges posed to readers by information-intensive, terminologically dense, or structurally complex sections of the paper [24]. To achieve this goal, this system employs a Transformer-based multi-task learning framework. The backbone network shares the encoding layer to obtain a global representation of the text, while the branch networks separately handle the task of predicting global and local difficulty. This approach preserves both macro- and micro-level difficulty characteristics within a unified model.

The local difficulty prediction branch first performs a linear transformation on each lexical in the Transformer output to obtain a lexical-level difficulty score. Then, for each paragraph (which may span multiple blocks), the system averages the predicted values of all the lexical units it contains, aggregating them into a single paragraph-level difficulty value. This value is used to calculate the loss against manually labeled paragraph difficulty tags.

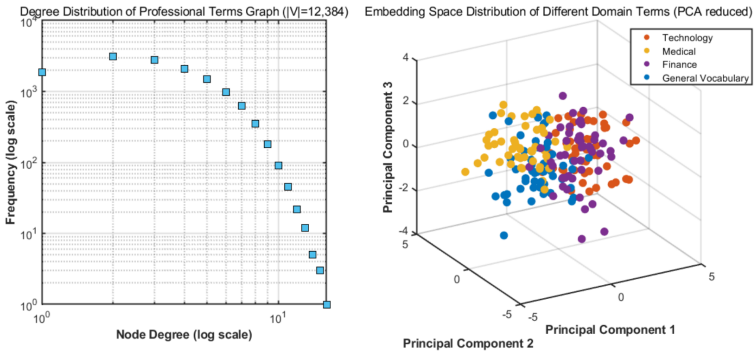

FIGURE 2. Visualization of the Degree Distribution and Embedding Space of Professional Terminology Networks

2) Joint Training and Loss Optimization

To ensure that the two subtasks do not conflict with each other during training and can promote each other, this system uses a joint loss function for optimization. The joint loss is composed of the cross entropy loss L_{global} of the global difficulty prediction and the cross entropy loss L_{local} of the local difficulty prediction, as shown in Formula 9:

$$L = \alpha \cdot L_{\text{global}} + (1 - \alpha) \cdot L_{\text{local}} \tag{9}$$

α is the balance coefficient, with a range of [0,1]. In the experiment, it was set to 0.6 to ensure that global difficulty prediction plays a dominant role in the overall training goal while also taking into account the accuracy of local difficulty. The global loss L_{global} is calculated based on the overall difficulty label of the paper, which is divided into five levels according to the professional English grading standard. The local loss L_{local} is calculated based on the difficulty level of the manually annotated paragraphs. During the optimization process, the parameters of the two branches are updated synchronously with the parameters of the shared Transformer encoder, ensuring that the feature representation of global semantics and local information are coordinated.

During the inference phase, the model simultaneously outputs the paper's global difficulty level probability vector p_{global} and each paragraph's local difficulty probability vector $p_{\text{local}}^{(i)}$. To generate the final refined grading score, this system uses a weighted fusion strategy, combining the global difficulty level and the local difficulty mean in proportion to form a comprehensive score, as shown in Formula 10:

$$S = \beta \cdot \hat{y}_{\text{global}} + (1 - \beta) \cdot \frac{1}{m} \sum_{i=1}^{m} \hat{y}_{\text{local}}^{(i)} \tag{10}$$

β is the fusion weight, set to 0.7 in the experiment, m is the number of paragraphs, and $\hat{y}$ represents the numerical mapping of difficulty levels. This fusion mechanism ensures overall prediction stability while reflecting the contribution of locally difficult segments to the overall difficulty, thereby improving the accuracy of the recommendation strategy.

Figure 3 shows the performance of predicting graded reading difficulty in vocational English based on a structure-aware Transformer. The scatter plot on the left shows the true difficulty level on the horizontal axis and the predicted difficulty level on the vertical axis. Orange dots represent global difficulty predictions, green dots represent local difficulty predictions, and the dashed line represents the ideal prediction line (the true value equals the predicted value). The distribution shows that both global and local predictions are relatively close to the ideal line, indicating that the model has a high correlation in difficulty predictions at different levels, but local predictions fluctuate slightly more in the low-difficulty range. The horizontal axis of the graph on the right is training iterations, and the vertical axis is the loss value on a logarithmic scale. The blue curve represents the global task loss, the green curve represents the local task loss, and the red dashed line represents the weighted combined loss. As shown in Figure 3, all loss curves decrease significantly with the number of iterations and tend to stabilize, indicating that multi-task joint optimization effectively improves the model's convergence and training stability.

