RACQC: Advanced Retrieval-Augmented Generation for Chinese Query Correction

Jinbo Su[*1,2], **Lingzhe Gao**[1], **Wei Li**[1], **Shihao Liu**[1], **Haojie Lei**[1], **Xinyi Wang**[3], **Yuanzhao Guo**[4], **Ke Wang**[2], **Daiting Shi**[1], **Dawei Yin**[†1]

[1]Baidu Inc.
[2]School of Information, Renmin University of China,Beijing, China
[3]Nankai University
[4]School of Artificial Intelligence, Jilin University, Changchun, China
sujinbo@ruc.edu.cn; *{gaolingzhe,yindawei02}@baidu.com*

Abstract

In web search scenarios, erroneous queries frequently degrade users' experience through irrelevant results, underscoring the pivotal role of Chinese Spelling Check (CSC) systems. Although large language models (LLMs) exhibit remarkable capabilities across many tasks, they face critical challenges in the CSC scenario: (1) poor generalization to rare entities in open-domain searches, and (2) failure to adapt to temporal entity variations due to static parameters, resulting in serious over-correction issues. To tackle this, we present RACQC, a **C**hinese **Q**uery **C**orrection system with **R**etrieval-**A**ugmented Generation(RAG) and multi-task learning. Specifically, our approach (1) integrates dynamic knowledge retrieval through entity-centric RAG to address rare entities and innovatively proposes an entity-title collaborative corpus, and (2) employs contrastive correction tasks to mitigate LLM over-correction tendencies. Furthermore, we propose MDCQC, a **M**ulti-**D**omain **C**hinese **Q**uery **C**orrection benchmark to test the model's entity correction capabilities. Extensive experiments on several datasets show that RACQC significantly outperforms existing baselines in CSC tasks. Specifically, RACQC achieves a maximum improvement of **+9.92%** on the search scenario benchmark and **+3.2%** on the general-domain dataset under the F_1 metric.[1]

1 Introduction

In real-world Chinese online search scenarios, users frequently make erroneous queries due to various factors such as input errors and knowledge gaps, resulting in poor relevance of search results. These errors manifest in multiple forms, including homophones, visually similar characters, and omissions or additions of characters. Searching with uncorrected queries often leads to substantial discrepancies between search results and users' needs.

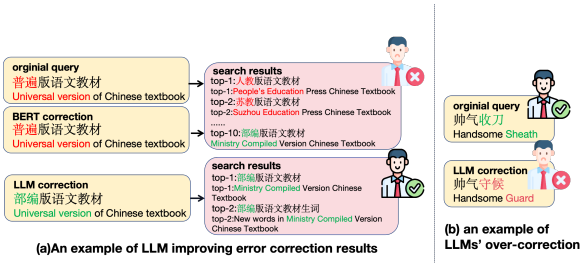

Figure 1: Examples of query correction,where the red characters represents the errors and green represents the correct result. The LLM is GPT-4.

Therefore, a Chinese Spelling Check (CSC) system aimed at detecting and correcting spelling errors is significant for search scenarios(Gao et al., 2010; Zhang et al., 2024). In the CSC task, the current mainstream methods based on the Sequence-to-Sequence(Seq2Seq) model conceptualize it as a machine translation problem, transforming erroneous sentences into correct ones(Raffel et al., 2020; Lewis, 2019). Furthermore, as shown in Figure 1, the development of large language models(LLMs) has further augmented the capabilities of Seq2Seq models in CSC(Achiam et al., 2023; Yang et al., 2024).

However, prior studies have demonstrated that LLMs do not perform well on CSC(Qu and Wu, 2023; Li et al., 2024). This limitation primarily stems from LLMs' propensity to over-correct for long-tail or temporal entities(Wang et al., 2024a). For instance, as illustrated in Figure 1, a user inputs "Handsome Sheath" and LLM erroneously corrects it into "Handsome Guard" because it tends to over-correction. The corrected query completely deviates from the user's original search demand, seriously disrupting the user's search experience.

[*]Work done during internship at Baidu Inc.
[†]Corresponding author
[1]Our MDCQC benchmark and training data can be accessed at https://github.com/Reinaqvq/RACQC

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 675–689
November 4-9, 2025

Mainstream research lacks solutions to this problem and corresponding CSC benchmarks.

To address this limitation, we propose RACQC, a **C**hinese **Q**uery **C**orrection system with **R**etrieval-**A**ugmented Generation. This approach aims to alleviate the issue of over-correction in CSC. Specifically, our approach is fundamentally designed to address two pivotal issues: (1) an intrinsic shortfall in the LLMs' CSC capabilities, and (2) a conspicuous absence of external knowledge within the model.

In terms of model capabilities, we innovatively propose five distinct error correction tasks to measure the model's error correction capability, including error detection, error correction scoring, error correction generation, error correction re-ranking, and error correction chain of thought(CoT)(Wei et al., 2022). Through a series of experiments, we observed that these tasks possess the potential to supplement and amplify each other. Inspired by this, RACQC has constructed a multi-task instruction fine-tuning dataset that encompasses these five types of tasks, aiming to enhance the performance of LLM in CSC.

In terms of utilizing external knowledge, RACQC innovatively introduces Retrieval-Augmented Generation(RAG) in error correction by exploiting webpage title data and entities extracted from the titles to establish an offline entity-title corpus. Upon encountering a query requiring external knowledge, the retriever searches for relevant information from the corpus to enhance the model's response, thereby addressing the over-correction issues generated by LLMs with out-of-distribution entities.

