Instruction-Driven In-Context Learning for Domain-Specific Chinese Spelling Correction

Hyunsoo Park[1,2,3], **Hongqiu Wu**[1,2,3], **Hai Zhao**[1,2,3†]

[1] AGI Institute, School of Computer Science, Shanghai Jiao Tong University
[2] Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University
[3] Shanghai Key Laboratory of Trusted Data Circulation and Governance in Web3
hspark0925@sjtu.edu.cn, wuhongqiu@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn

Abstract

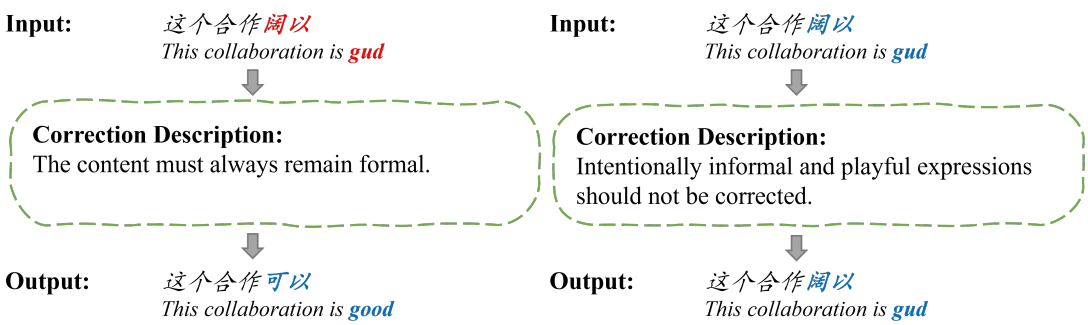
This paper investigates domain adaptation in Chinese Spelling Correction (CSC) based on the instruction-following ability of large language models (LLMs). In the instructions, we include a variety of domain-specific requirements for spelling correction, such as the domain's formality or writing tone, which go beyond the considerations of previous CSC research. To evaluate the LLMs' performance on instruction-following, we propose IDSpell, a semi-supervised construction pipeline for a CSC dataset containing a wide range of domain-specific sentences along with specific instructions. We construct a dataset with IDSpell and evaluate it on Qwen2.5 and GPT-4o, where we find that instructions serve a meaningful influence in correction, increasing the average F1 score by 10.4% compared to when the instructions are not provided. To further enhance the result, we propose Contrastive Prompting, a method incorporating contrastive false examples into the prompt to better guide the model to understand the instruction. Experiments demonstrate that our method outperforms baseline prompting with an average improvement of 5.4%. Our dataset and code are publicly available for further research.

Keywords: Chinese Spelling Correction , Domain Adaptation

1 Introduction

Chinese Spelling Correction (CSC) is a crucial task in Natural Language Processing (NLP) that identifies and corrects spelling errors within Chinese sentences. CSC plays a fundamental role in a series of NLP applications such as search query (Martins and Silva, 2004; Gao et al., 2010) or optical character recognition (Afli et al., 2016; Gupta et al., 2021). Since the applications are deployed in various domain-specific contexts, fully adapting CSC models to diverse real-world domains has been a long-time challenge. Numerous studies have tried to address this challenge by constructing CSC datasets with domain-specific terminologies (Lv et al., 2023; Wu et al., 2023b; Jiang et al., 2024). However, focusing exclusively on domain-specific corpora does not fully represent real-world scenarios. In practice, domains are defined not only by their corpus but also by their specific writing requirements. For instance, while the legal domain requires strict adherence to formal writing standards, a social media domain may expect intentional deviations from standard spelling. These requirements vary significantly and often conflict with each other across domains, yet cannot be properly conveyed by merely expanding the range of vocabularies for each domain. Figure 1 demonstrates how a CSC model should perform differently to the same input sentence when adapted to two different domain-specific requirements, emphasizing the importance of the requirements on avoiding unnecessary or even misleading spelling corrections. In this work, we leverage the instruction-following ability of large language models (LLMs) to explore these previously unaddressed domain-specific requirements, specifically by providing the requirements in the form of correction descriptions like in Figure 1 as part of the prompt.

† Corresponding author. This research was supported by the Joint Research Project of Yangtze River Delta Science and Technology Innovation Community (No. 2022CSJGG1400).



(a) Business Email Domain (b) Social Media Domain

Figure 1: Given the same input, in the (a) Business email domain, "阔以" is corrected to "可以" for formality, while in the (b) Social Media domain, "阔以" is accepted for its intentional informal tone.

Prompting on LLMs is not a mainstream direction in CSC, where the dominant approaches are based on fine-tuning BERT (Devlin et al., 2019) style pre-trained language models. Previous CSC studies show that LLM-based approaches significantly underperform compared to BERT-based, often over-correcting (Wu et al., 2023a; Wang et al., 2024) and struggling with mapping phonetic pronunciations to Chinese characters (Li et al., 2023). However, a fatal limitation of the BERT-based methods is that they lack generalizability to other tasks. BERT-based methods define CSC as a sequence tagging task, where characters in a sentence are mapped to their corresponding characters in the reference sentence (Wang et al., 2018; Wang et al., 2019). This structure strictly confines these models to tagging-based tasks. For instance, BERT-based CSC methods cannot be generalized to multilingual spelling correction, as spelling errors in other languages often do not have character-to-character mapping, such as in English "silable → syllable" or Korean "안된다→ 안*[space]*된다," where misspellings frequently arise from spacing issues. Domain adaption in CSC also cannot be defined as a tagging-based task when considering domain-specific requirements, where many erroneous word pairs do not align with a strict character-level mapping (details on section 3.2). Therefore, exploring domain-specific requirements in CSC using the instruction-following capability of LLMs can be a good first step in generalizing CSC beyond the constraints of BERT-based tagging methods.