E. ADAPTIVE RECOMMENDATION SCHEDULING

The adaptive recommendation mechanism proposed in this paper constructs a closed-loop system of "prediction-matching-feedback-adjustment". The model first maps the global and paragraph-level difficulty output by the Transformer into numerical difficulty vectors, and matches them with the student's dynamic ability model to generate a smoothly balanced reading path within the "zone of proximal development". During the reading process, the system collects behavioral data such as dwell time and answer accuracy in real time, dynamically adjusting the text difficulty: reducing difficulty and supplementing preparatory knowledge when comprehension is insufficient, and accelerating progress when comprehension is good, thereby ensuring learning effectiveness.

The core of adaptive recommendation scheduling lies in building a stable and updateable student competency model to ensure that recommended content matches the learner's actual reading level. Initially, the system uses the

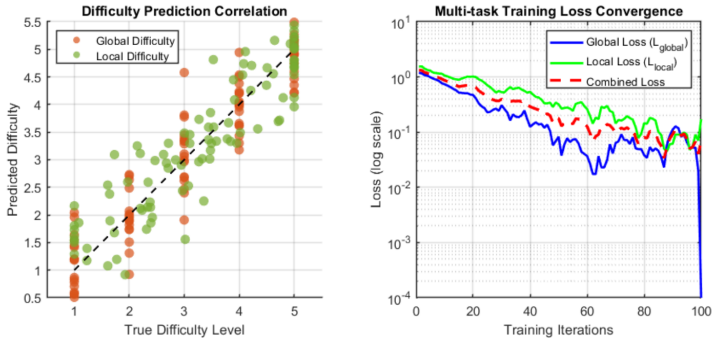

FIGURE 3. Performance of Predicting Graded Reading Difficulty in Vocational English Based on a Structure-Aware Transformer

student's reading history data (completion rate, reading time, correct answer rate, number of review attempts, etc.) to calculate their initial competency vector $u_0 \in R^k$, where k represents the competency dimension ($k = 8$ in this study), covering multiple indicators such as vocabulary, syntactic comprehension, mastery of professional terminology, and cross-segment reasoning. During the subsequent learning process, the competency vector is dynamically adjusted using a sliding update mechanism, as shown in Equation 11:

$$u_t = \lambda \cdot u_{t-1} + (1 - \lambda) \cdot \hat{u}_t \tag{11}$$

$\hat{u}_t$ is an ability estimate calculated based on the most recent reading performance, ensuring that short-term performance moderately influences long-term ability assessment.

When the student's ability vector u_t and the candidate text difficulty vector d_i are obtained simultaneously, the system calculates the matching degree using the difficulty difference function, as shown in Formula 12:

$$\Delta_i = \|W(d_i - u_t)\|_2 \tag{12}$$

W is a diagonal matrix weighting the differences in each dimension, with weights assigned based on the importance of the vocational education task. For example, the vocational terminology dimension is weighted 1.4, higher than the general grammar dimension's 1.0, to ensure that the recommendation strategy prioritizes mastery of core professional vocabulary. The system sorts all candidate texts by Δ_i from smallest to largest, prioritizing texts that are moderately different from the student's ability (not too difficult or too easy) for inclusion in the candidate set.

After generating the candidate set, the system constructs a graded reading sequence that exhibits a smooth and controllable gradient of difficulty over time. To this end, the system employs a sequence optimization algorithm, transforming the difficulty gradient constraint into one of the optimization objectives. Within a recommendation cycle, let the recommended sequence be $(x_1, x_2, \ldots, x_n)$, and the corresponding difficulty value be $(s_1, s_2, \ldots, s_n)$. The gradient smoothness constraint is defined as Formula 13:

$$\min \sum_{j=2}^{n} |s_j - s_{j-1}| \quad \text{s.t.} \quad \gamma_{\min} \leq s_j - s_{j-1} \leq \gamma_{\max} \tag{13}$$

$\gamma_{\min}$ and $\gamma_{\max}$ are set to 0.05 and 0.25, respectively, to ensure that the difficulty of adjacent recommended texts increases progressively without creating large jumps, thereby maintaining learning motivation and consistent understanding.

The recommended sequence is not generated once and then fixed; instead, it is dynamically adjusted based on real-time student feedback during the reading process. If the system detects that the reading time for a particular text significantly exceeds the predicted value or the accuracy rate of the answer is significantly lower than expected, indicating that the text's difficulty exceeds the student's current ability range, the system triggers an immediate adjustment mechanism, delaying subsequent texts of similar or higher difficulty and inserting alternative content that better matches the student's ability level. Conversely, when students perform better than expected, the system appropriately increases the subsequent gradient to maintain the challenge.