Furthermore, owing to the substantial discrepancies between the existing mainstream CSC datasets and search scenarios, the academic community is confronted with a dearth of authentic CSC datasets within the open-domain search scenario. To ameliorate this situation, we present MDCQC, a **M**ulti-**D**omain **C**hinese **Q**uery **C**orrection benchmark. Experiments on the search-domain datasets MDCQC and MCSC (Jiang et al., 2022), and the general-domain dataset LEMON (Wu et al., 2023), show that RACQC surpasses existing baselines, achieving state-of-the-art performance. Moreover, this system has been successfully deployed in real-world online scenarios. The contributions of this work can be summarized as follows:

- We propose RACQC, a CSC framework that introduces RAG using a novel entity-title corpus, specifically designed for entity-centric retrieval to support rare and evolving entity correction.

- We develop a multi-task training strategy that incorporates five complementary error correction tasks, including detection, scoring, generation, re-ranking, and chain-of-thought. Ablation studies demonstrate that these tasks mutually reinforce each other, leading to robust improvements in CSC performance.

- We release MDCQC, a challenging benchmark derived from real-world online search queries, containing over 4,000 examples across 10 domains with entity-level annotations.

- The experimental results substantiate that our method achieves SOTA performance across several datasets and has been successfully deployed in a online search engine with optimized real-time latency.

2 Related Work

2.1 Retrieval-Augmented Generation(RAG)

The RAG system aims to enhance the model's answer with external information (Lewis et al., 2020; Asai et al., 2023; Chen et al., 2024c). The retriever gathers relevant knowledge from an external base and feeds it into LLMs to improve the model's generation. Previous research has already proved the effectiveness of this method (Liu et al., 2024; Li et al., 2023; Chen et al., 2024b) and it has achieved outstanding performance in a variety of fields such as code generation (Islam et al., 2024; Wang et al., 2024d), open-domain QA (Wang et al., 2023, 2024e), table understanding (Chen et al., 2024a), and so on. However, according to our research, our work represents one of the earliest endeavors to integrate RAG into CSC and operationalize it in a production environment.

2.2 LLMs in Chinese Spelling Check(CSC)

CSC is an important task in Natural Language Processing. To improve CSC performance, previous research primarily used BERT-style models due to their contextual awareness and transfer learning capabilities(Devlin et al., 2019; Wu et al., 2023;

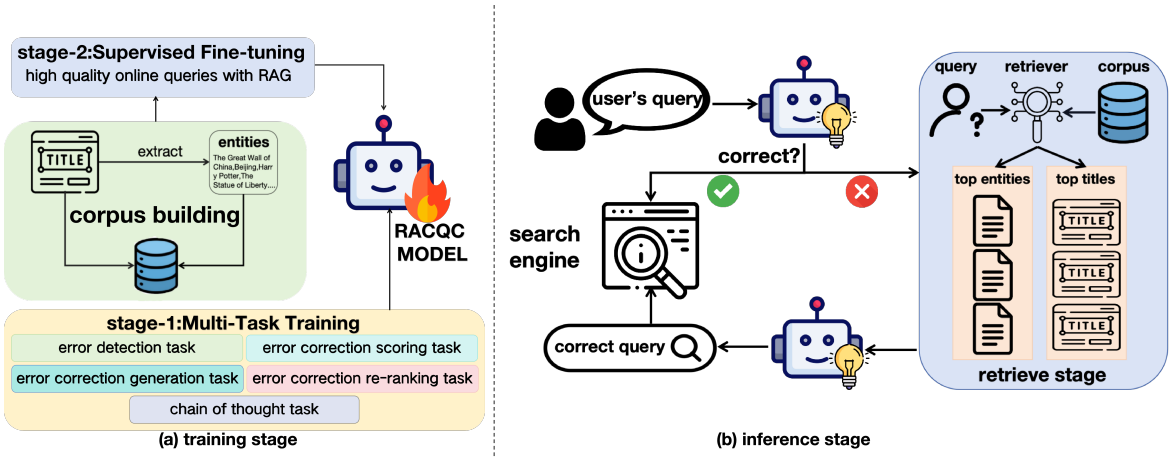

Figure 2: Overview of RACQC. RACQC introduces multi-task training and RAG into CSC tasks.

Cheng et al., 2020). To improve their error correction abilities, strategies such as data synthesis(Wang et al., 2024b; Hu et al., 2022), incorporating error detection modules(Zhang et al., 2020a), and specific character masking strategies(Liu et al., 2010) have been used.

With the advent of Large Language Models (LLMs) like ChatGPT(Achiam et al., 2023), their application in the CSC context has garnered interest. However, previous research indicates that LLMs tend to over-correct, resulting in an underperformance compared to the baseline BERT-style models(Qu and Wu, 2023; Wang et al., 2024c). In order to solve this problem, C-LLM(Li et al., 2024) and TIPA(Xu et al., 2024) proposed methods to make character-level alignment, while DeCoGLM(Li and Wang, 2024) incorporates a detection-correction structure based on the GLM. Additionally, $trigger^3$(Zhang et al., 2024) proposed a correction scheme based on the cooperation of large and small models. In contrast to them, in this work, we explored the correction training tasks needed by LLMs.

