Improving Proficiency and Grammar Accuracy for Chinese Language Learners with Large Language Models

Yuqi Liang and Wenjing Xu and Hongzhi Xu
Institute of Language Sciences
Shanghai International Studies University
{yuqiliang, xuwenjing, hxu}@shisu.edu.cn

Abstract

In this study, we evaluate the performance of large language models (LLMs) in detecting and correcting grammatical errors made by Chinese language learners. We find that incorporating various linguistic features—such as dependency structures, parts of speech, and pinyin transliteration—into the prompts can potentially enhance model performance. Among these features, parts of speech and pinyin prove to be the most effective across all tested models. Additionally, our findings show that the success of error correction also depends on the severity of the errors. When the intended meaning is preserved, LLMs tend to provide accurate revisions following the principle of minimal editing. However, when the meaning is obscured, LLMs are more likely to produce divergent outputs, both in comparison to reference corrections and to the responses of other models.

1 Introduction

Grammatical error correction (GEC) is a task of automatically identifying and correcting grammatical errors in text regarding words, syntax, semantics, and pragmatics, yielding sentences that are both grammatically correct and faithful to the intended meaning. GEC is of great importance for supporting and facilitating second language learners (Qiu et al., 2025). It is a difficult task due to its requirement of deep understanding of sentence structure. In recent years, large language models (LLMs) has led to significant breakthroughs in NLP (Qin et al., 2024). With the emergence of LLMs, GEC is able to obtain a significant boost.

Current research on applying LLMs to GEC can be broadly categorized into two lines. The first focuses on model training and optimization, including architecture adaptation, fine-tuning, instruction tuning, and leveraging pseudo-data generation from LLM to train smaller, task-specific models (Li and Wang, 2024; Li et al., 2024; Xiao et al., 2024). The second line centers on performance evaluation, with a primary focus on key issues such as the robustness of LLMs, alignment with human annotations, the effectiveness of error correction, and performance comparison across various data types (Zhang et al., 2023; Lin, 2024; Gao, 2025; Qu et al., 2025) .

While extensive work has explored English grammatical error correction using LLMs, research on their application to Chinese remains limited. Existing research primarily focuses on zero-shot learning, few-shot learning, chain-of-thought prompting, and instruction tuning (Jiang et al., 2023). However, they all assume that the given sentences are erroneous, which can potentially lead to model bias and over correction in real situations. Meanwhile, little research has been conducted on how linguistic features can influence the performance of models on the task.

In this study, we propose a unified evaluation framework for both grammatical error detection and correction in Chinese text. We assess five state-of-the-art LLMs under both basic and linguistically-enhanced prompts. Specifically, we examine the impact of features including dependency structure, parts of speech, and pinyin transliteration on the models' ability to detect and correct errors. The study aims to answer three key questions: 1) Can LLMs accurately detect grammatical error sentences in Chinese? (2) Does incorporating linguistic features improve correction performance, and which are most effective? (3) How does error severity affect LLMs' revision behavior, and what patterns emerge across different error types? By addressing these questions, our work provides important insights into how LLMs perform on Chinese grammatical error correction (CGEC) and how future model be optimized with better error correction strategies.

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1216–1232
December 20-24, 2025 ©2025 Association for Computational Linguistics

2 Related Works

GEC has become a critical task in natural language processing, aiming to improve the fluency and grammaticality of text. While early research primarily focused on English, especially through benchmark datasets such as CoNLL-2014 (Ng et al., 2014), JFLEG (Napoles et al., 2017), and BEA-2019 (Bryant et al., 2019), recent studies have extended GEC to a wide range of languages including Japanese, German, and Chinese. Initial approaches relied on statistical machine translation (SMT), which were gradually replaced by sequence-to-sequence models and Transformer-based architectures (Omelianchuk et al., 2020). GEC is typically decomposed into two sub-tasks: error detection, which identifies whether a sentence contains errors, and error correction, which generates the corrected version. A key distinction in the literature lies in whether these two sub-tasks are modeled jointly or separately. Several studies advocated for modular pipelines to better isolate detection errors and facilitate targeted improvements (Rei and Yannakoudakis, 2016; Chollampatt and Ng, 2018; Omelianchuk et al., 2020).

To improve performance, researchers have incorporated explicit linguistic features into GEC systems. In the English domain, features such as part-of-speech tags, syntactic dependencies, and semantic roles have been used to enhance error representations and inform correction strategies (Rozovskaya and Roth, 2016; Kaneko et al., 2020). Zhang et al. (2022b) demonstrated that integrating syntactic parse trees into neural architectures improves precision in GEC tasks, particularly for structural errors. Such findings suggest that linguistic features help models better generalize across diverse grammatical patterns and improves interpretability.

In the Chinese domain, GEC presents unique challenges due to the flexibility of Chinese grammar. While datasets such as NLPCC18, CGED-2021, and YACLC have supported system development, most existing approaches primarily focus on the correction stage, often treating detection as an auxiliary task. To enhance correction performance, Wang and Liang (2024) proposed a correction strategy based on linguistic knowledge and fluency enhancement. Deng et al. (2023) leveraged knowledge graphs to inject explicit syntactic rules into Chinese GEC models. These studies suggest that incorporating linguistic features into models is effective.

Additionally, considerable research has also been conducted on models for CGEC. Lin et al. (2023) proposed a model with syntax generalization and parameter sharing, achieving an F0.5 of 30.75 on two Chinese benchmarks while reducing parameters by about one-third. Yang and Quan (2024) introduced Alirector, an alignment-enhanced model that addresses overcorrection, demonstrating improved stability across three CGEC datasets. Wang et al. (2024) proposed a rewriting model that refines a single GEC hypothesis by filtering over-corrections, raising precision by 18.2% without sacrificing recall on a native-Chinese CGEC benchmark. Xiao et al. (2024) proposed an LLM-guided training method that leverages error types and confusion sets to generate diverse synthetic data and iteratively analyze traditional CGEC model predictions, significantly boosting Seq2Seq and Seq2Edit performance. Zhu et al. (2024) proposed a method using automatic sampling of heterogeneous corpora and weighted model ensembling, combining Seq2Seq and Seq2Edit models to achieve state-of-the-art performance in CGEC. These studies illustrate ongoing advances in CGEC via novel architectures and data-driven training, underscoring further opportunities for performance improvement.

The rise of LLMs such as ChatGPT has significantly reshaped the GEC field. In English, LLMs have achieved competitive results even with limited supervision, though studies have identified recurring problems including overcorrection, low recall on subtle errors, and stylistic inconsistencies (Loem et al., 2023; Ingólfsdóttir et al., 2023; Liang et al., 2025; Li and Lan, 2025). In the Chinese domain, efforts to adapt LLMs for CGEC include domain-specific pretraining (Fan et al., 2023), prompt engineering (Fang et al., 2023), and fine-tuning with error-annotated corpora (Yang and Quan, 2024). Recent advances such as retrieval-augmented generation (RAG) and parameter-efficient fine-tuning (PEFT) further enhance LLM performance by introducing external context and minimizing catastrophic forgetting (Soudani et al., 2024).

However, the behavior of LLMs under linguistically enhanced prompts, especially how features like parts of speech, dependency, or pinyin affect detection and correction, remains underexplored. Few studies provide fine-grained analysis of how LLMs revise different types of errors or how their behavior shifts across varying levels of error severity. To address these gaps, our study adopts a unified evaluation framework that incorporates both detection and correction, systematically examining how linguistic features influence model performance and how revision behaviors vary by error type.

3 Experimental Settings

In this section, we describe the setup of the experiments for testing LLMs in detection and correction of grammatical errors in Chinese text. Our goal is to understand: 1) how well LLMs can perform the tasks of error detection and correction; 2) whether adding linguistic features can help LLMs; 3) what are the error patterns produced by different LLMs.

3.1 Linguistic Features Extraction

As we discuss above, the grammatical errors span across various linguistic units including phonology, characters, words, syntax, and semantics. Therefore we consider several different features that can be automatically obtained through existing tools, including pinyin, parts of speech, and dependency structures. Pinyin provides important hints of abnormal phonological patterns; parts of speech convey important syntactic and semantic information within and between words; dependency structures include information about syntactic relations between words. For sentences with grammatical errors, these extracted features will show erroneous patterns, enabling the model to detect and address grammatical errors by leveraging inconsistencies embedded within the linguistic features.

In our study, we use the pypinyin library to obtain pinyin of the given sentence and LTP[1] to extract parts of speech, and dependency structures. We conduct an inspection after annotation and find that correct sentences are labeled accurately. However, for incorrect sentences, due to flawed semantics or structures, even humans struggle to annotate them reliably. Therefore, we retain the tool's annotations by default, as they still reflect the erroneous information to some extent. All these features are encoded as sequences of tuples and are derived directly from the sentences in the dataset. An example is shown in Table 1.