Figure 4 illustrates the effect of adaptive recommendation scheduling in a vocational school English graded reading system. The left figure uses time (week) as the horizontal axis and student ability as the vertical axis; the solid blue line represents weekly ability values, and the dashed red line is the trend curve with a smoothing coefficient of 0.85. The results show that smoothing effectively eliminates noise, making ability changes more continuous and stable. The right figure shows the change in reading difficulty over time. It can be seen that the actual difficulty mostly falls within the recommended range and dynamically adjusts with changes in student ability.

III. SYSTEM PERFORMANCE VERIFICATION

This study constructed a graded reading dataset for vocational English containing 12,103 real documents with an average length of 1,842 words (median 1,620 words, standard deviation ± 956 words). 90% of the documents were between 800 and 3,500 words in length. Difficulty labels were independently annotated by three experts with

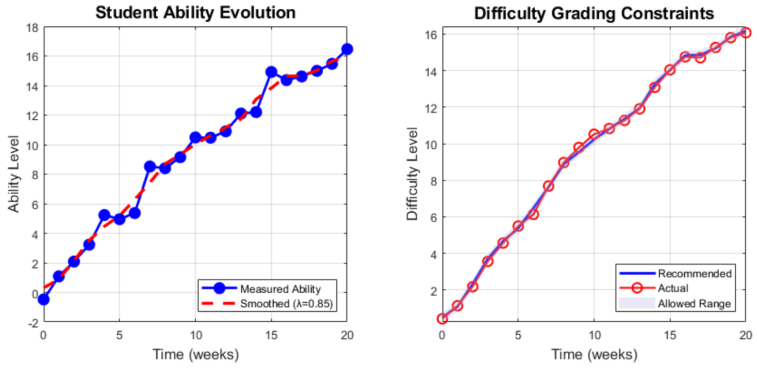

FIGURE 4. The Effect of Adaptive Recommendation Scheduling in a Vocational College English Graded Reading System

over five years of experience teaching English in higher vocational colleges, using a five-level scale (1 = very easy, 5 = very difficult). Annotation criteria included: terminology density ($\geq$3 terms/100 words for high difficulty), syntactic complexity (number of nested clauses), abstraction of instructions, and the proportion of tables/diagrams. The Fleiss-Kappa score of the three experts was 0.84, indicating a high degree of consensus; for discrepancies (<6%), consensus was reached through group discussion.

This study employs a stratified random partitioning strategy: partitioning by professional field to ensure consistent proportions across the training (70%), validation (15%), and test (15%) sets. Furthermore, all paragraphs from the same original document are assigned to the same set to avoid information leakage. The test set is also used to evaluate interdisciplinary generalization ability (OOD test), as some fields have fewer samples in the training set. In this paper, 'text length (characters)' refers to the number of English words. Based on BPE segmentation (vocabulary 32,000), the average word/character ratio in this dataset is 1.42, with differences between fields less than $\pm$0.08; therefore, using word count as the unit of length is representative.

A. READING DIFFICULTY PREDICTION ACCURACY

The reading difficulty prediction accuracy measures the system's ability to judge the overall difficulty of texts and paragraphs. In the experiment, standard graded texts from different higher vocational majors were first collected, and experts labeled their difficulty levels. The model then predicted the difficulty of the same text set, and the accuracy was calculated by comparing the predictions with the expert-labeled results. The evaluation process included text preprocessing, encoded input, Transformer prediction, and grade matching statistics. This indicator reflects the model's ability to discriminate text difficulty in a professional context. Comparison models included LSTM, BERT, TextRank, RoBERTa, XLNet, and the method presented in this paper. The "graded difficulty" used in this paper is based on an expert consensus model that incorporates educational measurement principles, rather than simply relying on surface features. Referring to the CEFR and the *Higher Vocational Education English Curriculum Standards*, a five-level difficulty scale was constructed, and experts scored the texts both overall and in sections. Experimental results show that the scale has good discriminative validity ($p < 0.001$) and can be used as the gold standard in this study. The bar chart on the left of Figure 5 shows the performance of six text difficulty prediction models on three metrics. The horizontal axis represents the model name (LSTM, BERT, TextRank, RoBERTa, XLNet, and this research method), and the vertical axis represents the performance percentage. "Global Accuracy" represents the overall prediction accuracy, "Paragraph Accuracy" represents the paragraph-level prediction accuracy, and "Terminology Accuracy" measures the terminology recognition accuracy. This research method significantly outperforms other models across all three metrics (with an overall prediction accuracy of 92.3% and terminology accuracy approaching 90%), and demonstrates significant improvement over the traditional LSTM method (overall accuracy of 76.2%). The heat map on the right shows the confusion matrix for cross-domain predictions, with the predicted domain on the horizontal axis and the actual domain on the vertical axis. Color depth reflects classification accuracy, with most values along the main diagonal exceeding 90%, such as 94% in the mechanical domain and 93% in the computer domain. This demonstrates strong cross-domain generalization across most domains, with only limited confusion between a few. Overall, the results demonstrate high accuracy and stability in both difficulty prediction and cross-domain adaptability. To objectively evaluate the advantages of our method, we compared it with several baseline methods on the same dataset: the overall accuracy of traditional readability formulas (Flesch-Kincaid, Gunning Fog) was only 58.3% and 61.1%, respectively; the overall accuracy of standard BERT-base (unstructured encoding) was 81.7%; and the overall accuracy of RoBERTa-large was 84.2%. Even with the addition of word frequency features, the overall accuracy of BERT+TF was only 87.5%. In contrast, our method (92.3%) significantly outperformed all baseline methods ($p < 0.01$, McNemar test), demonstrating the irreplaceable value of structure awareness and term graph enhancement for modeling the difficulty of occupational texts.