3 Methods

To overcome the limitations outlined above, we propose RACQC to augment the capabilities of LLMs in the CSC domain. As illustrated in Figure 2, our training process is divided into two main stages: the first stage employs multi-task training, and the second stage performs supervised fine-tuning (SFT) on high-quality samples. Additionally, during both the SFT and inference stages, we construct a high-quality entity-title corpus to enhance the response quality of LLMs.

3.1 Problem Formulation

The CSC task aims to correct all erroneous characters in Chinese sentences. Formally, let s denote a sentence containing erroneous characters, and let s^+ represent the set of correctly modified sentences. The model f generates a possibly correct modified sentence $s' = f(s)$. CSC task aims to ensure $s' \in s^+$. Additionally, s^- is defined as negative correction results generated by a random triggering method based on confusion sets (mined from our online scenarios) as shown in Algorithm 1.

3.2 Multi-task Training Data for CSC

At this stage, we introduced five types of CSC training tasks. Specifically, based on observations of the online cascaded error correction system, we propose the error detection, error correction scoring, and error correction generation tasks. Through further analysis of user error correction cases, we introduce more challenging error correction re-ranking and chain of thought tasks to unlock the full potential of LLMs in the CSC domain. These five tasks supplement and amplify each other, endowing the model with robust error correction abilities. The complete training instructions are provided in Appendix A, and a more detailed description of the roles of each task is provided in the Appendix B.

3.2.1 Error Detection Data

The goal of this task is to enhance the model's error identification capability. To this end, we formulate a binary classification task. More formally, the label $D_{ed}(s)$ in the error detection dataset is defined

as Equation 1.

$$D_{ed}(s) = \begin{cases} 1, & \text{if } s \text{ is incorrect} \\ 0, & \text{if } s \text{ is correct} \end{cases} \quad (1)$$

Through this task, we have augmented the model's adeptness in discerning errors and trained it to identify common Chinese error patterns.

3.2.2 Error Correction Scoring Data

The goal of this task is to enhance the model's ability to recognize high-quality error correction results. To achieve this, we also constructed a binary classification task. More formally, let c denote a possible candidate error correction randomly selected from s^+ and s^-. The label $D_{ecs}(s, c)$ in the error correction scoring dataset is defined as Equation 2.

$$D_{ecs}(s, c) = \begin{cases} 1, & \text{if } c \in s^+ \\ 0, & \text{if } c \in s^- \end{cases} \quad (2)$$

Through this task, we primarily enable the model to learn what kind of error correction results are necessary and of high quality.

3.2.3 Error Correction Re-Ranking Data

The goal of this task is to re-rank possible error correction candidates. To achieve this goal, we constructed an error correction ranking task to choose the best among multiple possible error correction results. For each piece of data, we first combine the error correction candidates from sets s^+ and s^-, verifying if the total count exceeds four. In case the candidates are insufficient, we employ Algorithm 1 to generate additional negative candidates until we obtain at least four candidates. These candidates are then numbered in ascending order. After this, we use the count of the candidates from the s^+ set among these four candidates as our labels.

Through this task, we have further enhanced the model's ability to recognize high-quality correction results. The model can better learn the differences between good and bad candidates by comparing multiple high-quality and low-quality correction candidates in the same sample.

3.2.4 Chain of Thought Data

Building on the foundation laid by (Wei et al., 2022), we explore the potential of Chain of Thought(CoT) reasoning to enhance error correction capabilities. Leveraging the advanced capabilities of GPT-4(Achiam et al., 2023), complemented by meticulous human review, we generate the detail thinking process of error correction for each piece of data and give the error correction results. With this, we aim to teach the model thinking process of the error correction task in complex scenarios. Prompts used when calling GPT-4, please refer to Appendix C. A more detailed discussion on the CoT task can be found in the Appendix D.

Algorithm 1 Get error correction negative samples

Input: Error query S , Corrected query S^+ , Confusion set ST , The position POS in S
Output: Negative sample S^-

S^-=S^+
P=Random(0,1)
if P<0.3 **then**
  POS_1 = Random(0,LENGTH(S^-))
  POS_2 = Random(0,LENGTH(S^-))
  SWAP($S^-_{POS_1}$,$S^-_{POS_2}$)
end if
POS = Random(0,LENGTH(S^-))
S^-_{POS} = $ST(S^-_{POS})$
return S^-

3.2.5 Error Correction Generation Data

The primary goal of this training task is to equip the model with the essential capabilities required for error correction generation. It further bolsters the model's aptitude for recognizing and understanding the error patterns learned from previous tasks, thereby refining its proficiency in discerning and amending errors. To achieve this goal, we input the erroneous sentence s and utilize all corrected sentences in s^+ set as the ground truth labels.

3.3 SFT and Inference Stage of RACQC

In both the Supervised Fine-tuning(SFT) and inference stages, we integrated RAG information to bolster the error correction capability of LLMs. To effectively utilize RAG within the CSC system, determining the appropriate content for our corpus is crucial. Considering the intent of user search behavior and synthesizing our experimental results, we conclude that title information plays a pivotal role. Regarding form, titles are similar to the user's search query but encapsulate more expansive information, thus providing a potential basis for error correction. This will be instrumental in helping LLMs to correct long-tail and temporal entities.