Sentence	周六我的男朋友的展览会。 My boyfriend's exhibition on Saturday.
Syntactic Structure	(1, '周六', 6, 'ATT'), (2, '我', 4, 'ATT'), (3, '的', 2, 'RAD'), (4, '男朋友', 6, 'ATT'), (5, '的', 4, 'RAD'), (6, '展览会', 0, 'HED'), (7, '。', 6, 'WP')
Parts of Speech	('周六', 'nt'), ('我', 'r'), ('的', 'u'), ('男朋友', 'n'), ('的', 'u'), ('展览会', 'n'), ('。', 'wp')
Pinyin	zhou1 liu4 wo3 de nan2 peng2 you3 de zhan3 lan3 hui4 。

Table 1: Linguistic Features Extraction Example.

3.2 The tested LLMs and Prompt

To comprehensively evaluate the ability of LLMs, we select five state-of-the-art LLMs for evaluation. These include two proprietary models GPT-4o and Claude 3.7 Sonnet as well as three open-source models DeepSeek-R1, DeepSeek-V3, and LLaMA 3.3 70B. The goal is to examine whether the integration of linguistic features enhances model performance in Chinese grammatical error detection and correction tasks, and to conduct a systematic evaluation of this enhancement. Prompt construction is based on five core components: role specification, task instruction, data description, output format constraints, and sentence input. The data description component is adjusted according to the type of linguistic features provided. In total, five types of prompts are designed: (1) a base prompt without any linguistic features; (2) a dependency-enhanced prompt; (3) a POS-enhanced prompt; (4) a pinyin-enhanced prompt; and (5) a combined prompt incorporating all three types of linguistic features. Detailed prompt templates can be found in the appendix. Note that here we demonstrate the prompt in English in Table 2. In real testing, we use Chinese language for prompt, and the Chinese version can be found in Appendix A.

3.3 Test Data

In this study, we utilize the publicly available data YACLC-Minimal[2], which serves as the development set for Track 3 of the CCL2022 Chinese Learner Text Correction (CLTC) task. The dataset comprises 1,839 sentences produced by Chinese language learners, each annotated by one or more annotators. The annotations involve minimal edits to grammatically ill-formed sentences, following the principles of meaning preservation and minimal

[1] https://github.com/HIT-SCIR/ltp

[2] https://github.com/blcuicall/CCL2022-CLTC/blob/main/datasets/track3

Component	Prompt
Role Specification	You are a Chinese language expert with advanced proficiency in Chinese grammar.
Task Instruction	You are given a set of Chinese sentences written by learners, which may contain grammatical errors. For each sentence, first determine whether it contains an error. If the sentence is correct, no revision is needed. If an error exists, revise the sentence following the minimum-editing principle: preserve the original structure as much as possible and minimize additions, deletions, or substitutions, ensuring the sentence conforms to standard Chinese grammar.
Data Description	The input is in JSON format and includes a serial number, the sentence, and its dependency structure. The dependencies are represented as tuples in the format (word index, word, head index, dependency label). These syntactic relations are provided for reference only and may contain errors.
Output Format Constraints	The output should be in JSON format and include: serial number, correctness status, and the modified sentence. Do not return any additional or irrelevant content. For example: {“serial number”: 1, “correctness status”: “correct”, “modified sentence”: “empty”}, {“serial number”: 2, “correctness status”: “wrong”, “modified sentence”: “revised sentence.”}
Sentence Input	Sentences to be processed: { “serial number": 1, “sentence": "I recognize some of the words article in.", “dependency structure": [(1, “-", 3, “WP"), (2, “I", 3, “SBV"), (3, “recognize", 0, “HED"), (4, “some", 5, “ATT"), (5, “word", 6, “ATT"), (6, “article", 7, “ATT"), (7, “in", 3, “VOB"), (8, “.", 3, “WP")] }

Table 2: The prompt for error detection and correction (English Version).

modification, with the goal of producing grammatically well-formed Chinese sentences.

The original dataset contains only erroneous sentences, which does not reflect real-world usage in principle and can possibly lead to model bias. To address this issue, we reformulate the task into two phases: error detection and error correction. The dataset is randomly split into two subsets containing 920 and 919 sentences respectively. For the first subset, the first gold standard for each sentence is retained as input; for the second, the original uncorrected sentences are preserved.

3.4 Evaluation

For grammatical error detection, we use precision, recall and F1-score as evaluation metrics. For grammatical error correction, the evaluation is based on the comparison of the system output and the reference correction. We adopt an existing evaluation toolkit ChERRANT[3](Zhang et al., 2022a), which transforms both system and reference corrections into sequences of error-type tags, capturing the nature of edits rather than their exact forms. By aligning these error-type sequences, it evaluates whether the model has correctly identified and corrected the grammatical errors. Model performance is measured using precision (P), recall (R), and the F0.5 score, which places more emphasis on precision.

4 Experimental Results

4.1 Error Detection

This study evaluates the performance of LLMs on Chinese grammatical error detection (CGED), specifically focusing on sentence-level acceptability classification. We investigate how different linguistic features impact model performance, including the following settings: baseline (no features), dependency structure, POS, pinyin, and all features. Table 3 presents the results across models under different feature conditions.

Overall, the best performance is obtained by DeepSeek-V3 with part-of-speech features. Under the baseline setting, DeepSeek-V3 achieves the highest performance, with a Macro F1 score of 0.7483, followed by GPT-4o, Claude 3.7 Sonnet, and LLaMA 3.3 70B. DeepSeek-R1 performs the worst, with a Macro F1 of only 0.5842. Notably, all models show higher recall and precision in identifying errors, indicating a possible tendency of over-correction.

Overall, all linguistic features contribute to improved model performance compared to the baseline. Most models demonstrate gains in both Macro F1 and Accuracy when these features are incorporated. Pinyin proves to be the most effective feature for DeepSeek R1 and GPT-4o, while DeepSeek V3 and LLaMA benefit most from part-of-speech (POS) information. Claude achieves the best performance when dependency structures are included. Interestingly, performance declines across all models when all features are combined. This sug-

[3] https://github.com/HillZhang1999/MuCGEC/tree/main/scorers/ChERRANT

gests that excessively long prompts may hinder the LLMs' ability to accurately extract and utilize relevant information.

4.2 Grammatical Error Correction

We analyze the models' performance across three different scopes: (1) the set of sentences identified as ungrammatical by each individual model, (2) the subset of sentences identified as erroneous by all models, and (3) the entire test set. The first setting highlights each model's ability to correct grammatical errors based on its own detection. The second setting allows for a more direct comparison across models by focusing on a shared set of detected errors. The third setting reflects the overall performance across the two-stage pipeline of error detection and correction. Although word-level and character-level F0.5 scores exhibit slight variations due to tokenization differences, the overall trends remain consistent. Therefore, we focus on character-level F0.5 in the main analysis.

The results are shown in Figure 1 and the detailed numbers are shown in Appendix B. We can see that the performance patterns are consistent across the three scopes. Similar to the first task, all linguistic features can further enhance performance. For example, DeepSeek-R1 with POS features achieves 0.5466 (vs. 0.4800 baseline), and DeepSeek-V3 with pinyin features reaches 0.6257, a gain of 9.24 percentage points over the baseline. GPT-4o also benefits from POS and pinyin, outperforming the baseline. Claude 3.7 Sonnet performs best with all features (0.5839). In contrast, LLaMA 3.3 70B shows the lowest scores and fails to benefit from feature fusion.

Overall, POS and pinyin features most effectively enhance revision accuracy, especially for DeepSeek models. Dependency features help in some cases, but with less consistency. Claude 3.7 Sonnet performs best among all the models in baseline and all features. Conversely, LLaMA 3.3 70B performs poorly throughout, with F0.5 dropping from 0.3544 to 0.3186 when all features are added.

In sum, DeepSeek-V3 ranks as the most effective model, followed by Claude 3.7 Sonnet, GPT-4o, and DeepSeek-R1, with LLaMA 3.3 70B lagging behind.

4.3 Error Analysis

The highest F0.5 scores on intersection sentences are mostly achieved when POS features are used. To further explore this, we analyze the distribution of character-level F0.5 scores. As shown in Figure 2, significant differences are observed. DeepSeek-V3 has the most top-scoring sentences (0.9–1.0), followed by Claude 3.7 Sonnet, indicating a preference for minimal, targeted edits. In contrast, LLaMA 3.3 70B shows more low scores (0.0–0.4), reflecting overcorrections and poor alignment with the original sentence structure. DeepSeek-R1 and GPT-4o strike a moderate balance, making necessary adjustments while retaining the original structure of the sentences.

