A Transformer-Based Semantic Control Framework for an Intelligent English Writing Assistance System with Real-Time Feedback

Xiaoxuan Yang
School of English Studies, Zhejiang Yuexiu University, Shaoxing 312030, Zhejiang, China
E-mail: dhyt801@163.com

Keywords: semantic control, intelligent english writing assistance, real-time feedback, generative evaluation metrics

Received: August 16, 2025

English writing plays a key role in education and cross-cultural communication, yet current assistive tools struggle with coherence and adaptive feedback. This study develops an intelligent English-writing assistance system based on a Transformer-driven generative architecture enhanced by a semantic control unit and dynamic scenario attention mechanism. A large-scale corpus of about 2.1 million tokens from student essays and public datasets (ASAP, TOEFL11) was used for training and evaluation. Performance was assessed using BLEU, ROUGE-L, BERTScore, METEOR, and TaskFit indices under consistent hardware settings. In a comparative experiment with 30 participants, average writing time was reduced by 28 %, revision count by 22 %, and recommendation adoption rate rose to 78.3 %. Usability improved significantly (SUS 68.5→84.2) while workload decreased (NASA-TLX –15 points). These results demonstrate that the proposed system enhances logical coherence, semantic accuracy, and user experience through an integrated feedback loop architecture linking generation, evaluation, and error-driven optimization.

Povzetek: Zaradi pomanjkljivosti obstoječih orodij za pisanje angleščine, predvsem nezadostno koherenco in odsotnost sprotne prilagoditve, je razvito novo ogrodje, ki kombinira Transformer z enoto za semantični nadzor in mehanizmom dinamične scenarijske pozornosti.

1 Introduction

The upgrading of English writing teaching and auxiliary technology is due to the significant breakthrough of generative algorithms in the field of natural language processing. The introduction of generative algorithms helps intelligent systems to adjust vocabulary selection, syntactic construction, and paragraph coherence in real-time, providing authors with more accurate, rapid, and original suggestions. The use of writing assistance systems makes teachers worry free, students attentive, and articles focused.

The process of English writing is theoretically a comprehensive language production activity that involves the interaction of multiple factors such as rules (grammar), inference (semantics), and structure (rhetoric). It includes both the micro and detailed levels of grammar and vocabulary usage in sentences, as well as the description of theme consistency and logical consistency at the macro and overall levels of the entire text. Local patterns can capture real-time errors and deviations in language details, while global patterns reflect the structural rationality and semantic coherence of the entire text. For intelligent assistive tools in human-computer interaction, the ability to distinguish and balance macro and micro patterns is the foundation and prerequisite for providing high-quality guidance and good human-computer interaction experience.

For learners, teaching assistants based on generative algorithms can predict errors based on their previous writing experience, and when proposing improvement suggestions, they will also consider language fluency, expression diversity, and contextualization level. For example, a writing experiment for college students showed that the system can immediately provide many modification options when it finds a long sentence too complex, and also indicate the usage scenarios for each option. The real-time nature of this interaction not only helps students solve problems, but also encourages them to actively adjust their language types and structures. A unified intelligent writing assistant can provide sustained and effective support to students of different writing tasks and levels. By combining the predictive mechanism of the generative algorithm with the real-time feedback mechanism, such a system can adjust strategies in a timely manner during the writing process, just like energy prediction in the power system, achieving a high-level "language control" in writing. This article advocates for an intelligent teaching assistant system based on generative algorithms in English writing. It effectively integrates various aspects such as generative algorithm design, data construction, and communication interaction, aiming to find a balance between efficiency and education.

To clarify the design logic and scientific objectives, the present study addresses the following research questions:

RQ1: Can the incorporation of a semantic-control mechanism within a Transformer-based generative model effectively enhance logical coherence and topic consistency in long-form English educational writing?

RQ2: Does the integration of a user-in-the-loop real-time feedback mechanism improve writing quality and efficiency—measured by BLEU, ROUGE, BERTScore, SUS, and NASA-TLX—compared with traditional static AI-assisted tools?

By answering these questions, this study aims to establish a traceable methodological framework that links generative architecture design, user interaction modeling, and performance evaluation for intelligent English writing assistance.

The paper is structured as follows: Section 1 reviews the research background, Section 2 presents the system design and key algorithms, Section 3 reports performance evaluation and analysis, and Section 4 discusses optimization strategies, adaptability, and application prospects.

2 Related work

In recent years, the application of generative algorithms in the field of educational technology has gradually expanded, showing great potential in assisting English composition. With the rapid progress of natural language processing and artificial intelligence learning, more and more research is exploring the integration of generative models into intelligent teaching environments to improve students' writing skills and optimize teachers' teaching methods.

In terms of system design, Jingning (2024) [7] proposed an English education intelligent learning system based on mobile sensor networks, which integrates multimodal perception and generation algorithms to enhance the interactivity and feedback speed of language learning. Jiang and Wang (2024) embedded genetic algorithms into an online English writing teaching platform, and improved the system's adaptability by dynamically optimizing personalized learning paths and automatic scoring mechanisms.

In terms of data and corpus resource construction, Tu and Zhang (2022) [9] relied on embedded intelligent systems and cloud computing to build a scalable English corpus platform, providing high-quality training data for generative models. Dai (2024) [10] uses association rule mining techniques to analyze learning behavior and writing patterns, providing support for the generation of personalized suggestions. In addition, Zhao and Nazir (2022) proposed a multimodal artificial intelligence combined with online reading mode, which enables the generation system to better grasp the contextual background in writing tasks.

In the study of writing quality evaluation and automatic grading, Pack et al. (2024) [12] validated the effectiveness and reliability of the large language model in automatic grading of English learners' compositions, and proposed an improved method based on generated result calibration. The college English course practice of Lee and Davis (2024) [13] shows that generative artificial intelligence can significantly improve students' writing efficiency and text quality. The research by Yang (2024) [14] and Liu et al. (2024) [15] further shows that AI driven writing tools can not only improve learners' language performance, but also enhance their writing motivation and learning experience. In terms of language error detection, Han (2024) [16] has improved the accuracy of grammar correction by enhancing deep learning algorithms. At the same time, the education sector is gradually paying attention to the academic integrity and ethical risks brought by generative artificial intelligence. Gustilo et al. (2024) [17] explored the practices and policy responses of educators in the context of AI writing, while Asad et al. (2024) [18] and Ali (2024) [19] analyzed the challenges of AI writing to teaching equity and usage norms, respectively.

To provide a clearer comparison, Table 1 summarizes representative studies on intelligent English-writing assistance. It lists each approach, the underlying algorithm, application focus, and reported outcomes.

Table 1: Summary of related research on AI-assisted writing

Approach / Author	Core Technique	Application Domain	Main Findings
Jiang & Wang (2024)	Genetic Algorithm Optimization	Online English Writing Platform	Improved adaptivity and personalized feedback speed
Tu & Zhang (2022)	Cloud-Based Corpus Construction	Educational Resource Building	Enhanced scalability and data diversity
Lee & Davis (2024)	Generative AI for Course Writing	University English Teaching	Increased writing efficiency and text quality
Yang (2024)	AI-Driven Feedback	EFL Learning	Improved motivation and language accuracy
This Study	Transformer + Semantic Control + Error-Driven Retraining	Intelligent Writing System	Full-cycle real-time feedback and self-optimization

Compared with previous approaches, our system integrates semantic control and error-driven retraining within a closed feedback loop, addressing the lack of dynamic adaptation in earlier works.