However, while the titles contain richer information, they often contain more noise, such as

TYPE	NE	NPE	NEE	LM
F&G	345	171	138	72
MED	722	326	229	98
NEW	133	49	38	18
LIF	1136	393	201	111
EDU	757	324	144	121
BOK	115	53	47	18
CAR	91	37	30	14
MUS	77	38	31	16
TEC	193	90	76	23
OTHER	484	161	97	37

Table 1: Overview of MDCQC dataset(**NE** means number of examples, **NPE** means number of positive examples, **NEE** means number of entity errors, **LM** means length mismatched).

redundant details and errors in the entities mentioned within the title. Such noise can significantly impair LLMs' generation. Therefore, we have enriched our corpus with entity information extracted from titles. A more detailed process of constructing the entity corpus is provided in the Appendix E. In both the SFT and inference stages, we retrieve four pieces of corpus data that are most similar to the query in terms of cosine similarity to augment LLMs' response.

3.4 MDCQC Benchmark

Our investigation reveals that general CSC datasets (Hu et al., 2022; Wu et al., 2023) in mainstream research often overlook entity-level error correction critical for open-domain search, focusing primarily on common entities with limited coverage of long-tail ones. Search-oriented CSC datasets (Jiang et al., 2022; Wang et al., 2024c) tend to be domain-specific, which may not fully capture the diversity of cross-domain and time-sensitive entities. Additionally, while mainstream academic datasets emphasize length-aligned error correction, length-mismatched errors are also prevalent in real search scenarios. Therefore, developing a dataset grounded in real open-domain search contexts is of significant value.

Based on this, we propose MDCQC,a **M**ulti-**D**omain **C**hinese **Q**uery **C**orrection dataset that spans ten diverse domains:film&game(F&G), medical(MED),news(NEW),life(LIF),education (EDU),books(BOK),cars(CAR),technology(TEC), music(MUS) and others. The data source is collected from representative queries that our online system struggles to handle in real online scenarios and incorporates our manually, meticulously annotated entity information.

Since the data comes from real online scenarios, it involves numerous long-tail and temporal queries, which bring challenges to the correction model at the entity level. The distribution of entities between different fields also has significant differences, posing challenges to the content of the external corpus it relies on. The overview of MDCQC is reported in Table 1. In the Appendix, we provide a detailed analysis of the specific discrepancies between MDCQC and mainstream CSC datasets.

4 Experiments

In this section, we present the details of SFT and the evaluation results of models on the three CSC benchmarks: LEMON, MCSC, and our multi-domain dataset MDCQC.

4.1 Experimental Settings

Datasets. Prior research in the general CSC domain predominantly leverages SIGHAN(Tseng et al., 2015; Wu et al., 2013; Yu et al., 2014) as a benchmark. However, SIGHAN has grown increasingly misaligned with contemporary Chinese input practices and diverges notably from real-world search query contexts. To address this, we employ the following three datasets in our study.

LEMON(Wu et al., 2023) is a large-scale multi-domain dataset with natural spelling errors, which spans seven domains, including game (GAM), encyclopedia (ENC), contract (COT), medical care (MEC), car (CAR), novel (NOV), and news (NEW).

MCSC(Jiang et al., 2022) is a specialist-annotated Medical Chinese Spelling Correction Dataset collected from a large-scale query log.

MDCQC is a multi-domain Chinese query correction benchmark, which comes from the online user query logs of a popular Chinese search engine. After careful manual annotation and filtering, high-quality Chinese error correction data are selected.

Metrics. Following previous studies(Zhang et al., 2024),we use the widely used metrics precision(P)/recall(R)/F-measure(F_1) to evaluate the performence of different models.

Baselines. We used the following models to compare our method. For traditional models, we selected the n-gram LM implemented based on KenLM(Heafield, 2011). For BERT-style models,

MODEL	MDCQC			MCSC		
	P	R	F_1	P	R	F_1
BERT	43.36	9.57	15.68	80.93	80.05	80.49
SM-BERT	38.68	11.89	18.19	81.21	80.51	80.86
GPT-4	22.12	26.24	24.00	25.11	31.12	27.79
ERNIE-4.0	43.54	37.90	40.52	51.01	50.05	50.52
N-GRAM LM	10.13	5.65	7.25	30.32	16.04	20.98
Qwen2-1.5B+SFT+RAG	64.84	40.15	49.59	75.64	75.05	75.34
RACQC + w/o RAG	53.13	40.32	45.85	68.05	69.99	69.00
RACQC	**75.03**	**49.31**	**59.51**	**81.39**	**81.04**	**81.21**

Table 2: Overall results of RACQC and baseline models on MDCQC and MCSC datasets. The best results are highlighted in bold and the second performence results are indicated by an underscore. W/o RAG means without RAG information from entity-title corpus.