To achieve this, we introduce ***IDSpell***, an ***I****nstruction-****D****riven dataset construction pipeline for* ***D****omain-Specific Chinese* ***Spell****ing Correction*. This semi-supervised pipeline of 5 steps constructs a CSC dataset consisting of domain-specific sentences, each paired with an instruction describing the domain-specific requirement. Using the pipeline, we construct a dataset and experiment with it on two autoregressive LLMs (QWEN2.5 and GPT-4o), where the instruction poses impactful performance improvement on both models. To further enhance the performance, we propose *Contrastive Prompting*, where a contrastive false example is provided along with each instruction in the few-shot examples. Contrastive prompting shows the best performance among all the prompting modes, with further analysis revealing the effectiveness of false cases on the models' understanding of the instructions.

Our contributions are summarized below:

- To the best of our knowledge, we are the first to explore generalizing CSC with the instruction-following ability of LLMs.
- We introduce a semi-supervised dataset construction pipeline for domain-specific CSC with instructions. We also present an accompanying dataset.
- We propose contrastive prompting for CSC with instructions, demonstrating the effectiveness of contrastive examples for future CSC research.

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095–1107, Jinan, China, August 11–14, 2025.

2 Related Works

2.1 Domain Adaptation in CSC

Domain adaptation refers to a model's ability to generalize and perform effectively across specialized domains, each characterized by its unique terminology and contextual nuances. Despite being trained on large quantities of data, pre-trained language models like BERT (Devlin et al., 2019) frequently perform poorly on downstream tasks because their training datasets lack adequate domain specialization and depth. This challenge is also evident in the widely-used CSC benchmark, SIGHAN (Wu et al., 2013; Yu et al., 2014; Tseng et al., 2015), which is annotated by Chinese learners as a Foreign Language (CFL), and thus contains minimal domain-specific vocabulary. Recognizing this limitation, Lv (2023) introduces the first annotated multi-domain CSC dataset, featuring manually curated data from three distinct domains: Law, Medical Treatment, and Official Document Writing. Further research expands upon this concept, incorporating a broader range of domains with domain-specific terminologies (Wu et al., 2023b; Jiang et al., 2024). However, these datasets still do not fully address domain adaptation in terms of domain-specific requirements beyond terminology.

2.2 Autoregressive LLMs in CSC

Recent state-of-the-art CSC methods predominantly use BERT (Devlin et al., 2019) as their backbone due to its strength in sequence tagging tasks (Zhang et al., 2020; Zhu et al., 2022; Huang et al., 2021). Liu (2024) introduces a rephrasing masking strategy on top of BERT's architecture, similar to autoregressive LLMs' inference in requiring full-sentence prediction but differing in that all characters are predicted simultaneously. Despite its similarities, this approach remains within the BERT framework. Few significant models are proposed that leverage autoregressive frameworks for CSC, while many studies indicate LLMs underperform on CSC compared to BERT-based methods (Li and Shi, 2021; Hu et al., 2022; Jiang et al., 2024). Some studies reveal their fluent yet over-corrected predictions as the reason behind the under-performance (Wu et al., 2023a; Wang et al., 2024). A recent study (Li et al., 2024) mitigates over-correction by introducing character-level tokenization on QWEN1.5 (Bai et al., 2023), an open-source Chinese LLM, achieving notable improvements. However, this approach still requires significant computation for re-tokenization and continued pre-training, and it still does not fully cherish the generalizability that LLMs could offer in CSC.

3 Representing Domain Requirements

To leverage the instruction-following ability of LLMs, we represent the domain-specific requirements by two means: 1) correction descriptions for in-context learning and 2) keyword pairs in the source-reference sentences for evaluation.

3.1 Correction Descriptions

Correction descriptions are sentences that illustrate unique requirements for various domains. Each correction description is paired with one domain and describes one requirement of the paired domain. They are written in an instructive manner, such as "You must always correct . . . " or ". . . can be retained when...". Step 2 in Figure 2 shows an example of a correction description for the news domain, instructing to pay attention to the formality of the context. These instructions are provided within the prompts.

3.2 Keyword Pairs

Keyword pairs are two words the instruction directs attention to, found in source and reference sentences. For instance, the instruction in the final dataset outline of Figure 2 specifies how to handle proper nouns within the domain. In this case, the keyword pairs are proper nouns, one in its misspelled form and the other in its corrected form, each found in the input and output sentences. These keyword pairs are then used in evaluation by checking whether the correct keywords are found in the predictions.

Keyword pairs are a new concept distinct from the edit pairs typically used in previous CSC studies (Wu et al., 2023b), which focus on character-level correction pairs. Keyword pairs differ from edit pairs in three aspects: 1) word-level definition rather than character-level, 2) flexible character count

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095–1107, Jinan, China, August 11–14, 2025.