To investigate how the severity and type of grammatical errors affect the revision behavior of LLMs, we conduct a detailed analysis based on four common categories of Chinese grammatical errors: redundant words (R), missing words (M), word selection errors (S), and word ordering errors (W). For each error type, we compute precision, recall, F0.5 scores, error density, false positive rate, false negative rate, aggressiveness (FP/TP), and conservativeness (FN/TP) using aligned edit operations between model outputs and gold-standard corrections on the subset of sentences commonly recognized as erroneous by all models.

As shown in Figure 3 (see Appendix C for detailed data), the results show that error density (i.e., the number of gold-standard edits per sentence) significantly influences the likelihood of model-initiated revisions. Missing words (M) and word selection errors (S) typically exhibit higher error densities, implying greater linguistic complexity and correction necessity. These types of errors often trigger more aggressive editing behavior, especially in models like LLaMA 3.3 70B, which shows a notably high false positive rate for these categories indicating frequent overcorrection.

In contrast, redundant word errors (R) are less frequent and tend to elicit more conservative model behavior. Models exhibit a high false negative rate for R-type errors, suggesting a general reluctance to delete tokens unless their redundancy is explicit. This cautious edit may be due to the potential semantic risk associated with deletion in Chinese syntax.

Word selection errors (S) involve subtle lexical and semantic nuances. They occur frequently and challenge model precision. While GPT-4o and Claude 3.7 maintain a relatively balanced correction strategy, LLaMA demonstrates overly aggressive revisions with poor precision and lower F0.5. Missing word errors (M) require strong syntactic inference. All models show elevated false negative

Model	Feature	Precision (Error)	Recall (Error)	F1 (Error)	Macro F1	Accuracy
Claude 3.7 Sonnet	Baseline	0.6632	0.9129	0.7683	0.7148	0.7249
	Dependency	0.7077	0.8694	0.7803	**0.7521**	**0.7553**
	POS	0.6893	0.8836	0.7744	0.7376	0.7428
	Pinyin	0.6868	0.8781	0.7708	0.7339	0.7390
	All	0.7138	0.8411	0.7722	0.7501	0.7520
DeepSeek-R1	Baseline	0.5795	0.9880	0.7305	0.5842	0.6357
	Dependency	0.5975	0.9771	0.7415	0.6216	0.6596
	POS	0.6068	0.9739	0.7477	0.6386	0.6716
	Pinyin	0.6127	0.9761	0.7528	**0.6490**	**0.6797**
	All	0.5931	0.9706	0.7363	0.6136	0.6525
DeepSeek-V3	Baseline	0.6979	0.8923	0.7832	0.7483	0.7531
	Dependency	0.7464	0.7911	0.7681	0.7611	0.7613
	POS	0.7522	0.8455	0.7961	**0.7828**	**0.7836**
	Pinyin	0.7542	0.8313	0.7909	0.7798	0.7803
	All	0.7802	0.7029	0.7396	0.7520	0.7526
GPT-4o	Baseline	0.6789	0.8836	0.7678	0.7269	0.7330
	Dependency	0.6928	0.8466	0.7620	0.7325	0.7357
	POS	0.6724	0.8890	0.7657	0.7209	0.7281
	Pinyin	0.6992	0.8498	0.7672	**0.7393**	**0.7423**
	All	0.6968	0.8400	0.7617	0.7346	0.7374
LLaMA 3.3 70B	Baseline	0.6282	0.8400	0.7188	0.6620	0.6716
	Dependency	0.6530	0.7291	0.6889	0.6699	0.6710
	POS	0.6604	0.7661	0.7093	**0.6843**	**0.6862**
	Pinyin	0.6677	0.7345	0.6995	0.6838	0.6846
	All	0.6560	0.6703	0.6631	0.6596	0.6596

Table 3: Evaluation results of error detection with different linguistics features. The precision, recall, F1 of the erroneous category, accuracy, and the macro average F1 are reported.

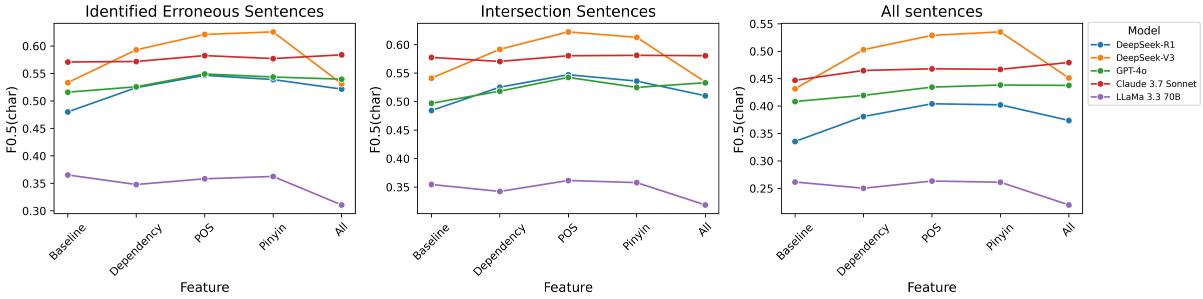

Figure 1: Results of Chinese Grammatical Error Correction.

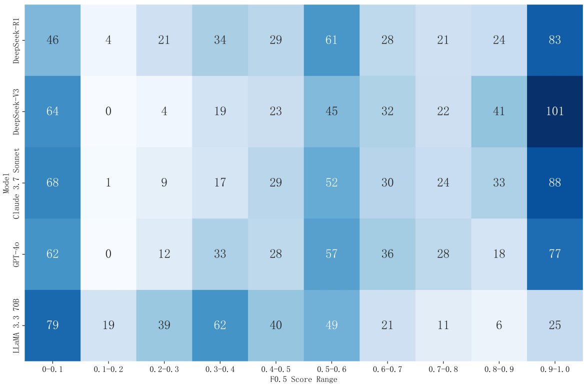

Figure 2: F0.5 Score Distribution of Editing Operations for Intersection Sentences.

rates in this category, reflecting difficulty in identifying omitted yet essential words in sentence structures. Redundant word errors (R) are structurally simpler but often undercorrected. Most models appear risk-averse when deleting tokens, possibly to avoid reducing fluency or meaning. Word ordering errors (W) are the least frequent but relatively well-handled. These errors are more syntactically salient and thus easier for LLMs to identify and revise correctly.

GPT-4o demonstrates the most balanced and stable performance across all error types, achieving high F0.5 scores with moderate aggressiveness and conservativeness. Claude 3.7 Sonnet is relatively cautious in its editing strategy, particularly for S and R errors. In contrast, LLaMA 3.3 70B exhibits consistently high aggressiveness, especially on selection and missing word errors, which leads to increased overcorrections and lower overall performance. DeepSeek-R1 and DeepSeek-V3 perform adequately on surface-level errors but tend to miss deeper semantic or syntactic inconsistencies.

Our analysis also finds that when learner texts

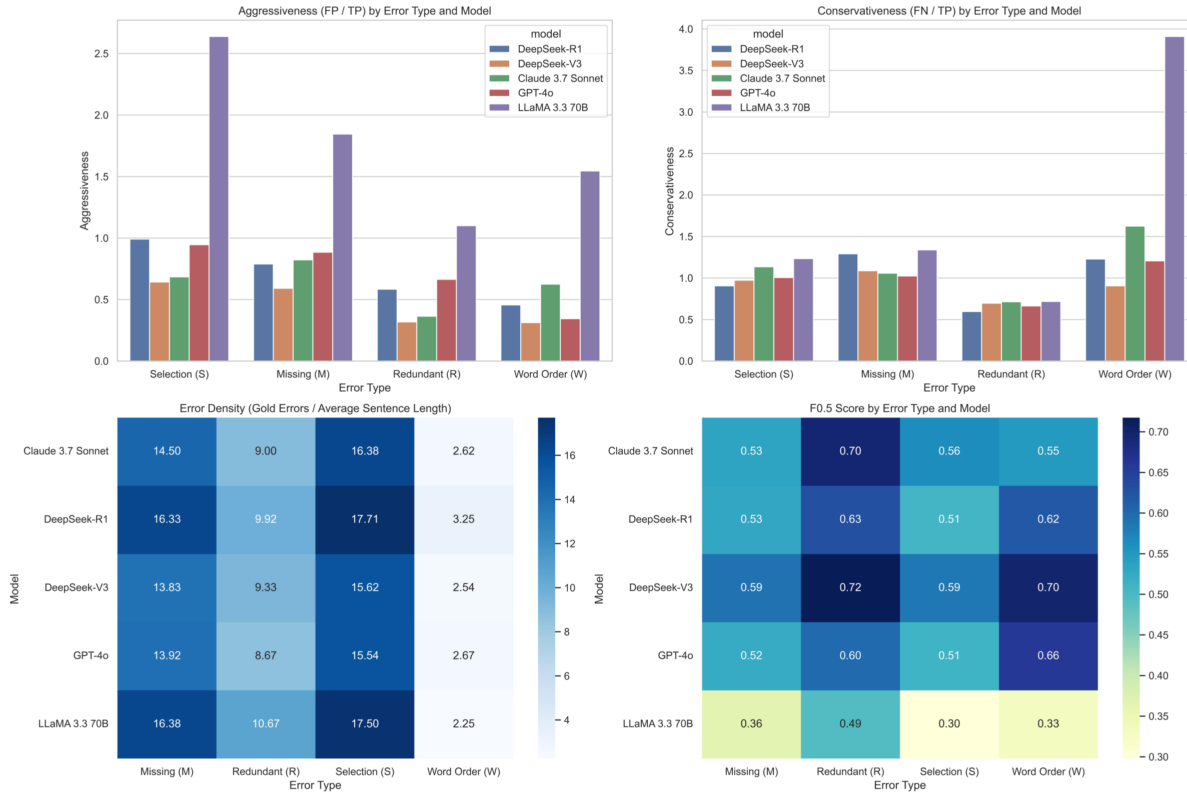

Figure 3: Editing Operations for Intersection Sentences: Aggressiveness, Conservativeness, Error Density, and F0.5 Score across Error Types and Models.