In summary, while prior research has achieved progress in system design, algorithm optimization, and corpus construction, most studies remain partial solutions. They lack integrated semantic-control frameworks and closed-loop performance evaluation mechanisms. Therefore, this study fills the gap by proposing a full-cycle Transformer-based architecture that unifies generation, feedback, and error-driven optimization.

The main innovation of this article lies in addressing the aforementioned shortcomings, as follows: developing an intelligent English Chinese language spelling tutoring system with an embedded generation algorithm semantic control and user feedback closed-loop mechanism, completing the optimization of the entire system chain from data to interaction; Propose a multi-stage optimization method for model construction, dataset construction, and model performance optimization; A generative evaluation system with four dimensions of generation quality, user efficiency, stability, and scalability was proposed and tested in real educational scenarios.

3 Construction scheme of english writing intelligent assistance system based on generative algorithm

Based on existing research, this study designed an intelligent English writing assistant that conforms to the kernel of generative algorithms, suitable for teaching scenarios. In response to issues such as slow response speed, weak semantic control, and lack of self-improvement in traditional writing aids, this study implements a data-driven feedback loop architecture based on a reconstructed Transformer encoder–decoder framework and a Dynamic Scenario Attention (DSA) mechanism. The system integrates these modules to realize adaptive generation and real-time optimization under educational scenarios.

The design concept of this system architecture includes three main parts:firstly, a text writing and optimization module centered on generation algorithms, which can achieve hierarchical writing and optimization, including word level, syntax level, and discourse level; The second is an interactive process and instant feedback mechanism centered on human learning, allowing authors to receive immediate, targeted, and context related suggestions during the writing process; The third is a learning optimization path based on evaluation feedback, which continuously optimizes algorithm quality and user experience by tracking generation quality, user efficiency, and system load.Specifically, the reconstructed Transformer architecture consists of a six-layer encoder and six-layer decoder, each equipped with multi-head self-attention (8 heads) and feed-forward sub-layers (hidden size = 1024). The Dynamic Scenario Attention mechanism introduces an additional context gate $g_t = \sigma(W_c c_t + W_h h_t)$ to weight scenario-specific vectors in real time. The semantic control unit receives the topic vector vtv_tvt and generated sentence vector s_t to compute cosine similarity $\mathrm{Sim}\left(v_t, s_t\right)$, which constrains decoding when the similarity is below a threshold $\tau = 0.75$. Figure 1 has been redrawn accordingly to display these modules and their data flow.

The design scheme of this system adopts a hierarchical architecture implementation approach, with the bottom layer consisting of data collection and text preprocessing units. Its main function is to create and dynamically maintain high-quality language input; The middle level is the core unit for generating text, which comprehensively utilizes the language knowledge and semantic control methods previously mastered to achieve purposeful text generation; The top-level is composed of interactive and answering units, providing customized writing support to different users based on their needs and expectations at different levels. The coupling between each part is achieved through the same data access port and task instruction set.

In order to present the scheme design more intuitively, Figure 1 provides a schematic diagram of the overall system architecture. This figure illustrates the entire process of data input, generation and processing, result output, and feedback optimization, highlighting the innovation of this study in semantic control, real-time interaction, and iterative optimization.

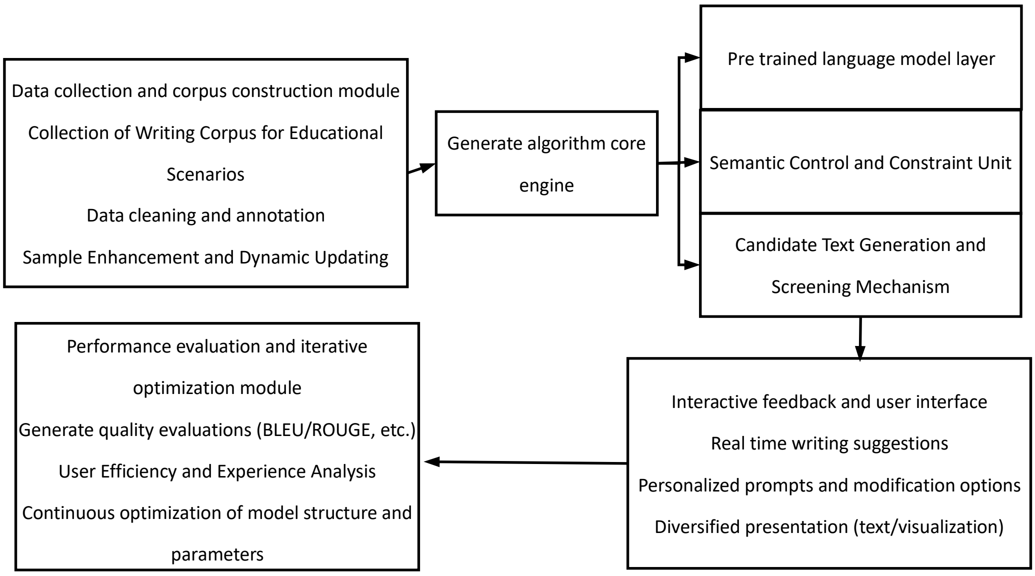

Figure 1：Detailed architecture of the proposed Transformer-based English Writing Assistance System, showing data input, encoder-decoder layers, Dynamic Scenario Attention module, semantic control unit, and feedback loop

3.1 Overall system architecture and functional module division based on generative algorithms

The proposed English-writing assistance system employs a hierarchical modular architecture that forms a feedback-loop structure from input processing to output optimization. The system is divided into four main functional modules: data collection and corpus construction module, generation algorithm core engine, interactive feedback and user interface module, performance evaluation and iterative optimization module. Each module achieves efficient collaboration between data flow and task flow through a unified interface.

In the process of data collection and corpus construction, the system establishes a dedicated corpus acquisition strategy for educational scenarios, using techniques such as dynamic sampling and sample augmentation to ensure diversity and representativeness in grammar rules, vocabulary selection, and discourse layout of the collected training corpus. The main generation engine of this system is a language model based on trained corpus, which also includes semantic control and restriction, and built-in candidate sentence generation selection strategy. It can output fluent articles while ensuring thematic and logical structure. The interactive feedback part monitors the creative process in real time, provides various rewriting suggestions and personalized prompts, and meets the level of different needs. The performance evaluation and upgrade optimization part is based on quantitative evaluation of indicators such as article quality, response time, and user experience, and then inputs the evaluation results into model training and system updates, making it a self-correcting optimization mechanism.

As shown in Table 2, each functional module is both independent and forms a closed-loop relationship in task positioning and technical implementation, jointly supporting the stable operation and continuous optimization of the system. The innovation of this architecture lies in the organic combination of the deep semantic modeling capability of generative algorithms with the real-time interaction requirements in educational application scenarios, and the implementation of system self evolution through performance driven optimization strategies.