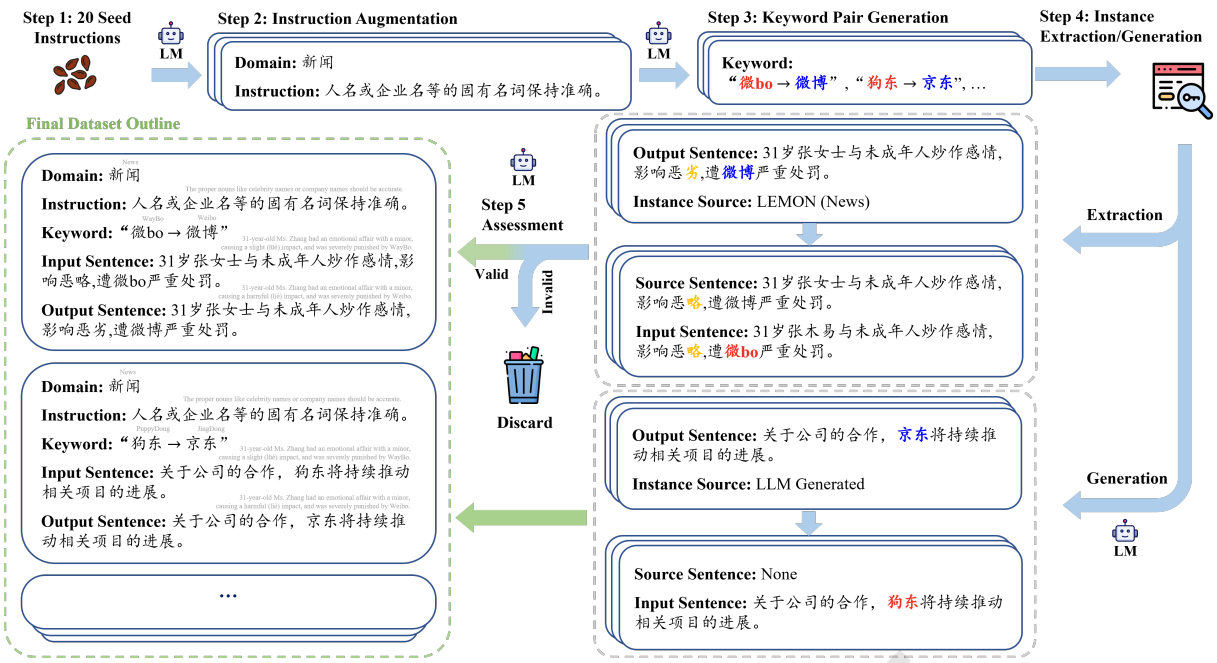


Figure 2: Overview of IDSpell. The final dataset is highlighted with a dotted outline on the left, with English translations accompanied in a smaller font for reference.

between paired words and 3) account for potential identity between two words, to emphasize preventing over-correction. Traditional CSC edit pairs cannot adequately capture keyword pairs, leading to undetected or inappropriate corrections.

For example, the Chinese word “报” (bào, meaning “to report”) is commonly used on social media as a playful substitute for “不好”(bù hǎo, meaning “not good”) due to phonetic similarity, despite their distinct meanings. However, this substitution would most likely be a spelling error in formal domains such as official documents. Therefore, the keyword pair for this case is “报→ 报” in the social media domain, when it is “报→ 不好” for the official document domain.

The formal definition of keyword pairs is given as follows. Given a source sentence X and a reference sentence Y', a keyword pair is defined as $T_X \rightarrow T_{Y'}$, where $T_X = x_{i,j}$ and $T_{Y'} = y'_{i,k}$. Here, i denotes the position of the first character, while j and k denote the positions of the last characters of the keywords in the source and reference sentences, respectively. The conventional definition of CSC restricts T_X and $T_{Y'}$ to be equal in length (i.e., $k = j$). Keyword pairs remove this constraint, allowing k to be any value regardless of the value of j.

3.3 Task Definition

Applying keyword pairs, the new task definition is as follows:

$$X = \{x_1, \cdots, x_{i-1}, x_{i,j}, x_{j+1}, \cdots, x_n\} \quad (1)$$

$$Y = \{y_1, \cdots, y_{i-1}, y_{i,j}, y_{j+1}, \cdots, y_n\} \quad (2)$$

$$Y' = \{y_1, \cdots, y_{i-1}, y'_{i,k}, y_{j+1}, \cdots, y_n\} \quad (3)$$

Given the source sentence X (1), Y (2) is the corrected sentence under the tagging-based CSC definition, while Y' (3) is the corrected sentence with keyword pairs. Y has character-level mapping with X, while for X to Y', the keyword pair $x_{i,j} \rightarrow y'_{i,k}$ (from Section 3.2) does not align character to character. From X to Y', T_X is replaced by $T_{Y'}$, while correcting character-level misspellings elsewhere.

This modification allows greater flexibility in spelling correction. However, as character-level mapping becomes impossible, ***CSC can no longer be treated as a sequence tagging problem*** under this definition, making previous BERT-based tagging methods inapplicable.