MODEL	CAR	COT	ENC	GAM	MEC	NEW	NOV	AVG
BERT	46.8	52.6	45.7	23.4	42.7	46.6	32.3	41.4
SM-BERT	**49.9**	**54.8**	49.3	26.1	46.9	49.1	34.6	44.3
GPT-4	26.8	27.8	33.7	29.4	32.7	28.1	29.0	29.6
ERNIE-4.0	32.6	40.8	37.4	30.6	38.1	41.6	27.5	35.5
Qwen2-1.5B+SFT	42.5	48.2	48.3	30.8	50.3	41.6	32.9	42.0
TIPA+1.5B	45.2	52.9	46.1	28.4	50.0	47.4	29.6	42.8
RACQC+w/o RAG	46.0	52.4	**51.5**	**35.3**	**60.0**	51.8	**35.4**	**47.5**
RACQC	46.1	53.7	50.3	33.3	58.4	**53.0**	34.2	47.0

Table 3: Overall results of RACQC and baseline models on LEMON dataset, are presented as F_1 scores. The best results are highlighted in bold and the second performence results are indicated by an underscore. W/o RAG means without RAG information from entity-title corpus.

we chose the most basic BERT(Devlin et al., 2019) and its improved Soft-Masked BERT(Zhang et al., 2020b). For seq2seq models, we chose to compare the error correction effects with the most popular closed-source LLMs, which mainly include ERNIE-4.0 and GPT-4(Achiam et al., 2023). For the specific prompts used when calling GPT-4 and ERNIE-4.0, please refer to Appendix G. TIPA(Xu et al., 2024) is a recent work of LLM on CSC, its main idea is to align the LLM error correction at the character level. We compared with it on the LEMON dataset. For RACQC, due to the strong resource constraints in actual search scenarios, we use Qwen2-1.5B(Yang et al., 2024), which has a lower resource overhead, as our basemodel and we mainly divided it into two settings: with RAG and without RAG (w/o RAG). To test the effect of our multi-task training, we also experimented on SFT directly on Qwen2-1.5B. In the settings without RAG, the model will only take the query as input, while in the settings with RAG (w RAG), we retrieve the top-4 information from the entity-title corpus to enhance the model's answers. For prompts used when calling RACQC, please refer to Appendix H.

4.2 Implementation Details

We used real online search scenario logs for multi-task training, selecting samples over 90% correction probability as positive samples and random correct queries. Using Algorithm 1, we constructed negative samples s^-, generating 40 million samples with a 1:1 positive-to-negative ratio across five tasks. In the SFT and inference stage, we used the title data and entity information extracted from the WuDAO dataset(Yuan et al., 2021) as our entity-title corpus and retrieved the top four results with the highest cosine similarity for each piece of data, creating 400000 samples for the SFT stage. Smaller models like BERT and SM-BERT were directly trained on all data. For RACQC, we fine-tuned Qwen2-1.5B with Adam (initial learning rate: 1e-5, batch size: 64), using a cosine schedule for one epoch in multi-task training and three epochs in SFT. Adopt cross-entropy for all training loss functions. Our retriever always uses bge-

MODEL	P	R	F_1
RACQC	**75.0**	49.3	**59.5**
RACQC w/o ec gene	68.5	42.9	52.8
RACQC w/o ec scoring	73.4	47.2	57.5
RACQC w/o ed	74.9	47.6	58.2
RACQC w/o ec rerank	71.4	48.9	58.0
RACQC w/o CoT	72.3	**49.5**	58.8

Table 4: Abalation studies of RACQC on MDCQC.

DATASET	CORPUS	P	R	F_1
MDCQC	entity only	74.6	49.8	59.7
	title only	74.6	52.9	61.9
	entity-title	**78.2**	**54.5**	**64.2**
MCSC	entity only	81.0	80.7	80.9
	title only	82.4	82.3	82.4
	entity-title	**84.3**	**84.0**	**84.2**

Table 5: The effect of RACQC under different text corpus settings.All entities and titles are dumped from real online scenario.

large-zh-v1.5(Xiao et al., 2023) for both entity and title retrieval. All experiments are performed on 8 NVIDIA A100 80GB GPUs.

4.3 Main Results

The main results on the MCSC and MCDQC benchmarks are presented in Table 2, and the results on the LEMON benchmark are presented in Table 3. Key observations include: (1)Our RACQC method achieved SOTA performance on all three datasets, affirming the effectiveness of our multi-task training and introduction of RAG information to enhance LLMs' ability in the CSC task. (2) RACQC consistently surpasses direct SFT on LLMs across all benchmarks, highlighting the necessity of our multi-task training paradigm. This approach enables LLMs to acquire diverse error correction capabilities, which synergistically enhance each other. (3) The introduction of RAG information yields significant performance improvements on MDCQC and MCSC, datasets rooted in real search scenarios. This divergence underscores the prevalence of long-tail entity challenges in actual search scenarios and validates our approach in addressing them. On the LEMON dataset, RAG integration yielded minimal impact, because LEMON is a general error correction dataset, and most of the entities it involves are relatively common and can be directly covered by LLM. (4)LLMs like GPT-4 demonstrate stronger zero-shot performance on MCSC than on MDCQC, highlighting MDCQC's heightened complexity in real-world entity correction, where its entities exceed the scope of the model's pre-training knowledge. To further understand the model's capability in addressing over-correction issues, we provide a more fine-grained analysis in the Appendix I.

5 Analysis

5.1 Ablation Study

During the multi-task training phase, RACQC leverages five tasks: error detection data(ed), error correction scoring data(ec scoring), error correction generation data(ec gene), chain of thought data(CoT), and error correction re-ranking data(ec re-rank). We conduct ablation studies by removing each task individually to assess its contribution to CSC. The ablation results on the MDCQC dataset are presented in Table 4. We have the following observations.

Ablation of the error correction generation task. With the ablation of the ec gene task, we observed that both precision and recall have significantly decreased. It means a considerable drop in the model's performance. This proves that the ec gene task is the most important among five tasks because it directly gives model the ability to correct errors and further enhances its ability to detect errors.