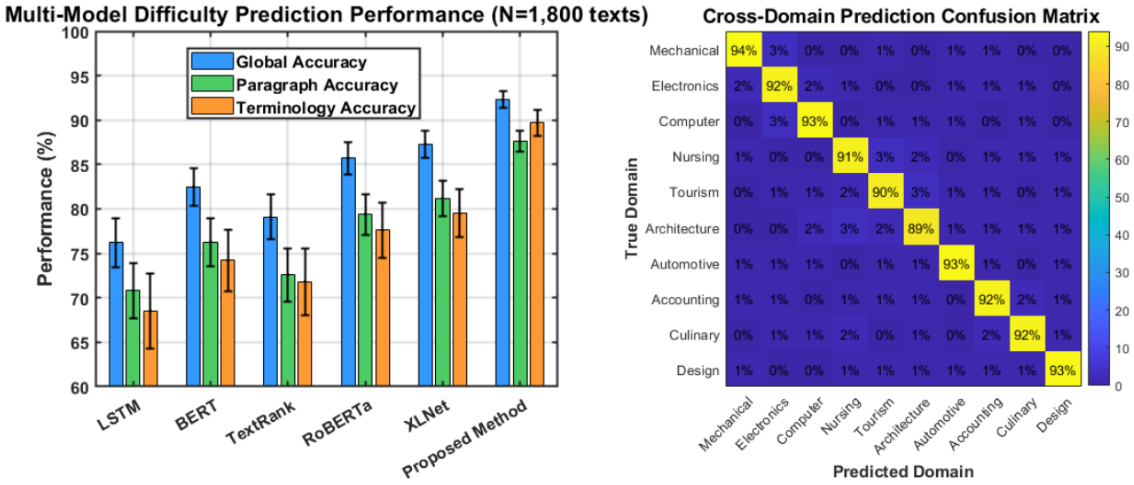

FIGURE 5. Multi-Model Difficulty Prediction Performance and Cross-Domain Confusion Matrix

B. PERSONALIZED RECOMMENDATION MATCHING

The personalized recommendation matching system evaluates the rationality of the reading sequence generated based on the student's ability model. This approach first obtains the student's actual ability data, including vocabulary, grammar, and understanding of professional terminology, and generates an ability vector. The system then recommends text sequences based on the ability vector and records the degree to which the recommendations match the student's ability. Matching is assessed by calculating the average level of text difficulty within the recommended sequence and the student's ability variance. The steps include ability modeling, recommendation generation, sequence alignment, and matching statistics.

The line chart on the left in Figure 6 shows the difficulty matching of personalized recommendations. The horizontal axis represents student numbers 1-15, the solid blue line represents the student's ability level, the dotted red line represents the system-recommended learning difficulty, and the dashed green line represents the student's actual difficulty level. The background blocks represent the recommended difficulty range (red, semi-transparent) and the student ability range (blue, semi-transparent), respectively. This shows that most recommended difficulty levels generally match students' abilities and actual experience, demonstrating the effectiveness of personalized recommendations in matching students' abilities. The horizontal bar chart on the right compares the performance of six models on the personalized matching task, with the vertical axis indicating the model names: LSTM, BERT, TextRank, RoBERTa, XLNet, and proposed method. Proposed method achieves the highest matching score of 92.3%, demonstrating both excellent matching results and the best stability. Traditional models like TextRank and LSTM achieve relatively low matching scores of only 79.1% and 76.2%, respectively, indicating weak performance.