Data Source	Description	Instances
SIGHAN15 (Tseng et al., 2015)	The most commonly used CSC benchmark dataset, collected from Chinese essays written by foreign speakers.	5
CSCD-NS (Hu et al., 2022)	A CSC dataset designed for native speakers, selectively collected from Weibo posts from official accounts.	82
EC-SPELL (Lv et al., 2023)	A small-scale multi-domain CSC dataset focusing on legal, medical, and official document writing.	6
LEMON (Wu et al., 2023b)	A large-scale multi-domain CSC dataset of 7 different domains, annotated from various real-world spelling errors.	36
WEIBO[1]	A widely used Chinese social media platform. Instances are collected from posts containing keywords, excluding posts with user mentions, and restricting keywords to be within the main content.	163
GPT-4 (Achiam et al., 2023)	LLM generated instances from the keyword pairs without matches found in the predefined sources.	110
Total		**402**

Table 1: Overview of the data sources for IDSpell constructed dataset.

4 Methods

4.1 IDSpell

We present IDSpell, a semi-supervised pipeline for constructing instruction-following datasets for CSC. All unsupervised components within the pipeline are implemented using GPT-4 (Achiam et al., 2023). As depicted in Figure 2, the pipeline consists of five primary stages.

1. **Seed Instruction Annotation**: Human annotators curate a set of 20 seed instructions, each paired with one domain. A diverse set of instructions spanning multiple domains is selected for coverage. The seed instructions curated for the final dataset in this paper cover 10 different domains with different domain-specific requirements from each other.

2. **Instruction Augmentation**: A pre-trained language model (LM) augments the seed data into 100 correction descriptions. No constraints are imposed on domain selection to maximize domain diversity.

3. **Keyword Pair Generation**: Given the correction description and the domain, the LM generates five keyword pairs that are distinct from each other.

4. **Instance Extraction and Generation**: For each correct keyword from the generated keyword pairs, instances containing the keyword are searched iteratively across predefined data sources, ranging from existing CSC datasets to real-world textual data. If no suitable instances are found, then LM generates new instances that incorporate the keyword, given the context of the instruction. For extracted sentences with parallel source sentences, all the correct keywords in the source sentence are replaced with their respective misspelled keyword to construct the final input sentence. For the generated sentences and the extracted sentences without parallel source sentences, keyword replacement is performed directly on the reference sentence.

5. **Assessment**: Finally, for the extracted instances, the LM evaluates whether each instance aligns with the paired domain and instruction, discarding the invalid ones that are either not within the context of the given domain or do not correlate with the instruction. After the final automated validation, human annotators conduct a final review to ensure dataset consistency.

[1]Weibo: https://m.weibo.cn/

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095-1107, Jinan, China, August 11-14, 2025.

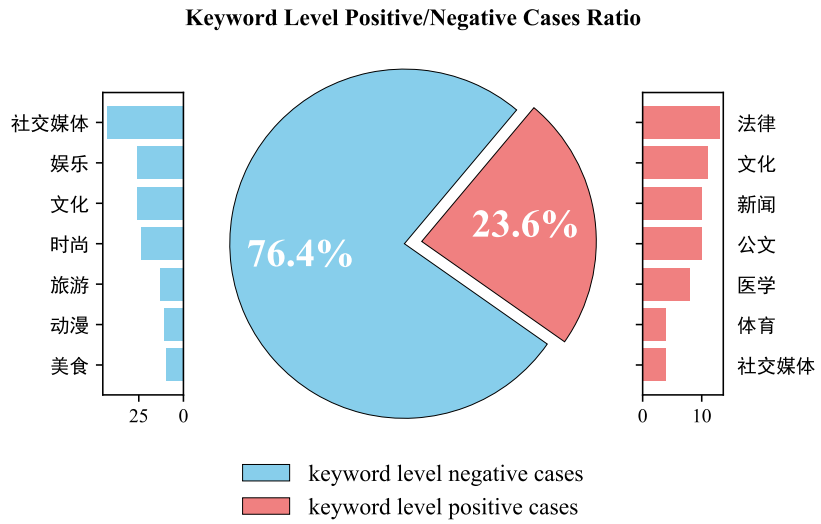


Figure 3: Ratio of instances with positive and negative keyword pairs across domains. The top seven domains with the most instances are listed on the left and right sides.

Through the pipeline, we construct a dataset for experiments in this paper. We select some existing CSC datasets and a real-world social media platform as the data sources, as listed in Table 1 with descriptions and the number of instances they contribute to our dataset. The final dataset contains 402 input-output instances with 87 instructions across 41 domains. It is characterized by 204 unique keyword pairs and 121 character-level edit pairs.

4.2 Contrastive Prompting

We present contrastive prompting, which employs contrastive false keywords within the instructions. This method is grounded in the following hypothesis.

Hypothesis Rooted in the inherent characteristic of domain-specific requirements in CSC, which emphasizes preventing over-correction, a higher number of instances involve negative keyword pairs, which consist of two identical keywords, rather than positive keyword pairs, which consist of two different keywords. We find that over 70% of instances have negative keyword pairs (Figure 3). Notably, instances with negative pairs are predominantly concentrated in informal domains such as “Entertainment” and “Fashion”, while positive keyword pairs are more clustered in formal domains like “Law” and “News”. Building on this finding, which highlights the impact of keyword pairs on domain characteristics, we hypothesize that incorporating additional dimensions into the keyword pairs by providing both positive and negative forms could further enhance the model’s understanding of domain-specific requirements.