Ablation of error correction scoring, re-rank and CoT task. With the ablation of these tasks, we observed a marginal decline in precision and recall. This suggests that the primary role of these tasks is to further enhance the model's error detection ability, error correction generation ability, and the ability to prioritize high-quality error correction results.

Ablation of error detection task. With the ablation task, we observed that the precision remained stable with a significant decline in recall. The overall F_1 score exhibits a marginal degradation. This suggests that the ed task mainly strengthens the model's understanding of errors, allowing the model to recall erroneous sentences accurately.

5.2 Corpus Build

Corpus quality critically influences RAG performance. This section examines how different corpus configurations affect RACQC, specifically comparing three settings: title-only, entity-only, and a combined entity-title corpus. To better reflect online deployment conditions, we replace WuDao data

DATASET	MODEL	P	R	F_1
MDCQC	directly SFT	58.3	33.7	42.7
	w/o RAG	61.4	40.3	48.7
	RACQC	**72.4**	**44.5**	**55.1**
MCSC	directly SFT	71.1	72.0	71.5
	w/o RAG	68.1	68.1	68.1
	RACQC	**77.1**	**77.0**	**77.1**

Table 6: The results of transferring the base model into LLAMA3-1B. "directly SFT" indicates fine-tuning the model only using SFT data, while "w/o RAG" denotes the exclusion of RAG information.

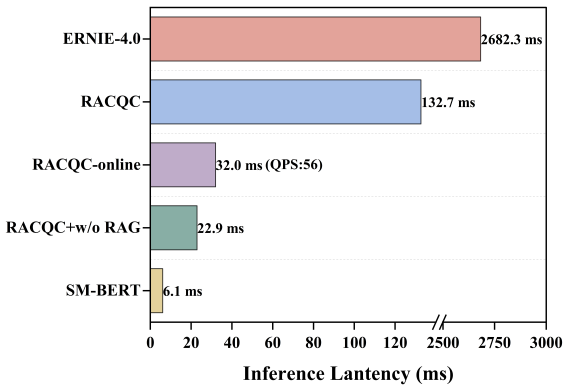

Figure 3: Average latency performance of different baselines and RACQC. "-online" indicates the optimized online deployment latency for actual search scenarios. All experiments were conducted on the MDCQC dataset.

Source	乙骨犹太
Target	乙骨忧太
GPT-4	易筋经太极
RACQC	乙骨忧太
RAG	entity:乙骨忧太,title:战神乙骨犹太!
Source	仿徨之刃
Target	彷徨之刃
GPT-4	放浪之刃
RACQC	彷徨之刃
RAG	title: 彷徨之刃电影-在线播放

Table 7: Case studies selected from MDCQC. Red indicates errors, green indicates correct corrections.

with real-world online entities and titles.

Experimental results in Table 5 reveal: (1) The entity-only setting underperforms on MCSC and MDCQC, primarily because the corpus limited to entity names fails to provide the model with sufficient context to understand entity specifics or facilitate direct error correction based on retrieved entity information. (2)The title-only setting suffers from real-world noise, which undermines the model's effectiveness. To address this, we ultimately combine entity and title information to construct a more robust entity-title corpus.

5.3 Transferability of RACQC

To demonstrate the transferability of our RACQC method, we migrated the base model of RACQC from Qwen2-1.5B to LLAMA3-1B(Dubey et al., 2024). The results are presented in Table 6. From the results, it can be observed that our five training tasks consistently deliver robust results on LLAMA3-1B. This indicates that the five training tasks we propose exhibit transferability. Furthermore, observations from the ablation study on RAG information reveal that our attempt to incorporate the RAG method into the CSC task is effective. In summary, our proposed method can seamlessly integrate into existing CSC approaches.

5.4 Efficiency Analysis

To evaluate RAG's impact on efficiency, we compare RACQC's latency with various baselines, as shown in Figure 3. Although RACQC has slightly higher latency than online BERT-like models due to the RAG process and increased model parameters, it significantly improves performance metrics.

To address latency in real-world large-scale search scenarios, we implemented several optimization strategies, primarily including INT-8 quantization and techniques based on caching. Through these optimization strategies, we reduced the average inference latency by **4.14x** in real-world online deployment compared to offline testing, while maintaining near-lossless performance in online evaluations. The optimized average latency meets the low-latency requirements of real online deployment, enabling application in real-world scenarios. Details of the online optimization mechanisms are provided in Appendix J.

5.5 Case Studies

We analyzed two representative samples from the MDCQC dataset, as shown in Table 7. In the first case, 乙骨忧太 (Okkotsu Yūta), a character from the anime Jujutsu Kaisen (2021), is mistakenly entered as 乙骨犹太. Due to a lack of knowledge after 2018, GPT-4 has corrected it to "Yijinjing Tai Chi". This represents a significant discrepancy from the actual needs of the user. If enhancement is only based on the title information, errors may occur because "忧" is wrongly spelled as "犹" in

Metric	QCR(↓)	BR(↑)	USR(↑)
Lift Rate	-1%	+1.5%	+1.15%

Table 8: Online Experimental Results. Lift Rate denotes the relative change compared to the online baseline; ↑ indicates higher is better, ↓ indicates lower is better.

the title. However, the entity information is correct, enabling RACQC to correct the correction. In the second case, we can make similar observations.