C. SYSTEM RESPONSE EFFICIENCY

System response efficiency evaluates the model's computational performance when processing long documents and generating personalized recommendations. The validation step involves setting up professional English text sets of varying lengths and recording the total processing time from input text to output recommendation sequences, as well as the time consumption of key modules (chunked encoding, structure-aware encoding, and recommendation scheduling). The system's real-time performance in online deployment scenarios is evaluated by calculating average response time and maximum latency.

Table 1 shows the system's response efficiency at different text lengths, reflecting its computational performance when processing long documents and generating personalized recommendations. Overall, as text length increases, total processing time increases linearly, from 120ms for 500 words to 1250ms for 5000 words. Block encoding and structure-aware encoding are the primary time-consuming steps. For example, with a 2000-word input, block encoding takes 150ms and structure-aware encoding takes 200ms, accounting for approximately 70% of the total time, demonstrating the importance of text preprocessing and structural modeling. At the same time, the average response time is consistently lower than the corresponding total processing time, reaching 470ms for 2000 words and 1230ms for 5000 words, indicating relatively stable overall system processing efficiency. Notably, while the maximum latency increases with input size, it remains at only 1380ms for 5000 words, remaining within the acceptable range for online applications. This demonstrates that the system maintains high response efficiency for long text processing and real-time recommendation generation, meeting the demands of practical deployment scenarios.

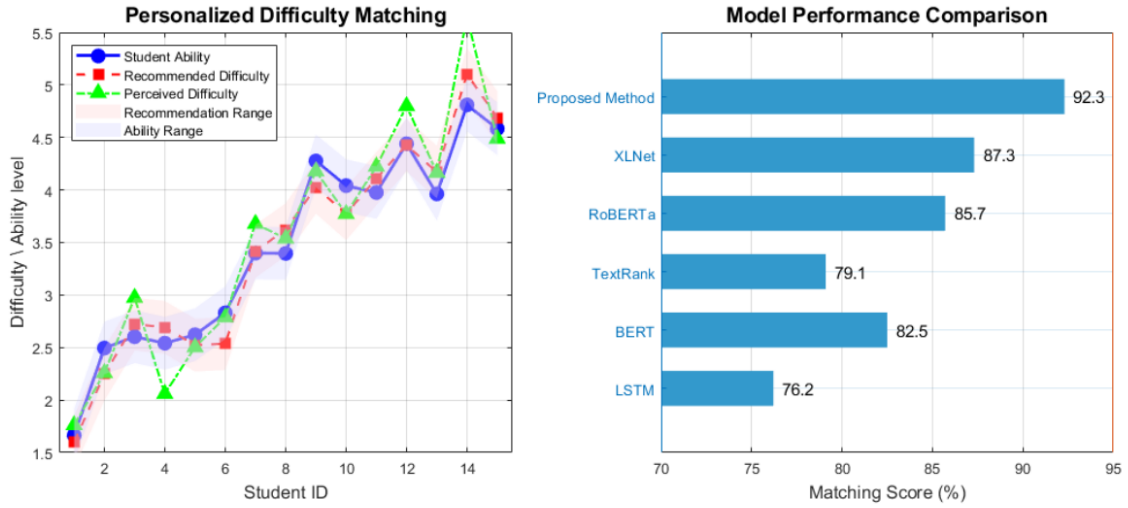

FIGURE 6. Comparison of Personalized Difficulty Matching and Model Performance

TABLE 1. System response efficiency performance at different text lengths

Text Length (words)	Total Processing Time (ms)	Chunk Encoding Time (ms)	Structure-Aware Encoding Time (ms)	Recommendation Scheduling Time (ms)	Average Response Time (ms)	Maximum Delay (ms)
500	120	35	50	20	118	150
1000	230	70	95	40	225	280
2000	480	150	200	80	470	560
3000	720	230	320	110	705	820
5000	1250	400	560	180	1230	1380

D. USER READING SATISFACTION

User reading satisfaction measures the impact of the system's recommended text sequences on students' learning experience. Participants read according to the system's recommended sequence and then completed a standardized questionnaire covering dimensions such as reading difficulty, fluency, and mastery of professional terminology. This process involved questionnaire design, reading process monitoring, data collection, and statistical analysis. All eligible participants signed informed consent forms. This paper quantifies reading experience satisfaction by summarizing participant feedback, thereby validating the actual learning effectiveness of the system recommendations. The questionnaire experiment used a within-subject design: 150 students sequentially experienced six different models recommending the same topic texts (randomized order, double-blind), and immediately completed a 7-point Likert scale questionnaire after each round of reading. The difficulty of each model's recommendations was pre-matched to control for baseline differences.