Contrastive False Keywords Based on the hypothesis, we generate a contrastive false case for each instance, where the correct keyword in the reference sentence is replaced with an incorrect one. For the keyword pairs of two different words, the incorrect keyword is directly used to replace the correct keyword. For keyword pairs with two identical words, we employ the LLM to generate the incorrect keyword by inferring a correction for the keyword without being provided the corresponding instruction or domain. Generated false keywords semantically align with the correct keywords, but fail to adhere to the domain-specific requirements. These contrastive false cases are included within the few-shot examples during prompting. The implementation of false keyword generation is detailed in the Appendix A.2.

5 Experiments

In this section, we evaluate the performance of different LLMs on constructed IDSpell dataset from section 4.1 and compare different prompting methods.

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095–1107, Jinan, China, August 11–14, 2025.

	Mode	Keyword Level						Sentence Level					
		Qwen2.5 7B			GPT 4o mini			Qwen2.5 7B			GPT 4o mini		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
IC=0	Inst.	23.3	49.7	31.7	42.6	51.7	46.7	11.3	24.1	15.4	34.1	41.4	37.4
	Non-Inst.	20.0	46.2	27.9	35.7	48.3	41.1	10.1	23.4	14.2	28.1	37.9	32.3
IC=3	Inst.	35.1	49.0	40.9	54.9	53.8	54.4	17.8	24.8	20.7	43.0	42.1	42.5
	Non-Inst.	29.6	**51.7**	37.7	45.5	51.7	48.4	17.4	**30.3**	22.1	37.6	42.8	40.0
	Contrast.	**37.4**	51.0	**43.1**	**58.2**	**56.6**	**57.3**	**20.7**	28.3	**23.9**	**46.8**	**45.5**	**46.2**

Table 2: LLMs' performance on the ICSpell dataset. Three prompting modes are 1) **Inst.**: Instructions are provided, 2) **Non-Inst.**: Instructions are not provided, and 3) **Contrast.**: Instructions are provided with contrastive false cases in few-shot examples. These abbreviations for prompting modes stay consistent in the following tables and figures. *IC* indicates the number of few-shot examples provided during inference (*IC=0* for zero-shot, *IC=3* for few-shot).

5.1 Experimental Settings

Models Two baseline autoregressive LLMs are evaluated. (1) *QWEN2.5* (Bai et al., 2023): One of the strongest open-source Chinese LLMs, available in sizes ranging from 0.5B to 72B parameters. We use the instruction-tuned versions of the models for the experiments. (2) *GPT-4o mini* (Achiam et al., 2023): A more compact version of GPT-4o, a state-of-the-art LLM variant based on an autoregressive transformer architecture.

Prompting Modes The LLMs are experimented on three different prompting modes, where 1) the input sentence is presented without the domain and instruction, and 2) the domain and instruction are given before the input sentence. Additionally, in the few-shot setting, we also evaluate contrastive prompting where 3) the domain is specified with the instruction, followed by contrastive false cases in the few-shot examples. For the few-shot examples, priority was given to instances from the same domain but with a different keyword pair (instances with the same keyword pair are completely excluded from the few-shot examples). Detailed prompts are in Appendix A.1.

Evaluation Metric To focus on the model's understanding of instructions, we report precision, recall, and F1 scores at both the keyword and sentence-level evaluations. For the keyword-level evaluation, a prediction is considered correct if the reference keyword and all character-level edit pairs are included, regardless of the surrounding context.

5.2 Main Results

Table 2 presents the main results comparing the performance of the LLMs across each prompting mode under zero-shot and few-shot settings. The results are reported in both the keyword level and the sentence level evaluation.

In the zero-shot setting, both models consistently demonstrate improved performance when the instruction is explicitly given with the input sentence compared to when they are omitted, indicating that the instructions help the models better align with the domain-specific requirements. The performance further improves in the few-shot setting, where the improvement is consistent across most metrics, except for a slight decline of 0.7 in keyword-level recall on the QWEN2.5 7B when instructions are provided. In the few-shot setting, contrastive prompting achieves the highest F1 scores on both models, with GPT-4o mini reaching a peak of 57.3 in the keyword-level evaluation. Among the three prompting modes, prompting with only instructions ranks second, while the model consistently performs the lowest when instruction is not provided.

Precisions follow the same pattern as F1 scores, with contrastive prompting mode reaching the highest precision, while performance is lowest when no instruction is provided. However, recalls show a slight

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095–1107, Jinan, China, August 11–14, 2025.

Table 3: F1 scores of BERT-based tagging methods on IDSpell dataset.

Model	Keyword Level	Sentence Level	
	IDSpell	IDSpell	SIGHAN15
BERT	20.9	15.7	53.1
Soft-Masked BERT	22.9	16.8	52.9
ReLM	18.7	12.6	**58.6**
GPT 4.1 *Non-Inst.*	50.1	38.3	37.7
GPT 4.1 *Inst.*	58.9	43.9	–
GPT 4.1 *Contrast*	**66.9**	**56.3**	–