contain semantic errors, LLMs often fail to recover intended meaning. Instead, they perform better in identifying and revising grammatically marked but semantically transparent errors.

Table 4 shows some examples of error correction by different models (see Appendix D for English translations). The original sentence combines 打太极拳 *da tai ji quan* ‘practice Tai Chi’ and 跑长跑 *pao chang pao* ‘run long distances’ simultaneously, creating temporal ambiguity. The reference revision improves coherence by deleting 参加 *can jia* ‘participate’, adjusting word order, and adding adverbs for temporal clarity.

The first three models leave the temporal adverbial clause unchanged but vary in main clause revisions. DeepSeek-V3 adds 都会 *dou hui* ‘will always’, highlighting habitual behavior. DeepSeek-R1 restructures the sentence into a 去体育馆 *qu ti yu guan* ‘go to the gym’ + verb-object pattern, shifting the meaning and adding 还会 *hai hui* ‘will also’, which implies additional activities rather than regularity. Claude 3.7 Sonnet uses 跑着 *pao zhe* ‘running’, suggesting simultaneity, but produces 跑着长跑 *pao zhe chang pao* ‘running long distances’, a plausible yet illogical phrase that reflects misunderstanding of real-world verb-object usage. GPT-4o makes extensive changes, reordering the sentence to ‘go to the gym’ followed by a 参加. . . 和. . . *can jia ... he ...* ‘participate in. . . and. . . ’ structure. While logically sound, this alters the original intended meaning, implying both activities occur at the gym. LLaMA 3.3 70B rewrites the sentence with 后 *hou* ‘after. . . ’ to indicate temporal order and nominalizes 活动 *huo dong* ‘activities’ to resolve the incomplete object issue. Though semantically clear and structurally refined, it violates the minimum-editing principle, consistent with its low F0.5 score. The results suggest that unclear original meanings lead to divergent model interpretations. Although corrections may be syntactically and semantically valid, they can deviate from the original intended meaning.

The second erroneous sentence in Table 4 has a clear meaning, expressing a desire from a past point in time, but lacks an adverbial marker. The reference uses 就 *jiu* ‘just then’ to form a “temporal adverbial + 就 + predicate” structure, emphasizing the link between time and action. DeepSeek-V3, DeepSeek-R1, and Claude 3.7 Sonnet correctly insert 就 *jiu* ‘just then’ to reflect this relationship.

Erroneous Sentence	她每周参加打太极拳的时候，跑长跑去体育馆。	我从以前想去看这层建筑。	他们什么时候我不懂帮助我。
Gold	1）她每周打太极拳的时候，跑长跑去体育馆。 2）她每周打太极的时候，长跑去体育馆。 3）她每周参加打太极拳的时候，都是跑长跑去体育馆。 4）她在每周打太极拳的时候，都长跑去体育馆。 5）她每周参加打太极拳活动的时候，跑去体育馆。 6）她每周参加打太极拳的时候，都是跑步去体育馆。 7）她每周打太极拳的时候，都跑去体育馆。 8）她每周打太极拳的时候，长跑去体育馆。 9）她每周去打太极拳的时候，就长跑去体育馆。 10）她每周打太极拳的时候，跑长跑去体育馆。	1）我在以前就想去看这层建筑。 2）我从以前就想去看这座建筑。 3）我以前就很想去看这座建筑。 4）我以前就想去看这个建筑。 5）我以前就想来看看这栋建筑。 6）我从很久之前就想去看这层建筑。 7）我之前就想去看这座建筑。 8）我从很久以前就想去看这座建筑。 9）我以前就想去看这栋建筑。 10）我从以前想去看这个建筑。 11）我以前就想去看看这个建筑。	1）什么时候我不懂了他们都帮助我。 2）什么时候我不懂了，他们会帮助我。 3）他们在我不懂的时候帮助我。 4）他们不论什么时候只要我不懂就会帮助我。 5）他们会在我有什么地方不懂的时候帮助我。 6）他们在我不懂什么的时候会帮助我。 7）他们在我不懂的任何时候帮助我。 8）我什么时候不懂，他们就帮助我。 9）他们在任何时候都帮助我。 10）什么时候我有不懂的，他们就会帮助我。
DeepSeek-V3 (POS)	她每周参加打太极拳的时候，都会跑长跑去体育馆。(F0.5 = 0)	我从以前就想去看这层建筑。(F0.5 = 0.8333)	他们什么时候我不懂帮助我。(F0.5 = 0.7143)
DeepSeek-R1 (POS)	她每周参加打太极拳的时候，还会去体育馆跑长跑。(F0.5 = 0)	我早就想去看这栋建筑。(F0.5 = 0.8333)	他们什么时候帮助我，我不懂。(F0.5 = 0.7143)
Claude 3.7 Sonnet (POS)	她每周参加打太极拳的时候，跑着长跑去体育馆。(F0.5 = 0)	我从以前就想去看这座建筑。(F0.5 = 1)	他们什么时候帮助我我不懂。(F0.5 = 0.4545)
GPT-4o (POS)	她每周去体育馆的时候，参加打太极拳和跑长跑。(F0.5 = 0)	我一直想去看这层建筑。(F0.5 = 0.3333)	我不懂他们什么时候帮助我。(F0.5 = 0)
LLaMa 3.3 70B (POS)	她每周在参加完太极拳活动后，会去体育馆跑长跑。(F0.5 = 0)	我以前就一直想去看这座建筑。(F0.5 = 0)	他们什么时候帮助我我不懂。(F0.5 = 0.6667)

Table 4: Error Correction Examples from Intersection Sentences (Chinese Version).

In contrast, GPT-4o and LLaMA 3.3 70B overcorrect by adding 一直 *yi zhi* 'all along' or 从以前就一直 *cong yi qian jiu yi zhi* 'have always been... since then', stressing continuity rather than temporal linkage. This result exhibits that when the intended meaning is clear, LLMs tend to make more accurate and appropriate edits.

The third erroneous sentence in Table 4 contains an error related to the nested structure of the temporal adverbial clause, intending to express the meaning: "They will help me whenever I do not understand." The manually annotated reference answers recognized this underlying meaning and made appropriate modifications. However, none of the LLMs are able to identify this logical relationship, resulting in a lack of corresponding corrections. Although GPT-4o attempts to reconstruct the sentence by rearranging the subject and object, it is different from the original meaning. This highlights a common limitation of current LLMs in processing hypotactic grammar in Chinese, as they overly rely on linear syntactic processing and lack the ability to reconstruct implicit logical relationships.

4.4 Summarization and Discussion

Overall, our study shows that LLMs exhibit a reasonable ability in both error detection and correction based on the minimum-editing principle, and that incorporating linguistic features phonological or syntactic, can further improve their performance to varying extents. It shows the strong capability of LLMs in integrating complex information to generate accurate answers.

The study also reveals that both error severity (measured by density) and error type substantially shape LLM revision behavior. High-severity errors (S and M) tend to provoke more model edits, often at the cost of precision. Meanwhile, less severe or more syntactically explicit errors (R and W) are often conservatively handled. These findings suggest that LLMs follow distinct behavioral patterns conditioned by error characteristics, and that fine-grained categorization of error types is essential for evaluating and improving GEC systems.

5 Conclusion

This study systematically evaluates the performance of various LLMs on the tasks of Chinese grammatical error detection and correction, while also exploring the impact of linguistic features on model performance. The experimental results indicate that these models possess a reasonable ability in identifying grammatical errors and performing minimum-editing corrections, with features such as parts of speech, pinyin, and dependency structure significantly enhancing the performance of some models. GPT-4o, DeepSeek-V3, and Claude 3.7 Sonnet demonstrate strong robustness and feature responsiveness in both detection and correction tasks, while DeepSeek-R1 is slightly less effective. LLaMA 3.3 70B still shows deficiencies in handling Chinese.