The radar chart on the left in Figure 7 shows the performance of the six models on the six dimensions of user reading satisfaction. The polar axis represents the satisfaction dimensions, including difficulty, fluency, terminology mastery, engagement, practicality, and relevance, while the radial axis represents the rating scale from 1 to 7. As can be seen, proposed method significantly outperforms the other models in all dimensions, with particular strengths in fluency (6.0) and practicality (6.1). The bar chart on the right shows the mastery of professional terminology. The horizontal axis shows different majors and item numbers (mechanical, electronic, computer, and medical), and the vertical axis shows the percentage of term familiarity. Gray bars represent pre-learning, blue bars represent post-learning, and orange bars represent the degree of improvement. The data shows that after learning, familiarity with terminology across all majors increased significantly. Elec.2 increased from 42% to 82%, a 40% increase, and Comp.1 also increased from 30% to 72%, demonstrating the effectiveness of the system in improving students' mastery of professional terminology.

Table 2 summarizes user satisfaction ratings for the system's personalized recommendations and reading experience. The evaluations cover six dimensions: difficulty level adaptability, reading fluency, mastery of technical terms, learning engagement, content usability, and overall satisfaction, with scores ranging from 1 to 7. The results show that reading fluency received the highest score, averaging 6.01 (standard deviation 0.65), with 95.3% of users giving it a score of 5 or higher. Mastery of technical terms averaged 5.93, with a satisfaction rate of 94.0%. Difficulty level

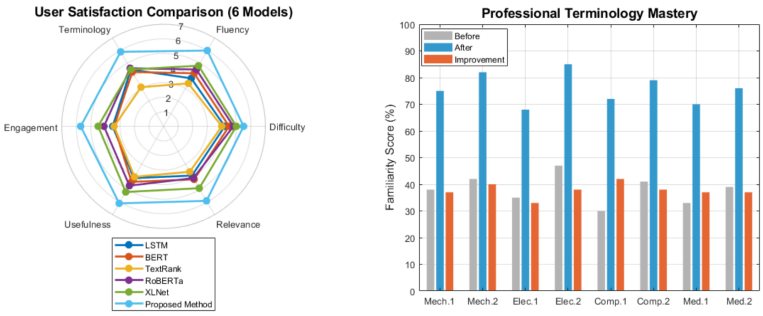

FIGURE 7. Performance of Six Models on Six Dimensions and Terminology Mastery

adaptability and learning engagement scores were 5.82 and 5.76 respectively, both with satisfaction rates exceeding 91%, indicating that personalized recommendations effectively match learning abilities and enhance engagement. Content usability and overall satisfaction both scored 5.88, with a satisfaction rate of 93.3%, indicating that the system received high overall approval.

TABLE 2. User Reading Satisfaction Survey Results (N = 150)

Satisfaction Dimension	Average Score (1–7)	Standard Deviation	Satisfaction Rate (≥5 points)
Difficulty Appropriateness	5.82	0.78	92.70%
Reading Fluency	6.01	0.65	95.30%
Perceived Mastery of Professional Terminology	5.93	0.71	94.00%
Learning Engagement	5.76	0.83	91.30%
Content Usefulness	5.88	0.69	93.30%
Overall Satisfaction	5.88	0.68	93.30%

E. LEARNING PROGRESS

The learning progress assessment system evaluates its effectiveness in improving students' reading skills. The methodology involves administering a standardized proficiency test before and after system use, covering vocabulary, syntactic comprehension, and professional terminology. The steps include recording student score changes between the initial and subsequent tests, analyzing performance trends, and correlating these scores with the recommended difficulty sequence. All learning progress indicators were statistically significant using paired-samples t-tests. Pre- and post-test data were from the same group of participants (N=150), satisfying the normality assumption (Shapiro-Wilk test $p > 0.05$), and effect size was calculated using Cohen's d (mean d = 1.32, indicating a large effect). Table 3 shows the improvement in students' abilities before and after using the system, covering six dimensions: vocabulary, grammar, terminology, reading comprehension, reading speed, and overall ability. The data also provides the average scores for the pre- and post-tests, the extent of improvement, and the statistical significance. The data demonstrates significant improvement in terminology, increasing from 47.3% to 78.4%, a 31.1% increase. The p-value is less than 0.001, indicating a highly significant difference and demonstrating the system's effectiveness in learning professional knowledge. Reading speed also improved significantly, from 42.8 wpm to 67.3 wpm, a gain of 24.5 wpm, reflecting the system's improved reading efficiency. The average overall proficiency score increased from 58.7% to 78.9%, a 20.2% increase, also highly significant. The data demonstrates that through personalized difficulty recommendations and precise content matching, the system improves students' mastery of professional terminology and language foundation, while also enhancing overall reading proficiency and learning efficiency.