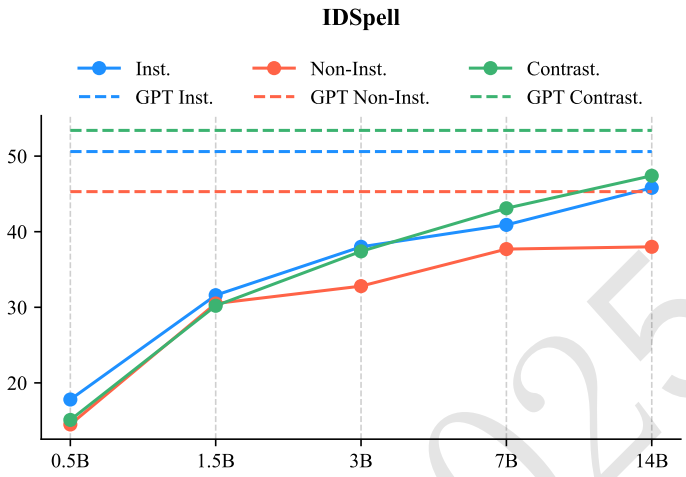


Figure 4: F1 score across three prompting modes on Qwen2.5 models of different parameter sizes. GPT-4o mini's performance is shown across the graph as a comparison. The experimental settings are the same with the 3-shot evaluation in the main result.

deviation from this trend. For GPT-4o Mini, the recall aligns similarly to the precision and F1 score patterns, with contrastive prompting getting the best results. However, the recall is unexpectedly the highest for QWEN2.5 7B when instructions are not provided, followed by contrastive prompting, and lowest when instruction is included.

Keyword-level evaluations are consistently higher than sentence-level performance, which suggests that the potential of LLMs in domain understanding for CSC tasks can be shadowed when only evaluated holistically by sentence-level comparison.

6 Further Analysis

6.1 Comparative Evaluation and Scaling

We evaluated three BERT-based tagging methods on IDSpell dataset. The BERT-based methods include pre-trained BERT (Devlin et al., 2019), which is applied to CSC as a sequence tagging task, the soft-masked BERT (Zhang et al., 2020), which employs an error detection model to provide a better representation of the `[MASK]` token, and ReLM (Liu et al., 2024), which rephrases the sentence on top of a tagging-based structure. We use the pre-trained models open-sourced by Wu (2023b) for all three BERT-based models, each trained on 34 million synthesized sentence pairs from `wiki2019zh` and `news2016zh` for CSC.

Table 3 shows the performance of three BERT-based tagging methods along with GPT-4.1 (Achiam et al., 2023), evaluated under three prompting modes. While BERT-based tagging models show decent performance on SIGHAN15 (Tseng et al., 2015), their results on IDSpell are notably poor. GPT-4.1 performs worse on SIGHAN15, consistent with known limitations of autoregressive models in traditional

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095–1107, Jinan, China, August 11–14, 2025.

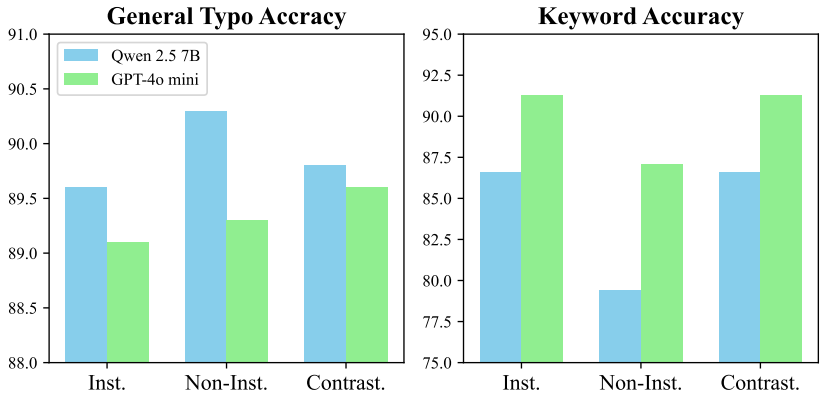


Figure 5: Accuracy comparison of LLMs in correcting general character-level typos and keywords across three prompting modes. The experimental setting is the same as the 3-shot evaluation in the main result.

Keywords	QWEN2.5 7B	GPT 4o mini
0	75.6%	81.5%
1	77.3%	89.1%
2	80.7%	89.9%
3	**84.0%**	**91.6%**
1 *(w/o. label)*	76.5%	85.7%
2 *(w/o. label)*	77.3%	88.2%
3 *(w/o. label)*	78.2%	89.9%

Table 4: Keyword correction accuracy, given increasing numbers of false keyword pairs as few-shot. "w/o. label" indicates that only the false keyword was provided without the correct keyword.

CSC (Li and Shi, 2021; Wu et al., 2023a; Wang et al., 2024). However, it significantly outperforms all BERT-based baselines on IDSpell dataset across all prompting modes, particularly in the contrastive setting, which highlights the limitations of tagging-based approaches as well as the potential of LLMs in domain-specific CSC.

Figure 4 presents the performances of QWEN2.5 models with parameter sizes ranging from 0.5B to 14B. These results are compared with those of GPT-4o mini in three prompting modes, all evaluated under identical experimental settings. Although the exact parameter sizes of the GPT models are not disclosed, the comparison provides useful insights. A consistent increase in performance is observed across all prompting modes as the size of the QWEN2.5 models increases. While the 14B model still performs slightly below the GPT-4o mini overall, the steady improvement trend suggests the potential to surpass GPT-4o mini with larger model scales. Notably, the performance of GPT-4o mini without instructions is exceeded by the Qwen2.5 with 14B parameters in the other two prompting modes.