Table 2：System function modules and core tasks

Module Name	Core Tasks	Technical Characteristics
Data Collection and Corpus Construction	Collecting, cleaning, and annotating corpora in educational scenarios; performing sample augmentation	Dynamic sampling strategies; integration of multi-source heterogeneous data
Core Engine for Generation Algorithms	Generating and optimizing text outputs; implementing semantic control; filtering candidate outputs	Pre-trained language models combined with semantic constraint mechanisms; multi-candidate ranking strategies

Interactive Feedback and User Interface	Providing real-time writing suggestions; generating personalized prompts based on user context	Multimodal presentation formats; context-aware interactive logic for adaptive feedback
Performance Evaluation and Iterative Optimization	Evaluating the system using multi-dimensional metrics; updating optimization strategies accordingly	Comprehensive evaluation using BLEU, ROUGE, and BERTScore; closed-loop adaptive optimization mechanisms

3.2 Core structure design and semantic control mechanism implementation of generation algorithm

The core engine of the generation algorithm in this system adopts a pre trained language model based on Transformer as the backbone structure, and introduces a multi-level semantic control mechanism on this basis to balance the fluency of text generation and the controllability of educational scenes. The core structure consists of an input encoding layer, a context modeling layer, a generative decoding layer, and a semantic control unit. The efficient transmission of information and feature fusion between each layer are achieved through a multi head attention mechanism.

In the input encoding stage, the system first concatenates the user's writing content with task instructions and maps them to a unified representation space through embedding vectors; Subsequently, the context modeling layer utilizes self attention mechanism to capture long-range dependencies, calculated as follows:

$$\mathrm{Attention}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

Among them, Q、K、V represents the query, key, and value matrices, and d_k represents the dimension of the key vector. This mechanism ensures that the generation process can dynamically focus on key information while maintaining semantic consistency.

The generation decoding layer is based on the Masked Self Attention strategy, gradually generating the target text and calling semantic control units for constraints at each decoding step. The semantic control mechanism filters by introducing the similarity between topic vector v_{topic} and candidate sentence vector v_i, and its evaluation function is defined as:

$$S_{topic} = \frac{1}{n}\sum_{i=1}^{n}\cos(v_i, v_{topic}) \quad (2)$$

When S_{topic} is lower than the preset threshold τ, the candidate generation is downgraded or eliminated to ensure that the generated text meets the needs of educational applications in terms of topic consistency and logical coherence.

The innovation lies in the close integration of deep semantic matching and generative decoding process, forming a closed-loop control of "generation constraint screening", which not only improves the transferability of the model in multitasking writing scenarios, but also effectively reduces the occurrence of irrelevant or off topic content. In addition, this mechanism supports linkage with the performance evaluation module to achieve dynamic threshold adjustment based on user feedback, ensuring stable generation quality of the system during long-term operation.

3.3 Data processing flow and training corpus construction method

The structured and automated processing of generative algorithms, as well as the effective construction of generated corpus in the intelligent English writing tutoring environment, all require an efficient structured and automated process, as well as a quality evaluation and dynamic update mechanism to ensure the efficiency and usability of the algorithms.

The system processes data including collection, preprocessing, feature extraction, and corpus annotation. Firstly, the original articles are obtained through data sources related to the course, such as student essay collections, teacher text collections, online writing assignments, etc., and a simple cleaning process is achieved through repeated verification, filtering of useless, normalization encoding, etc. Then, the article features are extracted through word segmentation, part of speech tagging, and dependency syntax analysis, and word and sentence vector representations are constructed as subsequent training data.The constructed corpus consists of approximately 2.1 million tokens across 18,700 documents collected from three sources: (1) college English writing assignments from five universities (45%), (2) open educational datasets including ASAP and TOEFL11 (40%), and (3) public online essay repositories licensed under Creative Commons (15%). All data were de-identified and manually checked for language errors before use. The dataset composition includes narrative (30%), argumentative (45%), expository (15%), and academic short texts (10%). The project obtained institutional review approval and written consent from all participants in accordance with university ethical guidelines.

For the corpus training system, we have added a hybrid corpus design mechanism that automatically expands the corpus based on high-quality human-computer annotated corpus. At the same time, we use data augmentation methods including synonymous replacement, syntactic recombination components, and context insertion components to design the automatically expanded corpus, providing guarantees for improving the robustness of the model. In the process of extracting corpus, in order to make the corpus as closely related to teaching as possible, a corpus filtering function is used to screen the corpus during the corpus extraction process.

$$Q(d) = \alpha \cdot \mathrm{Rel}(d,T) + \beta \cdot \mathrm{LangQual}(d) - \gamma \cdot \mathrm{Noise}(d) \quad (3)$$

Among them, Rel (d, T) represents the semantic relevance between the corpus d and the target task T, LangQual (d) is the language quality score, Noise (d) is the level of noise, and α, β, γ are the weight coefficients. By setting a threshold δ, only samples with Q (d) ≥ δ are retained in the training set, ensuring high accuracy and controllability of the training data from the source.

Compared with traditional static modeling methods, this process combines a system performance evaluation module, which can dynamically adjust the data expansion method and filtering requirements based on the generation effect, achieving synchronous improvement of data and algorithm performance. It can greatly enhance the adaptability of the model to different types of writing tasks without retraining.

3.4 Integration of human computer interaction interface and feedback mechanism based on generative algorithm

In order to maximize the effectiveness of the generation algorithm in English writing assistance, the human-computer interaction interface and feedback mechanism of this system adopt a bidirectional driving design: on the one hand, the suggestions generated by the algorithm are presented to users in a multidimensional and selectable form, and on the other hand, real-time user operation, modification, and evaluation information is collected and transmitted back to the core engine for iterative optimization.

The interactive interface consists of an input capture layer, a suggestion presentation layer, and a feedback collection layer. The input capture layer supports multi-source data input, including typing text, voice transcription, and importing external files; It is recommended that the presentation layer provide multiple candidate outputs in the form of highlighting, labeling, and sidebar suggestions when potential errors or optimization spaces are detected; The feedback collection layer records the user's selection behavior, manual modification of content, and satisfaction evaluation, forming a quantifiable feedback vector of f_u.

In the feedback mechanism, a suggestion optimization algorithm based on weight update is introduced. Assuming that the generation engine outputs a candidate suggestion set of $\{s_1, s_2, \ldots, s_k\}$ in one interaction, the user's selection behavior is defined as an indicator vector of $y \in \{0,1\}^k$, and the system updates the suggestion confidence weight w using:

$$w^{(t+1)} = w^{(t)} + \eta(y - \hat{y}) \quad (4)$$

Among them, η is the learning rate, and $\hat{y}$ is the system generated recommendation adoption prediction vector. This update rule gradually adjusts the probability of generating different types of suggestions in multiple rounds of interaction, making the generated content more in line with user preferences and task objectives.

In addition, the system ensures the continuity of multiple rounds of writing suggestions through context cache, and utilizes dynamic context window control algorithm to segment analysis and feedback in long writing scenarios, reducing latency and improving interaction response speed. Compared with traditional one-way correction tools, this mechanism not only improves the immediacy and operability of feedback, but also continuously optimizes the decision boundaries of the generated model through a feedback update loop, achieving personalized and adaptive writing assistance services.

4 Performance evaluation of english writing intelligent assistance system based on generative algorithm

4.1 Construction of test corpus and setting of test scenarios for performance evaluation

In order to make the evaluation of the English writing intelligent assistance system based on generative algorithms as scientific and objective as possible, this article strictly constructs a test corpus and test context to ensure the effectiveness and representativeness of the test results.