5.6 Online Results

To further validate the effectiveness of our proposed RACQC framework, we deployed the entire system online and conducted a performance comparison with the original online system. The online experiments primarily employed the following metrics:

- **Query Change Rate (QCR)**: The average number of query modifications per page view. A lower QCR indicates better system performance.
- **Bounce Rate (BR)**: The ratio of clicks with redirections to total page views. A higher BR suggests better performance.
- **User Satisfaction Rate (USR)**: The proportion of satisfactory page views to total page views. A higher USR reflects better user experience. (A "satisfactory page view" is defined as one in which the user clicks and remains on the page for a duration exceeding a predefined threshold.)

The experimental results are summarized in Table 8. Our online evaluation demonstrates that the proposed RACQC framework significantly enhances the error correction capability of the online system compared to the original baseline, achieving substantial improvements across multiple key user experience metrics. These results further validate the effectiveness of the RACQC system.

6 Conclusion

This paper highlights the over-correction issues LLMs face in real-world CSC scenarios due to their limited error correction capacity and knowledge gap. To address this issue, we propose a novel framework, RACQC. It encompasses five different types of training tasks to enhance the model's error correction capability. Concurrently, we construct an entity-title corpus to employ the RAG methodology to resolve the problem of the model lacking external knowledge. Experimental results indicate that RACQC achieves SOTA performance on both search and general datasets.

7 Limitations

Our work is designed for error correction in the chinese domain, so it may struggle with english error correction. Meanwhile, for the retrieval strategies of titles and entities, this work adopts a text vectorization-based approach. Employing more advanced methods, such as knowledge graph-based retrieval strategies, may further enhance performance. Furthermore, our multitask training requires additional training overhead, which may need to be improved in the future.

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*.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Si-An Chen, Lesly Miculicich, Julian Martin Eisenschlos, Zifeng Wang, Zilong Wang, Yanfei Chen, Yasuhisa Fujii, Hsuan-Tien Lin, Chen-Yu Lee, and Tomas Pfister. 2024a. Tablerag: Million-token table understanding with language models. *ArXiv*, abs/2410.04739.

Weijie Chen, Ting Bai, Jinbo Su, Jian Luan, Wei Liu, and Chuan Shi. 2024b. Kg-retriever: Efficient knowledge indexing for retrieval-augmented large language models.

Xiaoyang Chen, Ben He, Hongyu Lin, Xianpei Han, Tianshu Wang, Boxi Cao, Le Sun, and Yingfei Sun. 2024c. Spiral of silence: How is large language model killing information retrieval? - a case study on open domain question answering. In *Annual Meeting of the Association for Computational Linguistics*.

Xingyi Cheng, Weidi Xu, Kunlong Chen, Shaohua Jiang, Feng Wang, Taifeng Wang, Wei Chu, and Yuan Qi. 2020. Spellgcn: Incorporating phonological and visual similarities into language models for chinese spelling check. In *Annual Meeting of the Association for Computational Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep

bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony S. Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Cantón Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab A. AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriele Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guanglong Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Laurens Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Ju-Qing Jia, Kalyan Vasuden Alwala, K. Upasani, Kate Plawiak, Keqian Li, Ken-591 neth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Babu Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melissa Hall Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri S. Chatterji, Olivier Duchenne, Onur cCelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasić, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Chandra Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whit ney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yiqian Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zhengxu Yan, Zhengxing Chen, Zoe Papakipos, Aaditya K. Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adi Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Ben Leonhardi, Po-Yao (Bernie) Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Shang-Wen Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzm'an, Frank J. Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory G. Sizov, Guangyi Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Han Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kaixing(Kai) Wu, U KamHou, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, A Lavender, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael

L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollár, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sung-Bae Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Andrei Poenaru, Vlad T. Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xia Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models. *ArXiv*, abs/2407.21783.

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*.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

Yong Hu, Fandong Meng, and Jie Zhou. 2022. Cscd-ns: a chinese spelling check dataset for native speakers. *arXiv preprint arXiv:2211.08788*.

Md. Ashraful Islam, Mohammed Eunus Ali, and Md. Rizwan Parvez. 2024. Mapcoder: Multi-agent code generation for competitive problem solving. In *Annual Meeting of the Association for Computational Linguistics*.

Wangjie Jiang, Zhihao Ye, Zijing Ou, Ruihui Zhao, Jianguang Zheng, Yi Liu, Bang Liu, Siheng Li, Yujiu Yang, and Yefeng Zheng. 2022. Mcscset: A specialist-annotated dataset for medical-domain chinese spelling correction. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pages 4084–4088.

Mike Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

Kunting Li, Yong Hu, Liang He, Fandong Meng, and Jie Zhou. 2024. C-llm: Learn to check chinese spelling errors character by character. In *Conference on Empirical Methods in Natural Language Processing*.

Wei Li and Houfeng Wang. 2024. Detection-correction structure via general language model for grammatical error correction. *ArXiv*, abs/2405.17804.

Xingxuan Li, Ruochen Zhao, Yew Ken Chia, Bosheng Ding, Shafiq R. Joty, Soujanya Poria, and Lidong Bing. 2023. Chain-of-knowledge: Grounding large language models via dynamic knowledge adapting over heterogeneous sources. In *International Conference on Learning Representations*.