6.2 Effect of Contrastive Examples

Our analysis reveals that the three prompting modes show consistent patterns in correcting keywords and character-level typos. As shown in Figure 5, both LLMs achieve higher keyword accuracy when instructions are included. For general character-level typos, Qwen performs the best without instructions, while GPT performs best with contrastive few-shot examples, both performing the worst when only instructions are given. These results suggest that instructions improve keyword understanding but may impede general typo correction, while contrastive prompting balances both.

Table 4 further investigates the impact of contrastive false examples on keyword understanding. We select 10 instructions from the IDSpell dataset with more than four distinct keyword pairs, resulting in 119 instances with 54 unique keywords. The model was evaluated by progressively increasing the number of keyword pairs from a zero-shot setting to three keywords. Additionally, we test the effect

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095–1107, Jinan, China, August 11–14, 2025.

domain	News
instruction	Proper nouns like names and locations should be kept accurate.
source	... 深夜在微bo晒照片，网友不谈定了
target	... 深夜在微博晒照片，网友不淡定了
Inst.	... 深夜在微博晒照片，网友不谈定了
domain	Social Media
instruction	Identify and retain deliberately used colloquial expressions.
source	早上好，吃了嘛你呐?
target	早上好，吃了嘛你呐?
Non-Inst.	早上好，吃了吗你呐?

Table 5: Limitations of prompting modes.

target	营业最多的人已当上时间管理大师美美幸福了，...
pred.	营业最多的人已经当上时间管理大师，美美幸福了，...
target	啊啊啊啊遇见你是我最大的小美好！！
pred.	啊啊啊啊遇见你是我最大的小美满！！

Table 6: Cases of over-correction

of providing only the false keyword from the keyword pair to assess the sole impact of false examples without the correct label on keyword understanding. Results demonstrate improved accuracy with increasing keyword pairs in both complete keyword pair and false-keyword-only settings, indicating that contrastive examples significantly enhance keyword comprehension and contribute to the effectiveness of the contrastive prompting mode.

6.3 Case Study

Table 5 illustrates the limitations of different prompting modes. In the first example, when instruction is provided, the model correctly fixes the keyword "微bo" to "微博," yet fails to correct the general typo "谈." The second example shows the model over-correcting an intentional informal expression "吃了嘛" to "吃了吗" when the instruction is not provided. However, when presented with few-shot contrastive examples, the model successfully handles both cases.

Table 6 demonstrates common over-correction patterns. Despite accurate keyword-level corrections, these cases are evaluated as incorrect at the sentence level. Examples include adding grammatical elements (e.g., adding "经" and a comma) and substituting correct words with synonyms (e.g., "好" to "满"). While few-shot learning reduces these over-corrections, they persist across all prompting modes.

7 Conclusion

In this work, we investigate domain adaptation in Chinese Spelling Correction (CSC) in a more generalizable manner by focusing on the instruction-following ability of Large Language Models (LLMs). Our experiments with contrastive prompting demonstrate superior performance across two different evaluation metrics, achieving a keyword-level F1 score of 57.3 with GPT-4o mini, which is 15.5% higher than when the input sentences are given without correction descriptions, and 5.1% higher than when the correction descriptions are given. Further analysis reveals that contrastive false keywords contribute significantly to this improvement in performance. Additionally, our comparative evaluation demonstrates the weakness of BERT-based tagging methods in instruction-following CSC. Our findings establish a foundation for developing more nuanced, domain-aware CSC models that better serve real-world applications, while highlighting the potential of LLMs in CSC research.

Proceedings of the 24th China National Conference on Computational Linguistics, pages 1095-1107, Jinan, China, August 11-14, 2025.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Haithem Afli, Zhengwei Qui, Andy Way, and Páraic Sheridan. 2016. Using smt for ocr error correction of historical texts.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Jianfeng Gao, Chris Quirk, et al. 2010. A large scale ranker-based system for search query spelling correction. In *The 23rd international conference on computational linguistics*.

Harsh Gupta, Luciano Del Corro, Samuel Broscheit, Johannes Hoffart, and Eliot Brenner. 2021. Unsupervised multi-view post-OCR error correction with language models. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8647–8652, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

Yong Hu, Fandong Meng, and Jie Zhou. 2022. Cscd-ime: correcting spelling errors generated by pinyin ime. *arXiv preprint arXiv:2211.08788*.

Li Huang, Junjie Li, Weiwei Jiang, Zhiyu Zhang, Minchuan Chen, Shaojun Wang, and Jing Xiao. 2021. PHMOSpell: Phonological and morphological knowledge guided Chinese spelling check. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5958–5967, Online, August. Association for Computational Linguistics.

Lai Jiang, Hongqiu Wu, Hai Zhao, and Min Zhang. 2024. Chinese spelling corrector is just a language learner. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6933–6943.

Piji Li and Shuming Shi. 2021. Tail-to-tail non-autoregressive sequence prediction for chinese grammatical error correction. *arXiv preprint arXiv:2106.01609*.

Yinghui Li, Haojing Huang, Shirong Ma, Yong Jiang, Yangning Li, Feng Zhou, Hai-Tao Zheng, and Qingyu Zhou. 2023. On the (in) effectiveness of large language models for chinese text correction. *arXiv preprint arXiv:2307.09007*.

Kunting Li, Yong Hu, Liang He, Fandong Meng, and Jie Zhou. 2024. C-llm: Learn to check chinese spelling errors character by character. *arXiv preprint arXiv:2406.16536*.