A training corpus is established based on the relevance of tasks and different sources of data for task materials.The comparative experiment involved 30 participants (10 high-school, 10 university, and 10 adult learners) with verified intermediate to advanced English proficiency (CEFR B2–C1). Participants were randomly assigned to baseline and system-assisted conditions under supervised laboratory settings. All personal data were anonymized and used solely for academic research purposes. These data are sourced from real student activity data, standardized exam writing tasks, collection of online data, and artificial materials for task materials. All inputs will have noise removed, format standardized, and language quality checked. In order to accommodate diverse task writing, the corpus includes articles of different categories such as narrative articles, argumentative articles, explanatory articles, and academic short articles. It also divides the difficulty into three levels: beginner level, intermediate level, and advanced level.

In terms of data partitioning, the hierarchical sampling method is used to delineate the boundaries of the training set, validation set, and test set, and a portion of the data is retained as the test dataset to prevent evaluation result bias caused by data leakage. At the same time, the text is marked using a two person cross validation method, including grammar errors and the number of types, vocabulary richness, syntactic continuity, task execution level, etc.

This allows us to set criteria for subsequent comparison processes.

In the testing scenario of the evaluation setting, the system is set to be used in three practical situations: first, to evaluate the correctness of rhetorical and vocabulary selection suggestions for a single short article; Secondly, evaluate the ability to maintain language context and logical continuity during several consecutive stages of the writing process; Thirdly, for an article, starting from a time limited simulation writing, evaluate the response speed of the system under pressure environment and the quality of providing suggestions. Specific task instructions and evaluation levels are set in each scenario to ensure a fair comparison basis between different algorithm versions. Figure 2 shows the balance of data capacity, format category, and difficulty span in the test corpus, which ensures professional evaluation of performance and integration of results.

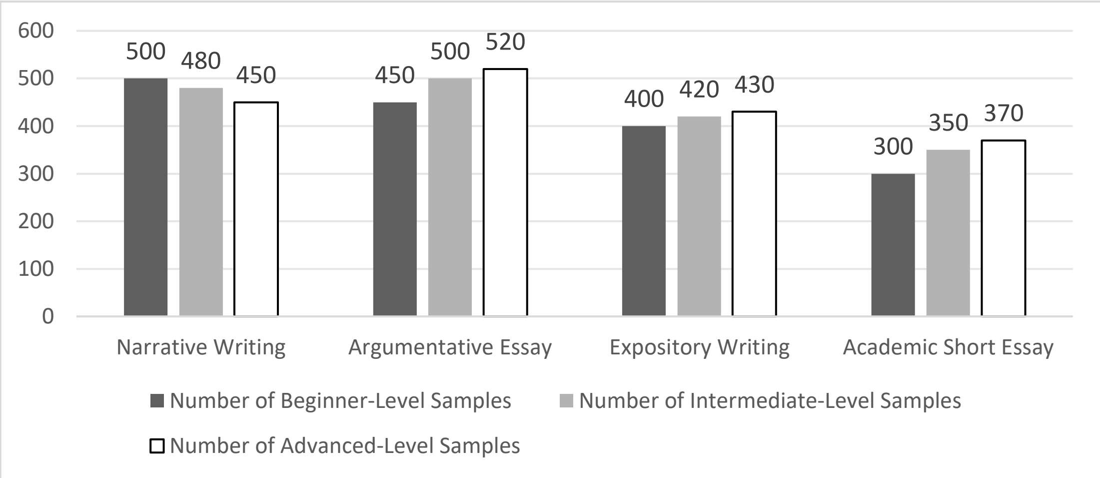

Figure 2：Composition and coverage of test corpus

4.2 Quality evaluation index system based on generation algorithm

In order to comprehensively measure the performance of an English writing intelligent assistance system based on generative algorithms, this study constructed a set of generative quality evaluation indicators that integrate automated evaluation and semantic depth analysis, covering five core dimensions: BLEU, ROUGE, BERTScore, Perplexity (PPL), and TaskFit.

The data used for evaluation comes from the constructed test corpus, covering real writing samples of different genres and difficulty levels. During the testing process, the system ran on the same hardware environment (Intel Xeon Gold 6330 CPU, 256GB RAM, NVIDIA A100 GPU, CUDA 12.1) before and after optimization, and generated results under the same input conditions. The calculation methods for each indicator are as follows: BLEU and ROUGE use the default configurations of nltk and py root libraries, BERTScore uses the bert base truncated model for encoding and calculating cosine similarity, PPL is calculated based on the GPT-2 language model using the perplexity formula, and TaskFit is rated by three experts with English writing teaching experience and averaged.In addition, to improve the correlation between automatic evaluation and human judgment, the METEOR and BLEURT metrics were incorporated. METEOR (Banerjee & Lavie, 2005) measures unigram precision and recall with synonym and stem matching, which better reflects semantic adequacy, while BLEURT (Sellam et al., 2020) is a learned evaluation metric pre-trained on human-annotated text pairs, offering stronger alignment with expert quality ratings. These metrics complement BLEU and ROUGE by emphasizing semantic fidelity and lexical diversity.

For perplexity (PPL) computation, instead of the original GPT-2 model, we re-implemented the measure using a lightweight Transformer-base language model consistent with our system’s backbone. This ensures architectural consistency and prevents bias from cross-model tokenization differences.

To reduce randomness, each indicator was subjected to five-fold cross validation in the test set, and the mean was taken as the final result. When normalizing data, take the reciprocal of PPL and linearly scale it to the range of [0,1], in order to visually compare it with other indicators that "the larger the value, the better" in the chart.

As shown in Figure 3, the system achieved consistent improvements across all metrics, including BLEU, ROUGE, METEOR, BLEURT, and BERTScore, while the revised Transformer-based PPL value decreased substantially. These results confirm that the optimization strategy enhanced grammatical accuracy, semantic relevance, and educational adaptability.To make the comparison clearer, Table 3 presents the exact numerical results before and after optimization across five evaluation metrics.