To better understand model behavior, error analysis shows that LLM revision behavior is strongly influenced by error severity and type. Models are more aggressive with high-severity errors but tend to act conservatively on surface-level issues. These patterns highlight the importance of fine-grained error categorization for understanding and improving GEC performance. The findings provide a new perspective for evaluating models in Chinese Grammatical Error Correction tasks and underscore the critical value of linguistic feature design for optimizing model performance. Error analysis reveals that LLMs face challenges in understanding sentences with semantic errors and have trouble handling hypotactic grammar, particularly in understanding and reconstructing implicit logical relationships.

Limitations

Although this study evaluates the performance of various LLMs in the tasks of Chinese error detection and correction, some limitations remain. First, although multiple linguistic features are introduced to enhance model input, redundancy or interference among them may occur. Some models show decreased performance after integrating all features. This suggests that the feature fusion strategy needs further optimization. Second, model corrections are evaluated by whether the edits are correct. However, their connection to specific error types remains unclear. This limits interpretability. Future work could explore the link between error types and editing operations to reveal correction strategies for different error types, thereby enhancing the depth of analysis and model controllability.

References

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Shamil Chollampatt and Hwee Tou Ng. 2018. A multi-layer convolutional encoder-decoder neural network for grammatical error correction. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

Qian Deng, Shu Chen, and Junmin Ye. 2023. Chinese grammatical error correction based on grammatical knowledge enhancement. *Computer Engineering*, 49(11):77–84. In Chinese.

Yaxin Fan, Feng Jiang, Peifeng Li, and Haizhou Li. 2023. Grammargpt: Exploring open-source llms fornbsp;native chinese grammatical error correction withnbsp;supervised fine-tuning. In *Natural Language Processing and Chinese Computing: 12th National CCF Conference, NLPCC 2023, Foshan, China, October 12–15, 2023, Proceedings, Part III*, page 69–80, Berlin, Heidelberg. Springer-Verlag.

Tao Fang, Shu Yang, Kaixin Lan, Derek F. Wong, Jinpeng Hu, Lidia S. Chao, and Yue Zhang. 2023. Is ChatGPT a highly fluent grammatical error correction system? a comprehensive evaluation. *Preprint*, arXiv:2304.01746.

Zhaoming Gao. 2025. *Grammatical Error Correction and Explanation for Learners of Chinese Using Large Language Models*, pages 375–414. Springer Nature Singapore, Singapore.

Svanhvít Lilja Ingólfsdóttir, Petur Ragnarsson, Haukur Jónsson, Haukur Simonarson, Vilhjalmur Thorsteinsson, and Vésteinn Snæbjarnarson. 2023. Byte-level grammatical error correction using synthetic and curated corpora. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7299–7316, Toronto, Canada. Association for Computational Linguistics.

Haochen Jiang, Yumeng Liu, Houquan Zhou, Ziheng Qiao, Bo Zhang, Chen Li, Zhenghua Li, and Min Zhang. 2023. CCL23-eval task 7 track 1 system report: Suda &Alibaba team text error correction system. In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 3: Evaluations)*, pages 220–229, Harbin, China. Chinese Information Processing Society of China.

Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. 2020. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4248–4254, Online. Association for Computational Linguistics.

Wei Li and Houfeng Wang. 2024. Detection-correction structure via general language model for grammatical error correction. pages 1748–1763, Bangkok, Thailand.

Xinyuan Li and Yunshi Lan. 2025. Large language models are good annotators for type-aware data augmentation in grammatical error correction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 199–213, Abu Dhabi, UAE. Association for Computational Linguistics.

Yinghui Li, Shang Qin, Haojing Huang, Yangning Li, Libo Qin, Xuming Hu, Wenhao Jiang, Hai-Tao Zheng, and Philip S. Yu. 2024. Rethinking the roles of large language models in chinese grammatical error correction. *Preprint*, arXiv:2402.11420.

Jiehao Liang, Haihui Yang, Shiping Gao, and Xiaojun Quan. 2025. Edit-wise preference optimization for grammatical error correction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3401–3414, Abu Dhabi, UAE. Association for Computational Linguistics.

Nankai Lin, Xiaotian Lin, Yingwen Fu, Shengyi Jiang, and Lianxi Wang. 2023. A chinese grammatical error correction model based on grammatical generalization and parameter sharing. *The Computer Journal*, 67(5):1628–1636.

Sha Lin. 2024. Evaluating llms' grammatical error correction performance in learner chinese. *PLOS ONE*, 19(10):1–18.

Mengsay Loem, Masahiro Kaneko, Sho Takase, and Naoaki Okazaki. 2023. Exploring effectiveness of GPT-3 in grammatical error correction: A study on performance and controllability in prompt-based methods. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 205–219, Toronto, Canada. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR – grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170, Seattle, WA, USA → Online. Association for Computational Linguistics.

Libo Qin, Qiguang Chen, Xiachong Feng, Yang Wu, Yongheng Zhang, Yinghui Li, Min Li, Wanxiang Che, and Philip S. Yu. 2024. Large language models meet nlp: A survey. *Preprint*, arXiv:2405.12819.

Mengyang Qiu, Qingyu Gao, Linxuan Yang, Yang Gu, Tran Minh Nguyen, Zihao Huang, and Jungyeul Park. 2025. Chinese grammatical error correction: A survey. *Preprint*, arXiv:2504.00977.

Fanyi Qu, Chenming Tang, and Yunfang Wu. 2025. Evaluating the capability of large-scale language models on chinese grammatical error correction task. *Preprint*, arXiv:2307.03972.

Marek Rei and Helen Yannakoudakis. 2016. Compositional sequence labeling models for error detection in learner writing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1181–1191, Berlin, Germany. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2016. Grammatical error correction: Machine translation and classifiers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2205–2215, Berlin, Germany. Association for Computational Linguistics.

Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. 2024. Fine tuning vs. retrieval augmented generation for less popular knowledge. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP 2024)*, SIGIR-AP 2024, pages 12–22, New York, NY, USA. Association for Computing Machinery.

Yan Wang and Yeling Liang. 2024. A chinese grammatical error correction model enhanced by knowledge and fluency strategies. *Information Technology and Informatization*, (05):107–110. In Chinese.

Yixuan Wang, Baoxin Wang, Yijun Liu, Dayong Wu, and Wanxiang Che. 2024. LM-combiner: A contextual rewriting model for Chinese grammatical error correction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10675–10685, Torino, Italia. ELRA and ICCL.

Liu Xiao, Ying Li, and Zhengtao Yu. 2024. Chinese grammatical error correction via large language model guided optimization training. In *Proceedings of the 23rd Chinese National Conference on Computational Linguistics (Volume 1: Main Conference)*, pages 1366–1380, Taiyuan, China. Chinese Information Processing Society of China.

Haihui Yang and Xiaojun Quan. 2024. Alirector: Alignment-enhanced Chinese grammatical error corrector. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2531–2546, Bangkok, Thailand. Association for Computational Linguistics.

Xiaowu Zhang, Xiaotian Zhang, Cheng Yang, Hang Yan, and Xipeng Qiu. 2023. Does correction remain a problem for large language models? *Preprint*, arXiv:2308.01776.

Yue Zhang, Zhenghua Li, Zuyi Bao, Jiacheng Li, Bo Zhang, Chen Li, Fei Huang, and Min Zhang. 2022a. MuCGEC: a multi-reference multi-source evaluation dataset for Chinese grammatical error correction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3118–3130, Seattle, United States. Association for Computational Linguistics.

Yue Zhang, Bo Zhang, Zhenghua Li, Zuyi Bao, Chen Li, and Min Zhang. 2022b. SynGEC: Syntax-enhanced grammatical error correction with a tailored GEC-oriented parser. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2518–2531, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Shichang Zhu, Jianjian Liu, Ying Li, and Zhengtao Yu. 2024. Automatical sampling with heterogeneous corpora for grammatical error correction. *Complex & Intelligent Systems*, 11(1):25.

A Prompts for Evaluation of LLMs

See Table 5 and Table 6.

B F0.5 Score for Grammatical Error Correction

See Figure 4, Table 7, Table 8 and Table 9.

C Detailed Evaluation Metrics by Error Type and Model

See Table 10.

D Error Correction Examples from Intersection Sentences

See Table 11.