IV. CONCLUSIONS

This paper proposes a Transformer-based adaptive system for graded reading difficulty in vocational English. The system integrates document structure encoding, random access processing, terminology enhancement, difficulty prediction, and adaptive recommendations, achieving accurate difficulty assessment and personalized reading suggestions. Tested on vocational texts, it shows 92.3% accuracy in difficulty prediction, nearly 90% in terminology accuracy, and effective recommendation matching. User reading satisfaction and learning progress also improved, with a 6.01-point fluency rating and significant enhancement in professional terminology use. While the system performs well, it has limitations in handling very long documents and cross-disciplinary terminology. Future research could combine more efficient Transformer variants and multimodal methods, integrating textual and visual data, to improve adaptability and scalability.

AUTHOR CONTRIBUTIONS

Conceptualization – Xinxin Li and Fengna Zhang; methodology – Xinxin Li and Fengna Zhang; investigation – Xinxin Li and Fengna Zhang; writing-original draft preparation – Xinxin Li and Fengna Zhang. All authors have read and agreed to the published version of the manuscript.

TABLE 3. Analysis of Learning Progress (N = 150)

Ability Dimension	Pre-test Average Score	Post-test Average Score	Improvement	Statistical Significance (p-value)
Vocabulary	63.20%	82.70%	19.5%	<0.001
Grammar Mastery	58.70%	76.50%	17.8%	<0.001
Professional Terminology Use	47.30%	78.40%	31.1%	<0.001
Reading Comprehension	61.50%	79.80%	18.3%	<0.001
Reading Speed (wpm)	42.8	67.3	24.5	<0.001
Overall Ability	58.70%	78.90%	20.20%	<0.001

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research received no external funding.

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

This study was approved by the Ethics Committee of Hebei Institute of International Business and Economics, and was performed in accordance with the principles of the Declaration of Helsinki. All eligible participants signed an informed consent form.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] D. Yapp, R. de Graaff, and H. van den Bergh, "Effects of reading strategy instruction in English as a second language on students' academic reading comprehension," Language Teaching Research, vol. 27, no. 6, pp. 1456-1479, 2023, doi: 10.1177/1362168820985236.

[2] R. Ivanova and A. Ivanov, "Online Reading Skills as an Object of Testing in International English Exams (IELTS, TOEFL, CAE)," International Journal of Instruction, vol. 14, no. 4, pp. 713-732, 2021.

[3] E. Bonner, R. Lege, and E. Frazier, "Large Language Model-Based Artificial Intelligence in the Language Classroom: Practical Ideas for Teaching," Teaching English with Technology, vol. 23, no. 1, pp. 23-41, 2023.

[4] M. Zulqarnain and M. Saqlain, "Text readability evaluation in higher education using CNNs," Journal of Industrial Intelligence, vol. 1, no. 3, pp. 184-193, 2023, doi: 10.56578/jii010305.

[5] P. Bannister, A. Santamaría-Urbieta, and E. Alcalde-Peñalver, "A Delphi Study on Generative Artificial Intelligence and English Medium Instruction Assessment: Implications for Social Justice," Iranian Journal of Language Teaching Research, vol. 11, no. 3, pp. 53-80, 2023.

[6] B. Markey, W. Brown D, M. Laudenbach, and A. Kohler, "Dense and disconnected: Analyzing the sedimented style of ChatGPT-generated text at scale," Written Communication, vol. 41, no. 4, pp. 571-600, 2024, doi: 10.1177/07410883241263528.

[7] M. Mohsen, "Artificial intelligence in academic translation: A comparative study of large language models and Google Translate," Psycholinguistics, vol. 35, no. 2, pp. 134-156, 2024, doi: 10.31470/2309-1797-2024-35-2-134-156.

[8] Y. Wang, J. Zhou, Z. Li, S. Zhang, and X. Han, "Automatically difficulty grading method for English reading corpus with multifeature embedding based on a pretrained language model," IEEE Transactions on Learning Technologies, vol. 17, pp. 474-484, 2023, doi: 10.1109/TLT.2023.3319582.