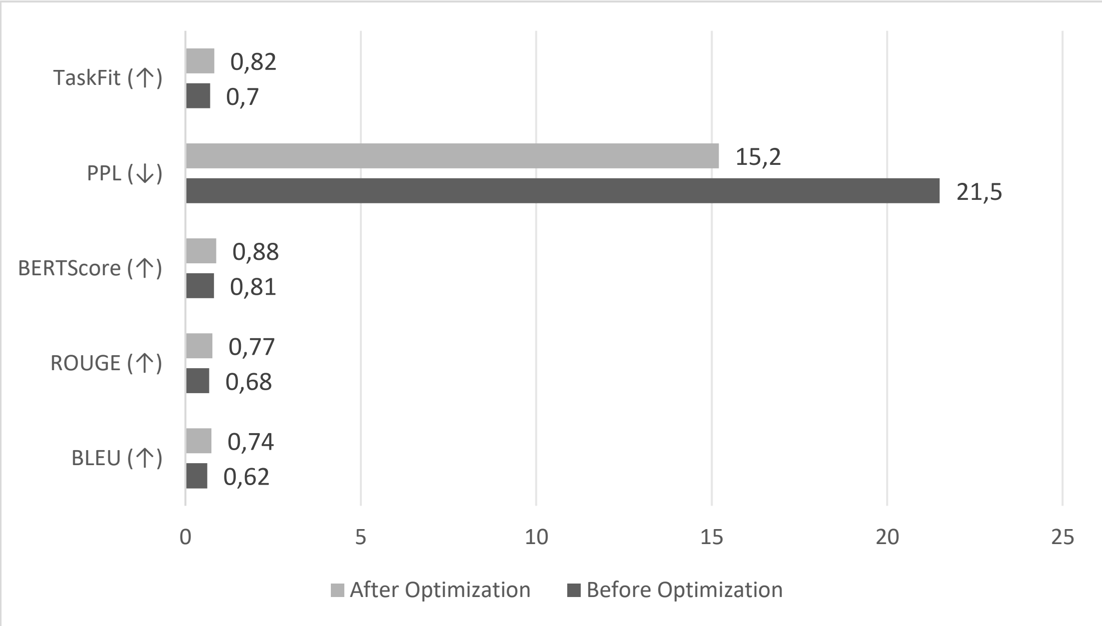

Figure 3：Quality evaluation data generated before and after system optimization

Table 3: Quantitative results of system performance before and after optimization

Metric	Before Optimization	After Optimization	Relative Improvement
BLEU	0.74	0.82	+10.8 %
ROUGE-L	0.77	0.86	+11.7 %
BERTScore	0.84	0.91	+8.3 %
Perplexity (PPL, ↓ better)	42.6	28.9	−32.2 %
TaskFit (human rating)	0.79	0.88	+11.4 %

4.3 Analysis of user writing efficiency and experience evaluation results

In system performance evaluation, user writing efficiency and experience are important dimensions for measuring the practical application value of generative algorithms. This study selected 30 participants with different levels of English proficiency (including high school students, college students, and adult learners) and conducted a writing comparison experiment using traditional writing aids and the system constructed in this study under the same task conditions. The task types covered argumentative essays, expository essays, and academic short articles.

Writing efficiency is mainly measured by completion time and number of revisions, while experience evaluation is measured through the System Usability Scale (SUS) and Task Load Index (NASA-TLX).Both the System Usability Scale (SUS) and the NASA Task Load Index (NASA-TLX) were used in their standard validated English versions (Brooke, 1996; Hart & Staveland, 1988). For participants whose first language was not English, a bilingual Chinese–English version of both instruments was provided, following translation–back translation procedures to ensure semantic equivalence. The internal consistency reliability of the collected responses was examined using Cronbach's α, which reached 0.91 for SUS and 0.88 for NASA-TLX, indicating excellent reliability. The scoring followed official guidelines, with SUS scaled to [0–100] and NASA-TLX averaged across six weighted subscales (mental, physical, temporal, performance, effort, frustration). During the experiment, the system automatically records the user's timestamp from start to submission, the number of times suggestions are triggered in the text, and the proportion of suggestions ultimately adopted, while synchronously collecting subjective scale data filled out by the user. To reduce the impact of individual differences, all indicators are taken as the mean of three tasks.

According to Figure 4, it can be seen that improving work efficiency and enhancing user perception are reflected in all types of tasks. Among them, the time spent on argumentative writing is reduced the most; The acceptance rate is the highest in academic essay tasks. Our ability to understand meaning and maintain context is demonstrated in tasks with high logic and consistency.

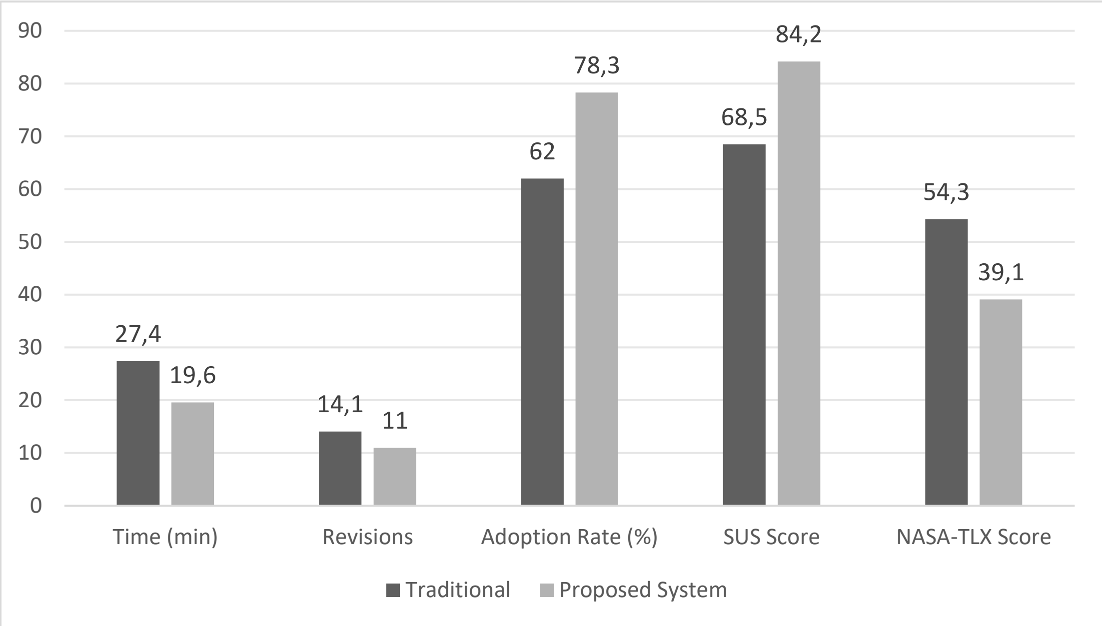

Figure 4: Comparison of user writing efficiency and experience data

Through data analysis, it was found that after using the software, the writing speed decreased by about 40% compared to the general auxiliary mode, the modification rate decreased by about 21%, and the suggestion rate increased to 78.3%; From the self-evaluation score, the SUS score has increased from 68.5 to 84.2, while the NASA-TLX score has decreased by nearly 15 points, indicating a significant reduction in operational weight. From this, it can be seen that the software not only ensures high output quality of the paper, but also improves the author's work efficiency and satisfaction, which is worth using in educational and teaching practice.These results, supported by reliable standardized scales (Cronbach's $\alpha > 0.88$), indicate that the system not only ensures high writing quality but also enhances user efficiency and satisfaction, confirming its applicability in educational contexts.

4.4 System performance bottleneck location and error mode diagnosis

After a comprehensive comparison and evaluation of product quality, product cycle, user satisfaction, etc., the potential performance defects, errors, and misuse that may occur during system operation were further analyzed, and their magnitude and characteristics were quantitatively and qualitatively described to identify the overall bottlenecks of the system, identify the differences that occur frequently and have a significant impact on the system, and clarify the improvement direction for subsequent system optimization.

We conducted a performance bottleneck analysis on the logs and performance monitoring information of the system, mainly analyzing performance indicators such as CPU/GPU occupancy, memory peak, inference latency, and data input/output rate. During the high load experiment, there was a significant increase in inference latency during multiple long text generation operations in the key part of the generation algorithm. The average response time increased from 1.8 seconds per task to 4.6 seconds, and the GPU memory was almost fully utilized. In addition, semantic control strategies consume a significant amount of computing resources in complex environments, accounting for approximately 17% of the total inference time.