Component	Prompt
Role Specification	你现在是一名中文专家，精通中文知识。 *You are a Chinese language expert with advanced proficiency in Chinese grammar.*
Task Instruction	现在有一些汉语学习者写的中文句子，这些句子可能存在错误，需要你结合你的中文知识进行分析，先判断句子是否存在语病，如果不存在语病，不需要修改；如果存在语病，请进行修改，修改的过程需要遵循最小改动的原则。最小改动维度需要尽可能地维持原句的结构，尽可能少地增删、替换句中的词语，使句子符合汉语语法规则。 *You are given a set of Chinese sentences written by learners, which may contain grammatical errors. For each sentence, first determine whether it contains an error. If the sentence is correct, no revision is needed. If an error exists, revise the sentence following the minimum editing principle: preserve the original structure as much as possible and minimize additions, deletions, or substitutions, ensuring the sentence conforms to standard Chinese grammar.*
Data Description	输入是JSON格式的句子信息，包括{序号, 句子, LTP依存句法关系}。依存句法关系是由一系列元组表示，其格式为（单词序号, 单词, 当前单词句法依存于另一个单词的序号, LTP句法依存标签）。句法依存给出的分析结果仅供参考，有可能出错。 *The input is in JSON format and includes a serial number, the sentence, and its dependency structure. The dependencies are represented as tuples in the format (word index, word, head index, dependency label). These syntactic relations are provided for reference only and may contain errors.*
Output Format Constraints	返回的输出需要是JSON格式，包括{序号, 是否正确, 修改后的句子}。其他无关内容不要返回。如：{“序号”: 1, “是否正确”: “正确”, “修改后的句子”: “空”}, {“序号”: 2, “是否正确”: “错误”, “修改后的句子”: “修改后的句子1”} *The output should be in JSON format and include: serial number, correctness status, and the modified sentence. Do not return any additional or irrelevant content. For example: {‘serial number”: 1, correctness status”: correct”, modified sentence”: empty”}, {serial number”: 2, correctness status”: wrong”, modified sentence”: revised sentence.”}*
Sentence Input	需要完成任务的句子： [‘序号’: 1, ‘句子’: ‘-我认识有些字文章里。’, ‘依存句法关系’: [(1, ‘-’, 3, ‘WP’), (2, ‘我’, 3, ‘SBV’), (3, ‘认识’, 0, ‘HED’), (4, ‘有些’, 5, ‘ATT’), (5, ‘字’, 6, ‘ATT’), (6, ‘文章’, 7, ‘ATT’), (7, ‘里’, 3, ‘VOB’), (8, ‘。’, 3, ‘WP’)]] *Sentences to be processed:* *{ “serial number": 1, “sentence": "I recognize some of the words article in.", “dependency structure": [(1, “-", 3, “WP"), (2, “I", 3, “SBV"), (3, “recognize", 0, “HED"), (4, “some", 5, “ATT"), (5, “word", 6, “ATT"), (6, “article", 7, “ATT"), (7, “in", 3, “VOB"), (8, “.", 3, “WP")] }*

Table 5: Prompt Example (Chinese Version).

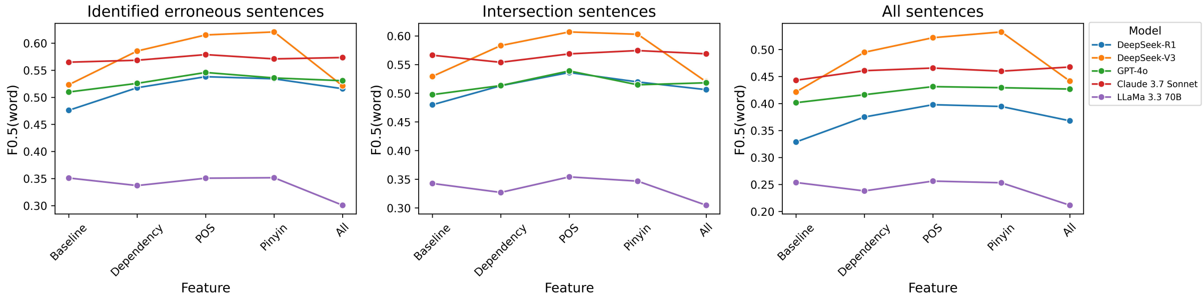

Figure 4: Results of Chinese Grammatical Error Correction (Word-level).

Types of Data Description	Prompt
Data Description (Baseline)	输入是JSON格式的句子信息，包括{序号, 句子}。 *The input is in JSON format and includes a serial number and the sentence.*
Data Description (Dependency Structure)	输入是JSON格式的句子信息，包括{序号, 句子, LTP依存句法关系}。依存句法关系是由一系列元组表示，其格式为（单词序号, 单词, 当前单词句法依存于另一个单词的序号, LTP句法依存标签）。句法依存给出的分析结果仅供参考，有可能出错。 *The input is in JSON format and includes a serial number, the sentence, and its dependency structure. The dependencies are represented as tuples in the format (word index, word, head index, dependency label). These syntactic relations are provided for reference only and may contain errors.*
Data Description (POS)	输入是JSON格式的句子信息，包括{序号, 句子, 句子的每个单词的词性}。词性是由一系列元组表示，其格式为（单词，当前单词的LTP词性标签）。句子单词词性给出的分析结果仅供参考，有可能出错。 *The input is in JSON format and includes a serial number, the sentence, and the part-of-speech tags for each word. The POS tags are represented as tuples in the format (word, POS tag). These POS tags are provided for reference only and may contain errors.*
Data Description (Pinyin)	输入是JSON格式的句子信息，包括{序号, 句子, 拼音信息}。拼音是由一个列表表示，每个中括号内是一个字的拼音。句子单词拼音给出的分析结果仅供参考，有可能出错。 *The input is in JSON format and includes a serial number, the sentence, and the pinyin information. The pinyin is represented as a list where each bracket contains the pinyin of a character. The pinyin analysis is provided for reference only and may contain errors.*
Data Description (All)	输入是JSON格式的句子信息，包括{序号, 句子, 依存句法关系, 词性, 拼音}。依存句法关系是由一系列元组表示，其格式为（单词序号，单词，当前单词句法依存于另一个单词的序号，LTP句法依存标签）。词性是由一系列元组表示，其格式为（单词，当前单词的LTP词性标签）。拼音是由一个列表表示，每个中括号内是一个字的拼音。给出的分析结果仅供参考，有可能出错。 *The input is in JSON format and includes a serial number, the sentence, dependency structure, POS tags, and pinyin. The dependencies are represented as tuples in the format (word index, word, head index, dependency label). The POS tags are represented as tuples in the format (word, POS tag). The pinyin is represented as a list where each bracket contains the pinyin of a character. The analysis results are provided for reference only and may contain errors.*

Table 6: Data Descriptions in Prompt for Chinese Grammar Error Detection and Correction Task.

Model	Feature	Sentence Count	TP	FP	FN	Prec	Rec	F0.5 (char)	F0.5 (word)
DeepSeek-R1	Baseline	908	1428	1620	1254	0.4685	0.5324	0.4800	0.4760
	Dependency	898	1273	1153	1167	0.5247	0.5217	0.5241	0.5177
	POS	895	1306	1056	1192	0.5529	0.5228	0.5466	0.5381
	Pinyin	897	1321	1122	1163	0.5407	0.5318	0.5389	0.5342
	All	892	1254	1130	1234	0.5260	0.5040	0.5215	0.5159
DeepSeek-V3	Baseline	820	1224	1054	1140	0.5373	0.5178	0.5333	0.5232
	Dependency	727	977	600	953	0.6195	0.5062	0.5930	0.5855
	POS	777	1037	546	980	0.6551	0.5141	0.6210	0.6152
	Pinyin	764	986	492	981	0.6671	0.5013	0.6257	0.6208
	All	646	814	668	917	0.5493	0.4702	0.5314	0.5213
GPT-4o	Baseline	812	1144	1050	1167	0.5214	0.4950	0.5159	0.5098
	Dependency	778	1012	873	1074	0.5369	0.4851	0.5257	0.5258
	POS	817	1083	841	1082	0.5629	0.5002	0.5491	0.5459
	Pinyin	781	1041	815	1112	0.5609	0.4835	0.5435	0.5358
	All	772	1054	852	1089	0.5530	0.4918	0.5396	0.5307
Claude 3.7 Sonnet	Baseline	839	1137	786	1130	0.5913	0.5015	0.5708	0.5649
	Dependency	799	901	590	1013	0.6043	0.4707	0.5718	0.5685
	POS	812	999	636	1036	0.6110	0.4909	0.5825	0.5789
	Pinyin	807	958	608	1078	0.6117	0.4705	0.5771	0.5710
	All	773	850	502	1021	0.6287	0.4543	0.5839	0.5735
LLaMa 3.3. 70B	Baseline	772	1104	2073	1304	0.3475	0.4585	0.3652	0.3510
	Dependency	670	983	2023	1125	0.3270	0.4663	0.3478	0.3370
	POS	704	998	1955	1119	0.3380	0.4714	0.3582	0.3506
	Pinyin	675	994	1898	1150	0.3437	0.4636	0.3625	0.3515
	All	616	820	1995	1116	0.2913	0.4236	0.3107	0.3007

Table 7: F0.5 Score for Edit Operations on Identified Erroneous Sentences.