Linfeng Liu, Hongqiu Wu, and Hai Zhao. 2024. Chinese spelling correction as rephrasing language model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18662–18670.

Qi Lv, Ziqiang Cao, Lei Geng, Chunhui Ai, Xu Yan, and Guohong Fu. 2023. General and domain-adaptive chinese spelling check with error-consistent pretraining. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(5):1–18.

Bruno Martins and Mário J Silva. 2004. Spelling correction for search engine queries. In *Advances in Natural Language Processing: 4th International Conference, EsTAL 2004, Alicante, Spain, October 20-22, 2004. Proceedings 4*, pages 372–383. Springer.

Yuen-Hsien Tseng, Lung-Hao Lee, Li-Ping Chang, and Hsin-Hsi Chen. 2015. Introduction to sighan 2015 bake-off for chinese spelling check. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing*, pages 32–37.

Dingmin Wang, Yan Song, Jing Li, Jialong Han, and Haisong Zhang. 2018. A hybrid approach to automatic corpus generation for chinese spelling check. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2517–2527.

Dingmin Wang, Yi Tay, and Li Zhong. 2019. Confusionset-guided pointer networks for Chinese spelling check. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5780–5785, Florence, Italy, July. Association for Computational Linguistics.

Xi Wang, Ruoqing Zhao, Hongliang Dai, and Piji Li. 2024. An empirical investigation of domain adaptation ability for chinese spelling check models. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9996–10000. IEEE.

Shih-Hung Wu, Chao-Lin Liu, and Lung-Hao Lee. 2013. Chinese spelling check evaluation at sighan bake-off 2013. In *Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing*, pages 35–42.

Haoran Wu, Wenxuan Wang, Yuxuan Wan, Wenxiang Jiao, and Michael Lyu. 2023a. Chatgpt or grammarly? evaluating chatgpt on grammatical error correction benchmark. *arXiv preprint arXiv:2303.13648*.

Hongqiu Wu, Shaohua Zhang, Yuchen Zhang, and Hai Zhao. 2023b. Rethinking masked language modeling for chinese spelling correction. *arXiv preprint arXiv:2305.17721*.

Liang-Chih Yu, Lung-Hao Lee, Yuen-Hsien Tseng, and Hsin-Hsi Chen. 2014. Overview of sighan 2014 bake-off for chinese spelling check. In *Proceedings of The Third CIPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 126–132.

Shaohua Zhang, Haoran Huang, Jicong Liu, and Hang Li. 2020. Spelling error correction with soft-masked BERT. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 882–890, Online, July. Association for Computational Linguistics.

Chenxi Zhu, Ziqiang Ying, Boyu Zhang, and Feng Mao. 2022. MDCSpell: A multi-task detector-corrector framework for Chinese spelling correction. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1244–1253, Dublin, Ireland, May. Association for Computational Linguistics.

A Prompts

A.1 Prompting Modes

All prompts for the experiments follow the general structure in Table 7, where the section {`Prompting Mode`} varies across prompting modes as in Table 8.

Table 7: General Prompt Structure (Shared Across All Modes)

General Structure	
Chinese	你是一位精通中文的人，对中文的各个领域都有深入的理解。 {`Prompting Mode`} 输入文本: {`input`} 正确输出:
Translation	You are a fluent Chinese speaker with deep understanding across various domains. {`Prompting Mode`} Input: {`input`} Correct Output:

A.2 Contrastive False Keyword Generation

The QWEN2.5 7B instruct model is used for the contrastive keyword generation. Positive keyword pairs from the dataset are provided as few-shot examples to guide the model in generating words that are different from, yet remain semantically identical to, the given keyword. The inference is iterated until the model's prediction deviates from the original keyword, progressively increasing the temperature, top-k, and top-p values to enhance the diversity of the generated output. We then apply post-processing to refine the quality of the generated false keyword pairs. Prompt used for false keyword generation is at Table 9.

Table 8: Prompting templates in Chinese and English.

Instruction	
Chinese	以下是{domain}领域中输入的文本。请纠正输入文本中的错别字，然后直接输出纠正后的文本。如果错别字不存在，直接输出原本输入。注意，{instruction}
Translation	The following is an input text from the {domain} domain. Please correct any typos and output the corrected sentence directly. If there are no typos, copy the original input. Caution: {instruction}
Non-Instruction	
Chinese	请纠正输入文本中的错别字，然后直接输出纠正后的文本。如果错别字不存在，直接输出原本输入。
Translation	Please correct any typos in the input and output the corrected sentence directly. If there are no typos, copy the original input.
Contrastive	
Chinese	以下是{domain}领域中输入的文本。请纠正输入文本中的错别字，然后直接输出纠正后的文本。如果错别字不存在，直接输出原本输入。注意，{instruction} 比如，“{false_output}”是一个纠正错误的示例。
Translation	The following is an input text from the {domain} domain. Please correct any typos and output the corrected sentence directly. If there are no typos, copy the original input. Caution: {instruction} For example, “{false_output}” is an example of an incorrect typo correction.

Table 9: Contrastive False Case Generation Prompt)

Contrastive False Cases Generation	
Chinese	参考以下给定的上下文，直接输出指定关键词的纠正方案。 上下文: {input} 关键词: {keyword}
Translation	Given the context, output the correction for the specified keyword. Context: {input} Keyword: {keyword}