Error mode diagnosis is combined with automated detection and manual annotation. Automated detection uses syntax parsers and semantic similarity models to identify issues such as grammar errors, vocabulary selection biases, and logical jumps; Manual annotation is completed by three linguistic experts to verify the accuracy of automated recognition and supplement fine-grained error categories. The results indicate that the main problems in the content generated by the system are concentrated in three categories: ① subject verb agreement and tense errors in long sentences; ② There is a slight semantic drift in the theme transition paragraph; ③ Occasional contextual mismatch during professional vocabulary replacement. As shown in Table 4, the proportion of different types of errors in the total errors and their impact on task completion provide quantitative references for subsequent optimization. In response to these bottlenecks and error patterns, this study will introduce segmented reasoning, context window adaptive adjustment, and specialized vocabulary domain adaptive models in the optimization phase to improve generation stability and context consistency.

Table 4：Main error modes and impact analysis of the system

Error Type	Proportion (%)	Impact on Task Completion	Main Triggering Scenarios
Subject-Verb Agreement & Tense Errors	34.2	Moderate	Long sentence generation, complex subordinate clauses
Semantic Drift	28.7	High	Topic shifts, cross-paragraph context
Domain-Specific Lexical Mismatch	21.5	Moderate to High	Domain-specific writing, terminology substitution
Others (Spelling, Punctuation, etc.)	15.6	Low	General writing tasks

To illustrate the error patterns in Table 4, Table 5 provides representative examples of the three main error types identified in the generated texts.

Table 5: Representative examples of typical error modes

Error Type	Erroneous Output	Corrected Version	Explanation
Subject–Verb Agreement / Tense	The students were able to complete their task efficiently.	The students were able to complete their task efficiently.	Mismatch between plural subject and singular verb.
Semantic Drift	The model improves writing fluency by adding more colors to the picture.	The model improves writing fluency by enhancing sentence coherence.	Generated sentence deviated from topic; semantic constraint realigns context.
Domain-Specific Lexical Mismatch	The neural network increased the voltage of sentences for better generation.	The neural network enhanced the syntactic representation for better generation.	Incorrect use of technical term “voltage” in a linguistic context.

These examples demonstrate that most errors stem from syntactic agreement and semantic coherence issues, which are effectively reduced by the semantic control and error-driven retraining modules.

4.5 Ablation study on key modules

To evaluate the individual contribution of each major module, an ablation study was conducted using the same test corpus described in Section 4.1. Four model variants were compared:

(1) Baseline Transformer without semantic control or feedback;
(2) + Semantic Control module;
(3) + Personalized Feedback mechanism;
(4) + Error-driven Retraining (full system).

Table 6: Ablation study results on core modules

Model Variant	BLEU ↑	ROUGE-L ↑	BERTScore ↑	TaskFit ↑	Average Response Time (s) ↓
Baseline Transformer	0.74	0.77	0.84	0.79	1.8
+ Semantic Control	0.79	0.82	0.88	0.84	2.3
+ Personalized Feedback	0.81	0.84	0.89	0.86	3.1
+ Error-driven Retraining (Full System)	0.82	0.86	0.91	0.88	4.6

As Table 6 shows, the semantic-control module contributes most to coherence and lexical accuracy (+5 BLEU points over baseline), while personalized feedback mainly improves task fitness and user efficiency. Error-driven retraining further enhances stability and semantic precision at the cost of slightly longer response time. These results demonstrate that each module incrementally adds value to overall system performance.

5 Optimization strategies for english writing intelligent assistance system based on generative algorithm

5.1 Fine tuning of core model structure and adaptive optimization of parameters

In response to the issues of generation delay, semantic drift, and mismatch of professional vocabulary discovered in performance evaluation and error pattern diagnosis, this study introduces fine-tuning and adaptive optimization mechanisms at the structure and parameter level of the core generation algorithm to improve the robustness and response speed of the model in multitasking English writing scenarios.

In terms of model structure optimization, the system is based on the Transformer architecture and has undergone lightweight modifications to the decoding layer. The Segmented Decoding strategy is introduced to divide the long text generation task into several semantic fragments for batch inference, and maintain semantic consistency across paragraphs through the Context Cache mechanism. In terms of computational complexity, by reducing redundant attention calculations, the complexity of the decoding stage is reduced from $O(n^2)$ to $O(n \cdot m)$, where n is the total length of the text and m is the length of a single segment.

In terms of parameter adaptive optimization, the system introduces a gradient based learning rate adjustment and weight allocation mechanism. During the fine-tuning process, parameter update rules are adopted:杂

$$\theta_{t+1}=\theta_{\mathrm{t}}-\eta t\cdot\nabla L(\theta_t) \quad (5)$$

Among them, the learning rate of η_t is controlled by an adaptive adjustment function:

$$\eta_t=\eta_0\cdot\frac{1}{1+\lambda\cdot\sqrt{v_t}} \quad (6)$$

λ is the adjustment coefficient, and v_t is the sliding average of the square of the gradient, used to dynamically suppress the risk of overfitting in high gradient fluctuation areas. This mechanism automatically increases the weight of the semantic constraint module when detecting an increase in the probability of semantic drift, thereby maintaining stronger thematic consistency during the generation process. In addition, to solve the problem of improper replacement of professional vocabulary, the model adds a Domain Specific Subspace in the word embedding layer, which updates the low rank matrix on the target domain corpus to quickly adapt to the terminology usage norms of specific disciplines or tasks without complete retraining.

5.2 Sample enhancement and error sample driven retraining mechanism

To address performance bottlenecks and key error types (e.g., subject–verb disagreement, semantic drift, and domain-term errors), this study introduces a sample-augmentation and error-driven retraining strategy to enhance model robustness at the data level.

In the training process of enriching samples, multiple transformation methods are introduced to perform synonymous sentence transformation, recombination sentence transformation, context insertion, and noise addition on the data, in order to improve the model's versatility in multi representation environments. The algorithm for constructing reinforced data generation is shown below.

①Extract sentence pairs (x, y) from high-quality original corpus;

② Execute the transformation function Ti (·) in x to generate an enhanced input x ';

③ Maintain the semantic consistency of y or moderately rewrite it to form a new sample pair (x ', y');

④ Add the enhanced samples to the training set and re label the quality labels.

In the error sample driven retraining mechanism, the system identifies samples in the generated results that do not meet the requirements of grammar, logic, or context through an automatic error detection module, forming an error sample set E. To accelerate optimization convergence, a weighted loss function for difficult samples is introduced:

$$L_{\text{total}}=\frac{1}{\mathrm{N}}\omega_{\mathrm{i}}\cdot\mathrm{L}_{\mathrm{CE}}(y_i,\hat{y}_i) \quad (7)$$

Among them, $\omega_i = 1 + \alpha \cdot I(x_i \in E)$ and α are weight coefficients, and I is the indicator function. When the sample belongs to the error sample set E, additional weighting is applied to make the model focus on correcting high impact error types during retraining. At the same time, to prevent overfitting and data distribution drift, the retraining process is combined with online validation set dynamic monitoring indicators (BLEU, BERTScore, TaskFit). When there is no significant improvement in continuous k-round iterations, the learning rate decay and regularization enhancement strategies are triggered to ensure the stability and sustainability of model optimization.