Model	Feature	Sentence Count	TP	FP	FN	Prec	Rec	F0.5(char)	F0.5(word)
DeepSeek-R1	Baseline	351	603	672	524	0.4729	0.5350	0.4842	0.4798
	Dependency	351	559	508	495	0.5239	0.5304	0.5252	0.5131
	POS	351	573	461	528	0.5542	0.5204	0.5471	0.5361
	Pinyin	351	569	494	488	0.5353	0.5383	0.5359	0.5197
	All	351	536	506	549	0.5144	0.4940	0.5102	0.5062
DeepSeek-V3	Baseline	351	573	480	511	0.5442	0.5286	0.5410	0.5293
	Dependency	351	504	313	487	0.6169	0.5086	0.5917	0.5831
	POS	351	510	270	468	0.6538	0.5215	0.6223	0.6069
	Pinyin	351	490	268	477	0.6464	0.5067	0.6127	0.6028
	All	351	459	372	511	0.5523	0.4732	0.5345	0.5200
GPT-4o	Baseline	351	521	524	542	0.4986	0.4901	0.4969	0.4975
	Dependency	351	490	440	518	0.5269	0.4861	0.5182	0.5134
	POS	351	504	416	462	0.5478	0.5217	0.5424	0.5389
	Pinyin	351	485	425	496	0.5330	0.4944	0.5248	0.5147
	All	351	512	442	475	0.5367	0.5187	0.5330	0.5182
Claude 3.7 Sonnet	Baseline	351	550	368	541	0.5991	0.5041	0.5774	0.5663
	Dependency	351	444	296	488	0.6000	0.4764	0.5704	0.5539
	POS	351	501	327	503	0.6051	0.4990	0.5804	0.5686
	Pinyin	351	485	310	508	0.6101	0.4884	0.5811	0.5744
	All	351	445	278	496	0.6155	0.4729	0.5805	0.5687
LLaMa 3.3. 70B	Baseline	351	510	998	654	0.3382	0.4381	0.3544	0.3427
	Dependency	351	506	1052	656	0.3248	0.4355	0.3422	0.3270
	POS	351	514	989	585	0.3420	0.4677	0.3614	0.3541
	Pinyin	351	523	1010	656	0.3412	0.4436	0.3577	0.3467
	All	351	458	1068	626	0.3001	0.4225	0.3186	0.3047

Table 8: F0.5 Score for Edit Operations on Intersection Sentences.

Model	Feature	Sentence Count	TP	FP	FN	Prec	Rec	F0.5(char)	F0.5(word)
DeepSeek-R1	Baseline	1839	1438	3234	1309	0.3078	0.5235	0.3354	0.3287
	Dependency	1839	1286	2301	1253	0.3585	0.5065	0.3808	0.3750
	POS	1839	1315	2110	1255	0.3839	0.5117	0.4041	0.3978
	Pinyin	1839	1331	2166	1228	0.3806	0.5201	0.4022	0.3945
	All	1839	1270	2324	1343	0.3534	0.4860	0.3738	0.3680
DeepSeek-V3	Baseline	1839	1230	1709	1269	0.4185	0.4922	0.4314	0.4214
	Dependency	1839	996	934	1190	0.5161	0.4556	0.5027	0.4949
	POS	1839	1045	874	1160	0.5446	0.4739	0.5288	0.5219
	Pinyin	1839	998	783	1202	0.5604	0.4536	0.5352	0.5326
	All	1839	833	940	1305	0.4698	0.3896	0.4512	0.4416
GPT-4o	Baseline	1839	1151	1756	1319	0.3959	0.4660	0.4082	0.4015
	Dependency	1839	1019	1443	1282	0.4139	0.4429	0.4194	0.4163
	POS	1839	1092	1465	1247	0.4271	0.4669	0.4345	0.4313
	Pinyin	1839	1047	1355	1288	0.4359	0.4484	0.4383	0.4293
	All	1839	1063	1385	1292	0.4342	0.4514	0.4376	0.4268
Claude 3.7 Sonnet	Baseline	1839	1148	1463	1251	0.4397	0.4785	0.4469	0.4430
	Dependency	1839	913	1013	1204	0.4740	0.4313	0.4648	0.4608
	POS	1839	1007	1134	1189	0.4703	0.4586	0.4679	0.4657
	Pinyin	1839	971	1072	1256	0.4753	0.4360	0.4669	0.4599
	All	1839	859	860	1224	0.4997	0.4124	0.4794	0.4676
LLaMa 3.3. 70B	Baseline	1839	1109	3543	1489	0.2384	0.4269	0.2615	0.2537
	Dependency	1839	992	3347	1488	0.2286	0.4000	0.2501	0.2381
	POS	1839	1005	3158	1427	0.2414	0.4132	0.2633	0.2563
	Pinyin	1839	1001	3167	1493	0.2402	0.4014	0.2611	0.2532
	All	1839	824	3275	1521	0.2010	0.3514	0.2198	0.2116

Table 9: F0.5 Score for Edit Operations on All Sentences.

Model	Type	Prec.	Rec.	F0.5	TP	FP	FN	Gold	Pred	Err. Density	FP Rate	FN Rate	Aggr.	Cons.
DeepSeek-R1	S	0.502	0.525	0.507	223	221	202	425	444	17.71	0.498	0.475	0.991	0.906
DeepSeek-R1	M	0.559	0.436	0.529	171	135	221	392	306	16.33	0.441	0.564	0.789	1.292
DeepSeek-R1	R	0.631	0.626	0.630	149	87	89	238	236	9.92	0.369	0.374	0.584	0.597
DeepSeek-R1	W	0.686	0.449	0.621	35	16	43	78	51	3.25	0.314	0.551	0.457	1.229
DeepSeek-V3	S	0.609	0.507	0.585	190	122	185	375	312	15.63	0.391	0.493	0.642	0.974
DeepSeek-V3	M	0.628	0.479	0.592	159	94	173	332	253	13.83	0.372	0.521	0.591	1.088
DeepSeek-V3	R	0.759	0.589	0.717	132	42	92	224	174	9.33	0.241	0.411	0.318	0.697
DeepSeek-V3	W	0.762	0.525	0.699	32	10	29	61	42	2.54	0.238	0.475	0.313	0.906
Claude 3.7 Sonnet	S	0.594	0.468	0.563	184	126	209	393	310	16.38	0.406	0.532	0.685	1.136
Claude 3.7 Sonnet	M	0.549	0.486	0.535	169	139	179	348	308	14.50	0.451	0.514	0.822	1.059
Claude 3.7 Sonnet	R	0.733	0.583	0.697	126	46	90	216	172	9.00	0.267	0.417	0.365	0.714
Claude 3.7 Sonnet	W	0.615	0.381	0.548	24	15	39	63	39	2.63	0.385	0.619	0.625	1.625
GPT-4o	S	0.514	0.499	0.511	186	176	187	373	362	15.54	0.486	0.501	0.946	1.005
GPT-4o	M	0.531	0.494	0.523	165	146	169	334	311	13.92	0.469	0.506	0.885	1.024
GPT-4o	R	0.601	0.601	0.601	125	83	83	208	208	8.67	0.399	0.399	0.664	0.664
GPT-4o	W	0.744	0.453	0.659	29	10	35	64	39	2.67	0.256	0.547	0.345	1.207
LLaMA 3.3 70B	S	0.275	0.448	0.298	188	496	232	420	684	17.50	0.725	0.552	2.638	1.234
LLaMA 3.3 70B	M	0.351	0.427	0.364	168	310	225	393	478	16.38	0.649	0.573	1.845	1.339
LLaMA 3.3 70B	R	0.476	0.582	0.494	149	164	107	256	313	10.67	0.524	0.418	1.101	0.718
LLaMA 3.3 70B	W	0.393	0.204	0.331	11	17	43	54	28	2.25	0.607	0.796	1.545	3.909

Table 10: Model Performance Comparison across Error Types (S: Word Selection Errors, M: Missing Words, R: Redundant Words, W: Word Ordering Errors).

Table 11: Error Correction Examples from Intersection Sentences.