[9] J. H. Lee, D. Shin, and W. Noh, "Artificial intelligence-based content generator technology for young English-as-a-foreign-language learners' reading enjoyment," Relc Journal, vol. 54, no. 2, pp. 508-516, 2023, doi: 10.1177/00336882231165060.

[10] D. Agostini and F. Picasso, "Large language models for sustainable assessment and feedback in higher education: Towards a pedagogical and technological framework," Intelligenza Artificiale, vol. 18, no. 1, pp. 121-138, 2024, doi: 10.3233/IA-240033.

[11] G. Jianan, R. Kehao, and G. Binwei, "Deep learning-based text knowledge classification for whole-process engineering consulting standards," Journal of Engineering Research, vol. 12, no. 2, pp. 61-71, 2024, doi: 10.1016/j.jer.2023.07.011.

[12] H. Majeed and S. Taha, "A comparative analysis of machine and AI translation quality: A case study in Kurdish-English translation," Journal of Kurdistani for Strategic Studies (JKSS), vol. 2, no. 9, pp. 175-189, 2024.

[13] G. Mischler, Y. A. Li, S. Bickel, A. D. Mehta, and N. Mesgarani, "Contextual feature extraction hierarchies converge in large language models and the brain," Nature Machine Intelligence, vol. 6, no. 12, pp. 1467-1477, 2024, doi: 10.1038/s42256-024-00925-4.

[14] G. Venkateshwarlu, G. Sireesha, B. P. Reddy, and C. Yogender, "Transformer-Based Adaptive Examination Systems for Scalable, Accurate, and Personalized Assessments in Education," Frontiers in Collaborative Research, vol. 2, no. 1s, pp. 10-19, 2024.

[15] J. Hao, A. A. von Davier, V. Yaneva, S. Lottridge, M. von Davier, and D. J. Harris, "Transforming assessment: The impacts and implications of large language models and generative AI," Educational Measurement: Issues and Practice, vol. 43, no. 2, pp. 16-29, 2024, doi: 10.1111/emip.12602.

[16] J. Mudkanna Gavhane and R. Pagare, "Artificial intelligence for education and its emphasis on assessment and adversity quotient: A review," Education + Training, vol. 66, no. 6, pp. 609-645, 2024, doi: 10.1108/ET-04-2023-0117.

[17] M. U. Hadi, R. Qureshi, A. Shah, M. Irfan, A. Zafar, and Shaikh MB et al, "Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects," Authorea preprints, vol. 1, no. 3, pp. 1-26, 2023, doi: 10.1007/s10639-022-11254-7.

[18] M. Messer, N. C. C. Brown, M. Kölling, and M. Shi, "Automated grading and feedback tools for programming education: A systematic review," ACM Transactions on Computing Education, vol. 24, no. 1, pp. 1-43, 2024, doi: 10.1145/3636515.

[19] J. Nehyba and M. Štefánik, "Applications of deep language models for reflective writings," Education and Information Technologies, vol. 28, no. 3, pp. 2961-2999, 2023, doi: 10.1007/s10639-022-11254-7.

[20] O. Buinytska, T. Terletska, V. Smirnova, A. Tiutiunnyk, I. Kovalenko, and Hrytseliak, "Artificial intelligence in open university ecosystem context," Informatsiyni texnologii i zacobi navchannya, vol. 105, no. 1, pp. 204-221, 2025, [Online]. Available: https://elibrary.kubg.edu.ua/id/eprint/51288.

[21] C. L. Tsang and T. Isaacs, "Hong Kong secondary students' perspectives on selecting test difficulty level and learner washback: Effects of a graded approach to assessment," Language Testing, vol. 39, no. 2, pp. 212-238, 2022, doi: 10.1177/02655322211050600.

[22] A. A. Sanabria, M. A. Restrepo, E. Walker, and A. Glenberg, "A reading comprehension intervention for dual language learners with weak language and reading skills," Journal of Speech, Language, and Hearing Research, vol. 65, no. 2, pp. 738-759, 2022, doi: 10.1044/2021_JSLHR-21-00266.

[23] X. Tong, L. Yu, and S. H. Deacon, "A meta-analysis of the relation between syntactic skills and reading comprehension: A cross-linguistic and developmental investigation," Review of Educational Research, vol. 95, no. 3, pp. 385-426, 2025, doi: 10.3102/00346543241228185.

[24] K. Landerl, A. Castles, and R. Parrila, "Cognitive precursors of reading: A cross-linguistic perspective," Scientific Studies of Reading, vol. 26, no. 2, pp. 111-124, 2022, doi: 10.1080/10888438.2021.1983820.