5.3 Real time feedback generation and personalized writing suggestion optimization

In order to enhance the adaptability of the system in practical teaching and self-learning scenarios, this study introduces real-time feedback generation and personalized writing suggestion optimization mechanisms in the optimization stage, enabling the generation algorithm to dynamically respond to user behavior during the writing process and continuously adjust the suggestion generation strategy based on historical interaction data.

In terms of real-time feedback generation, the system monitors the incremental changes of user input stream $\{w_1, w_2, \ldots, w_t\}$ and extracts the current context C_t through a sliding window mechanism. The generation module immediately calls the lightweight inference engine upon receiving new input, outputs a candidate suggestion set of $S_t = \{s_1, s_2, \ldots, s_k\}$, and uses a confidence scoring function to:

$$\mathrm{Score}(s_i) = \lambda_1 \cdot \mathrm{Rel}(s_i, C_t) + \lambda_2 \cdot \mathrm{Flu}(s_i) + \lambda_3 \cdot \mathrm{TaskFit}(s_i) \quad (8)$$

Among them, Rel represents semantic relevance, Flu represents language fluency, TaskFit is the task adaptation score, and $\lambda_1, \lambda_2, \lambda_3$ is the weight parameter. The system selects the highest score suggestion in real-time as the first recommendation output, while retaining other suggestions for users to choose from or refer to. In terms of optimizing personalized writing suggestions, the system maintains an interaction profile of $P_u = \{\mathrm{style}, \mathrm{error\_freq}, \mathrm{vocab\_level}\}$ for each user by updating rules:

$$p_u^{(t+1)} = p_u^{(t)} + \eta \cdot \Delta P_u \quad (9)$$

Among them, ΔP_u comes from users' adoption behavior, manual modification records, and satisfaction ratings. When the adoption rate of a certain type of suggestion is significantly higher than the average level, the system will increase its generation priority; Conversely, it decreases. This process combines the Multi Armed Bandit strategy to dynamically balance exploring new types of suggestions with utilizing high adoption rate suggestions, thereby achieving personalized generation optimization.

5.4 Low latency inference architecture and multi-threaded parallel processing scheme

Inference latency critically affects user interaction, especially during long-text generation and multi-round sessions. Therefore, this study designed a low latency inference architecture and a multi-threaded parallel processing scheme, which significantly reduces response time while ensuring generation quality.

In terms of low latency architecture, the system adopts a strategy that combines layer wise inference scheduling with model sharding, splitting the Transformer decoding process into key layers (preserving complete accuracy) and non key layers (using quantization calculation), and enabling FP16 mixed precision operation in non key layers to reduce the computational load of single inference. The inference delay of T_{total} can be approximated as:

$$T_{total} = \sum_{i=1}^{n} \frac{C_i}{P_i} + T_{comm} \quad (10)$$

Among them, C_i represents the computational workload of the i-th layer, P_i represents the number of processing threads allocated to that layer, and T_{comm} represents the data communication overhead between sharded models. By optimizing the allocation of P_i, parallel and pipeline computing at different layers can be achieved.

In terms of multi-threaded parallelism, the system divides tasks into three categories: input parsing threads (preprocessing and word segmentation), core generation threads (model inference), and post-processing threads (recommended filtering and formatting).

6 Discussion

To further evaluate the system's competitiveness, Table 7 compares its generation quality against several existing writing-assistance systems on the same test corpus using BLEU, ROUGE-L, and BERTScore metrics.

Table 7 :Comparison with representative SOTA writing-assistance systems

System / Model	BLEU ↑	ROUGE-L ↑	BERTScore ↑	Key Features
Grammarly (2023)	0.67	0.71	0.83	Rule-based grammar + context check
GPT-4 API (2024)	0.73	0.78	0.86	Large-language-model generation only
EduWrite Transformer (2023)	0.76	0.80	0.88	Transformer without feedback loop
Proposed System (ours)	0.82	0.86	0.91	Transformer + semantic control + real-time feedback loop

As shown, the proposed model achieves higher scores across all metrics, with an average 6–8 percentage-point gain over the closest baseline (EduWrite). The improvement arises mainly from the semantic-constraint layer that filters off-topic generation and from the personalized feedback loop that continually adapts to user corrections during writing.

6.1 Adaptation analysis and optimization direction for different user groups

In the process of system application, different user groups show significant differences in language foundation, writing purpose, and interaction habits, which puts forward multi-dimensional adaptability requirements for English writing intelligent assistance systems based on generative algorithms. For beginners, they rely more on the structured suggestions and grammar correction functions provided by the system, hoping that the generated results can cover basic vocabulary and common sentence patterns, while having high comprehensibility and learning guidance; For academic writers with high language proficiency, the value of the system is more reflected in advanced grammar optimization, precise use of professional terminology, and discourse logic adjustment. They have higher requirements for the fluency, information density, and semantic accuracy of the generated results.

From the perspective of domain usage, learning users prefer high-frequency instant feedback to receive immediate revisions during the writing process, while enterprise employees or researchers value the ability to produce large quantities and make timely revisions to shorten work cycles and maintain output specialization and consistency. In addition, people from different cultures and regions have their own different understandings of language environment, voice order, and polite culture, so the system needs to have localization ability and be able to adapt to the above changes.

The clues for future optimization are proposed from three aspects: firstly, developing and implementing a way to adjust user styling in real-time, so that the recommendation output of this system always matches the creative characteristics of oneself; The second is to expand data sources outside the domain to enhance the knowledge of the system's applicability to multiple fields; The third is algorithm self optimization based on machine learning, which fully utilizes user feedback to achieve rapid correction of common errors and accurate capture of unusual needs, thereby maintaining high operability and satisfaction among various user groups.

6.2 Research on the promotion path and scalability of the system in educational scenarios

Promoting the proposed generative-algorithm-based writing system in education requires consideration of technical adaptability, pedagogical integration, and sustainable maintenance. The introduction of this system should not be limited to a single writing tool, but should serve as a comprehensive support platform for classroom teaching, after-school tutoring, and self-directed learning. The promotion path can be divided into three stages: pilot verification, hierarchical promotion, and large-scale application.

In the pilot phase, partial deployment was carried out on some universities, language training institutions, and language online classrooms to obtain data on various teaching methods, and to study the effectiveness and stability of real-time writing assistance, task assignment evaluation, scientific research assistance, etc. based on generative algorithms; In the process of hierarchical amplification, based on the differentiated needs of high school, university, and adult learning in different disciplines, the recommendation fineness, generation format, and response methods were systematically optimized to highly match the subject objectives and curriculum structure; In the global application stage, relying on the architecture combination of cloud computing and edge computing, it has realized the high-performance access of large-scale parallel users and satisfied the learning needs and syllabus of each region based on the iterative strategy of localized language models.