Erroneous Sentence	她每周参加打太极拳的时候，跑长跑去体育馆。 ***ta mei zhou can jia da tai ji quan de shi hou, pao chang pao qu ti yu guan*** **She runs long-distance to the gym while practicing Tai Chi every week.**	我从以前想去看这层建筑。 ***wo cong yi qian xiang qu kan zhe ceng jian zhu*** **I've wanted to see this building since a long time ago.**	他们什么时候我不懂帮助我。 ***ta men shen me shi hou wo bu dong bang zhu wo*** **They help me whenever I don't understand.**
Gold	1）她每周打太极拳的时候，跑长跑去体育馆。 *ta mei zhou da tai ji quan de shi hou, pao chang pao qu ti yu guan* When she practices Tai Chi every week, she runs long-distance to the gym. 2）她每周打太极的时候，长跑去体育馆。 *ta mei zhou da tai ji de shi hou, chang pao qu ti yu guan* When she practices Tai Chi, she runs long-distance to the gym. 3）她每周参加打太极拳的时候，都是跑长跑去体育馆。 *ta mei zhou can jia da tai ji quan de shi hou, dou shi pao chang pao qu ti yu guan* Whenever she participates in Tai Chi every week, she runs long-distance to the gym. 4）她在每周打太极拳的时候，都长跑去体育馆。 *ta zai mei zhou da tai ji quan de shi hou, dou chang pao qu ti yu guan* She always runs to the gym when doing Tai Chi every week. 5）她每周参加打太极拳活动的时候，跑去体育馆。 *ta mei zhou can jia da tai ji quan huo dong de shi hou, pao qu ti yu guan* When she joins Tai Chi activities each week, she runs to the gym. 6）她每周参加打太极拳的时候，都是跑步去体育馆。 *ta mei zhou can jia da tai ji quan de shi hou, dou shi pao bu qu ti yu guan* She always runs to the gym after practicing Tai Chi. 7）她每周打太极拳的时候，都跑去体育馆。 *ta mei zhou da tai ji quan de shi hou, dou pao qu ti yu guan* She always goes running to the gym when doing Tai Chi. 8）她每周打太极拳的时候，长跑去体育馆。 *ta mei zhou da tai ji quan de shi hou, chang pao qu ti yu guan* When she does Tai Chi, she runs long-distance to the gym. 9）她每周去打太极拳的时候，就长跑去体育馆。 *ta mei zhou qu da tai ji quan de shi hou, jiu chang pao qu ti yu guan* She goes for a run when she does Tai Chi each week. 10）她每周打太极拳的时候，跑长跑去体育馆。 *ta mei zhou da tai ji quan de shi hou, pao chang pao qu ti yu guan* She runs long-distance when practicing Tai Chi weekly.	1）我在以前就想去看这层建筑。 *wo zai yi qian jiu xiang qu kan zhe ceng jian zhu* I've wanted to see this building since a long time ago. 2）我从以前就想去看这座建筑。 *wo cong yi qian jiu xiang qu kan zhe zuo jian zhu* I had wanted to visit this building for a long time. 3）我以前就很想去看这座建筑。 *wo yi qian jiu hen xiang qu kan zhe zuo jian zhu* I had long wanted to see this building. 4）我以前就想去看这个建筑。 *wo yi qian jiu xiang qu kan zhe ge jian zhu* I've long wanted to see this building. 5）我以前就想来看看这栋建筑。 *wo yi qian jiu xiang lai kan kan zhe dong jian zhu* I've wanted to come see this building. 6）我从很久之前就想去看这层建筑。 *wo cong hen jiu zhi qian jiu xiang qu kan zhe ceng jian zhu* I've wanted to visit this floor for a long time. 7）我之前就想去看这座建筑。 *wo zhi qian jiu xiang qu kan zhe zuo jian zhu* I had already wanted to see this building. 8）我从很久以前就想去看这座建筑。 *wo cong hen jiu yi qian jiu xiang qu kan zhe zuo jian zhu* I've long wanted to see this building. 9）我以前就想去看这栋建筑。 *wo yi qian jiu xiang qu kan zhe dong jian zhu* I had wanted to see this building. 10）我从以前想去看这个建筑。 *wo cong yi qian xiang qu kan zhe ge jian zhu* I've wanted to see this building for a long time. 11）我以前就想去看看这个建筑。 *wo yi qian jiu xiang qu kan kan zhe ge jian zhu* I had wanted to take a look at this building for a long time.	1）什么时候我不懂了他们都帮助我。 *shen me shi hou wo bu dong le ta men dou bang zhu wo* Whenever I didn't understand, they all helped me. 2）什么时候我不懂了，他们会帮助我。 *shen me shi hou wo bu dong le, ta men hui bang zhu wo* Whenever I didn't understand, they would help me. 3）他们在我不懂的时候帮助我。 *ta men zai wo bu dong de shi hou bang zhu wo* They help me when I don't understand. 4）他们不论什么时候只要我不懂就会帮助我。 *ta men bu lun shen me shi hou zhi yao wo bu dong jiu hui bang zhu wo* They help me whenever I don't understand. 5）他们会在我有什么地方不懂的时候帮助我。 *ta men hui zai wo you shen me di fang bu dong de shi hou bang zhu wo* They help me when there's something I don't understand. 6）他们在我不懂什么的时候会帮助我。 *ta men zai wo bu dong shen me de shi hou hui bang zhu wo* They help me when I don't understand something. 7）他们在我不懂的任何时候帮助我。 *ta men zai wo bu dong de ren he shi hou bang zhu wo* They help me anytime I don't understand. 8）我什么时候不懂，他们就帮助我。 *wo shen me shi hou bu dong, ta men jiu bang zhu wo* Whenever I don't understand, they help me. 9）他们在任何时候都帮助我。 *ta men zai ren he shi hou dou bang zhu wo* They help me at any time. 10）什么时候我有不懂的，他们就会帮助我。 *shen me shi hou wo you bu dong de, ta men jiu hui bang zhu wo* Whenever I have questions, they help me.
DeepSeek-V3 (POS)	她每周参加打太极拳的时候，都会跑长跑去体育馆。 (F0.5 = 0) *ta mei zhou can jia da tai ji quan de shi hou, dou hui pao chang pao qu ti yu guan* She goes to the gym to run long-distance every week when doing Tai Chi.	我从以前就想去看这层建筑。 (F0.5 = 0.8333) *wo cong yi qian jiu xiang qu kan zhe ceng jian zhu* I've long wanted to visit this floor of the building.	他们什么时候我不懂帮助我。 (F0.5 = 0.7143) *ta men shen me shi hou wo bu dong bang zhu wo* They when I don't understand help me.

Table 11: Error Correction Examples from Intersection Sentences (Continued).

DeepSeek-R1 (POS)	她每周参加打太极拳的时候，还会去体育馆跑长跑。 (F0.5 = 0) *ta mei zhou can jia da tai ji quan de shi hou, hai hui qu ti yu guan pao chang pao* When attending Tai Chi every week, she also goes to the gym to run.	我早就想去看这栋建筑。 (F0.5 = 0.8333) *wo zao jiu xiang qu kan zhe dong jian zhu* I've long wanted to see this building.	他们什么时候帮助我，我不懂。 (F0.5 = 0.7143) *ta men shen me shi hou bang zhu wo, wo bu dong* I don't understand when they help me.
Claude 3.7 Sonnet (POS)	她每周参加打太极拳的时候，跑着长跑去体育馆。 (F0.5 = 0) *ta mei zhou can jia da tai ji quan de shi hou, pao zhe chang pao qu ti yu guan* She jogs to the gym while attending Tai Chi every week.	我从以前就想去看这座建筑。 (F0.5 = 1) *wo cong yi qian jiu xiang qu kan zhe zuo jian zhu* I've long wanted to see this building.	他们什么时候帮助我我不懂。 (F0.5 = 0.4545) *ta men shen me shi hou bang zhu wo wo bu dong* I don't understand when they help me.
GPT-4o (POS)	她每周去体育馆的时候，参加打太极拳和跑长跑。 (F0.5 = 0) *ta mei zhou qu ti yu guan de shi hou, can jia da tai ji quan he pao chang pao* Every week she goes to the gym to do Tai Chi and run.	我一直想去看这层建筑。 (F0.5 = 0.3333) *wo yi zhi xiang qu kan zhe ceng jian zhu* I've always wanted to see this floor of the building.	我不懂他们什么时候帮助我。 (F0.5 = 0) *wo bu dong ta men shen me shi hou bang zhu wo* I don't understand when they help me.
LLaMa 3.3 70B (POS)	她每周在参加完太极拳活动后，会去体育馆跑长跑。 (F0.5 = 0) *ta mei zhou zai can jia wan tai ji quan huo dong hou, hui qu ti yu guan pao chang pao* Every week after attending Tai Chi, she goes to the gym for a long run.	我以前就一直想去看这座建筑。 (F0.5 = 0) *wo yi qian jiu yi zhi xiang qu kan zhe zuo jian zhu* I've always wanted to see this building.	他们什幺时候帮助我我不懂。 (F0.5 = 0.6667) *ta men shen me shi hou bang zhu wo wo bu dong* I don't understand when they help me.