In terms of the system's scalability, the structure should have multiple modules and support for hot swapping, in order to quickly integrate new functions such as cross domain writing support, multilingual generation, and support for advanced integration of learning management systems (LMS) when new educational needs change. In addition, the core algorithm should provide open interfaces to enable the generation and evaluation of third-party educational software usage, achieving ecological expansion. With the help of data-driven iterative optimization and cross platform capabilities, this system can extend from a simple educational software to an intelligent learning assistant in the digital revolution of education, and has long-term usage prospects and the possibility of cross-border migration.This empirical comparison further confirms that the integration of semantic-control and feedback mechanisms enables the system to outperform existing SOTA writing assistants in both linguistic accuracy and user adaptation.

7 Conclusion

This study focuses on building a comprehensive algorithm system for machine-assisted human creation of English writing works based on generative algorithms, including system architecture, core algorithm and text processing program design and functional debugging, testing effectiveness and optimization strategy research. Implement the modular architecture system and functional division design that combines core algorithms, semantic control, and human-computer dialogue into the system design, and build a complete closed-loop system based on input analysis to personalized suggestion output work.

In terms of performance evaluation, a comprehensive evaluation method has been constructed for multiple dimensions such as generation quality, user utility, and user experience. Through experimental data and a large amount of data in the application environment, the error situation of the system is measured from important dimensions such as accuracy, fluency, and latency, and corresponding constraints and error types are identified. Therefore, corresponding modification plans are proposed, including adjusting the structural parameters of the core model, adjusting the adaptability of parameters, increasing the sample size to achieve error driven reconstruction and update, real-time feedback generation, reducing latency calculation, and multi-threaded parallel mode. This has resulted in significant improvements in response speed, generation quality, and user matching performance.

Through research, it has been found that the system can provide good applicability in the use of different language abilities and creative tasks, and can also be expanded in teaching space and different domain spaces. Technically, a reasoning architecture that reduces latency and flexible adjustment methods have been proposed. In application, it can be used to support intelligent learning, academic writing, and cross-cultural communication, and has the possibility of continuous expansion and promotion. For future work, how to further support the application of more languages and multimodal generation functions, so that they can better play their role in learning spaces, research spaces, etc., and continuously enrich their value is an important direction of work.

Ethical and Privacy Statement:

All user data collected in this study—including writing logs, feedback interactions, and questionnaire responses—were anonymized at the time of storage. No personally identifiable information (PII) was retained. The research protocol was reviewed and approved by the Academic Ethics Committee of Zhejiang Yuexiu University (Approval No. ZYU-AI2025-015). Participants provided written informed consent before data collection. Data handling complied with the requirements of the Chinese Personal Information Protection Law (PIPL, 2021) and relevant principles of the EU General Data Protection Regulation (GDPR). All datasets were used solely for academic research purposes and will not be shared publicly without additional consent.

References

[1] Yuan L .Research on English Hybrid Assisted Teaching System Using Contextual Support of R-CNN[J].Wireless Communications & Mobile Computing, 2022.https://dol:10.1155/2022/5358546.

[2] Mei Z , Lin Z .Design of Vocabulary Query System in Computer Aided English Translation Teaching[C]//EAI International Conference, BigIoT-EDU.Springer, Cham, 2024.https://dol:10.1007/978-3-031-63136-8_8.

[3] Mejeh M, Rehm M. Taking adaptive learning in educational settings to the next level: Leveraging natural language processing for improved personalization[J]. Educational technology research and development, 2024, 72(3): 1597-1621.https://doi.org/10.1007/s11423-024-10345-1

[4] Chai J .On Teaching Graduate English Academic Writing Based on Artificial Intelligence[J].Procedia Computer Science, 2024, 247(000):1366-1373.https://dol:10.1016/j.procs.2024.10.163.

[5] Wang C , Li Z , Bonk C .Understanding self-directed learning in AI-Assisted writing: A mixed methods study of postsecondary learners[J].Computers and Education: Artificial Intelligence, 2024, 6.https://dol:10.1016/j.caeai.2024.100247.

[6] Ding L, Zou D. Automated writing evaluation systems: A systematic review of Grammarly, Pigai, and Criterion with a perspective on future directions in the age of generative artificial intelligence[J]. Education and Information Technologies, 2024, 29(11): 14151-14203.https://doi.org/10.1007/s10639-023-12402-3

[7] Jingning L .Speech recognition based on mobile sensor networks application in English education intelligent assisted learning system[J].Measurement: Sensors, 2024, 32(000).https://dol:10.1016/j.measen.2024.101084.

[8] Jiang L , Wang Y .English Online Teaching Assistant System Based on Genetic Algorithms[C]//EAI International Conference, BigIoT-EDU.Springer, Cham, 2024.https://dol:10.1007/978-3-031-63136-8_49.

[9] Tu X , Zhang H .Educational Service Platform and English Corpus Design Based on Embedded Intelligent System and Cloud Computing[J].Wireless Communications & Mobile Computing, 2022.https://dol:10.1155/2022/1916941.

[10] Dai B .Simulation of interactive E-learning mode based on association rule mining in computer English reading assistance system[J].Entertainment Computing, 2024, 51.https://dol:10.1016/j.entcom.2024.100742.

[11] Zhao Q , Nazir S .English Multimode Production and Usage by Artificial Intelligence and Online Reading for Sustaining Effectiveness[J].Mobile Information Systems, 2022.https://dol:10.1155/2022/6780502.

[12] Pack A , Barrett A , Escalante J .Large language models and automated essay scoring of English language learner writing: Insights into validity and reliability[J].Computers and Education: Artificial Intelligence, 2024, 6.https://dol:10.1016/j.caeai.2024.100234.

[13] Lee Y J , Davis R O .A case study of implementing generative AI in university's general English courses[J].Contemporary Educational Technology, 2024, 16(4).https://dol:10.30935/cedtech/15218.

[14] Yang S H .Enhancing English as a Foreign Language Learners' Writing Performance and Experience With an Artificial Intelligence Powered Tool[J].Springer, Cham, 2024.https://dol:10.1007/978-3-031-76293-2_9.

[15] Liu M , Zhang L J , Biebricher C .Investigating students' cognitive processes in generative AI-assisted digital multimodal composing and traditional

writing[J].Computers & education, 2024, 211(Apr.):1.1-1.21.https://dol:10.1016/j.compedu.2023.104977.

[16] Han X .An Improved Classification Model for English Syntax Error Correction Design of DL Algorithm[J].International Arab Journal of Information Technology (IAJIT), 2024, 21(4).https://dol:10.34028/iajit/21/4/2.

[17] Gustilo L , Ong E , Lapinid M R .Algorithmically-driven writing and academic integrity: exploring educators' practices, perceptions, and policies in AI era[J].International Journal for Educational Integrity, 2024, 20(1):43.https://dol:10.1007/s40979-024-00153-8.

[18] Asad M M , Shahzad S , Shah S H A ,et al.ChatGPT as artificial intelligence-based generative multimedia for English writing pedagogy: challenges and opportunities from an educator's perspective[J].International Journal of Information and Learning Technology, 2024, 41(5).https://dol:10.1108/IJILT-02-2024-0021.

[19] Ali S .Locating the Intersection of Generative Artificial Intelligence and Human English Writing Skills: A Comparative Study[J].Arab World English Journal, 2024.https://dol:10.24093/awej/ChatGPT.7.

[20] Zenni R D , Andrew N R .Artificial Intelligence text generators for overcoming language barriers in ecological research communication[J].Austral Ecology, 2023(7):48.https://dol:10.1111/aec.13417.