Chao-Lin Liu, Min-Hua Lai, Yi-Hsuan Chuang, and Chia-Ying Lee. 2010. Visually and phonologically similar characters in incorrect simplified chinese words. In *Coling 2010: Posters*, pages 739–747.

Yanming Liu, Xinyue Peng, Xuhong Zhang, Weihao Liu, Jianwei Yin, Jiannan Cao, and Tianyu Du. 2024. Ra-isf: Learning to answer and understand from retrieval augmentation via iterative self-feedback. In *Annual Meeting of the Association for Computational Linguistics*.

Fanyi Qu and Yunfang Wu. 2023. Evaluating the capability of large-scale language models on chinese grammatical error correction task. *arXiv preprint arXiv:2307.03972*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

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.

Xintao Wang, Qian Yang, Yongting Qiu, Jiaqing Liang, Qi He, Zhouhong Gu, Yanghua Xiao, and W. Wang. 2023. Knowledgpt: Enhancing large language models with retrieval and storage access on knowledge bases. *ArXiv*, abs/2308.11761.

Yixuan Wang, Baoxin Wang, Yijun Liu, Dayong Wu, and Wanxiang Che. 2024a. 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.

Yixuan Wang, Baoxin Wang, Yijun Liu, Qingfu Zhu, Dayong Wu, and Wanxiang Che. 2024b. Improving grammatical error correction via contextual data augmentation. In *Annual Meeting of the Association for Computational Linguistics*.

Yue Wang, Zilong Zheng, Zecheng Tang, Juntao Li, Zhihui Liu, Kunlong Chen, Jinxiong Chang, Qishen Zhang, Zhongyi Liu, and Min Zhang. 2024c. Towards better chinese spelling check for search engines: A new dataset and strong baseline. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 769–778.

Zihao Wang, Anji Liu, Haowei Lin, Jiaqi Li, Xiaojian Ma, and Yitao Liang. 2024d. Rat: Retrieval augmented thoughts elicit context-aware reasoning in long-horizon generation. *ArXiv*, abs/2403.05313.

Zora Z. Wang, Akari Asai, Xinyan V. Yu, Frank F. Xu, Yiqing Xie, Graham Neubig, and Daniel Fried. 2024e. Coderag-bench: Can retrieval augment code generation?

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

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

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.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Zhuo Xu, Zhiqiang Zhao, Zihan Zhang, Yuchi Liu, Quanwei Shen, Fei Liu, and Yu Kuang. 2024. Enhancing character-level understanding in llms through token internal structure learning. *ArXiv*, abs/2411.17679.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

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.

Sha Yuan, Hanyu Zhao, Zhengxiao Du, Ming Ding, and Jie Tang. 2021. Wudaocorpora: A super large-scale chinese corpora for pre-training language models. *AI Open*.

Kepu Zhang, ZhongXiang Sun, Xiao Zhang, Xiaoxue Zang, Kai Zheng, Yang Song, and Jun Xu. 2024. Trigger3: Refining query correction via adaptive model selector.

Shaohua Zhang, Haoran Huang, Jicong Liu, and Hang Li. 2020a. Spelling error correction with soft-masked bert. In *Annual Meeting of the Association for Computational Linguistics*.

Shaohua Zhang, Haoran Huang, Jicong Liu, and Hang Li. 2020b. Spelling error correction with soft-masked BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 882–890, Online. Association for Computational Linguistics.

A Training instructions

Training instructions

ed task:
You are an expert in query text correction for search engines. Your task is to: determine whether the query has grammatical or factual errors.
ec ranking:
You are an expert in search engine query text correction. Your task is to:
1)determine whether there are grammatical or factual errors in the query
2)determine whether the correction result is correct based on the given search query correction result
ec gene:
You are an expert in search engine query text correction.Your task is to:
1) determine whether there are grammatical or factual errors in the query
2) If there are errors,analyze the user's search intent, and provide possible correction results.
ec rerank:
You are a search engine text correction specialist.
Your task is to:
1)rank given correction options for a query
2)identify the most suitable one with minimal changes and no errors
3)output its number
CoT:
You are a search engine text correction specialist. Your task is to:
Correct the original sentence with minimal changes and no errors.
You're also required to explain your thought process in making the correction.

B Discussion of Multi-task Building

In our early exploration of fine-tuning LLMs directly on ec gene task, we observed a critical limitation: the model tended to memorize high-frequency entities (e.g., erroneously correcting "PAVA algorithm" to "JAVA algorithm",) rather than learning genuine error correction patterns.

To address this issue, we drew inspiration from our online cascading error correction system consisting of two key stages: error detection and correction scoring. These stages effectively capture the model's error correction capability, which motivated our design of the ec scoring task and error detection task.

Building upon this foundation, we further argue that introducing more sophisticated correction tasks can push the boundaries of model performance. This rationale led to the development of our more challenging ec rerank and CoT tasks.

Ablation experiments demonstrated that all tasks positively influenced the experimental results. This exploration, in fact, highlights the upper limits of LLMs' capabilities in the field of error correction. The joint fine-tuning across these five distinct task types serves to enhance the model's genuine error correction ability, rather than merely promoting memorization of training data.

C Prompt for generating CoT task

prompt for generating CoT task

你是一个搜索引擎query文本纠错专家,你的任务是:
1)判断query是否有语法或者事实性错误;
2)给出纠错后的query，请你补充思考过程
现在，原始的query是：{original_query}
纠错后的query是:{correct_query}
请按照如下格式输出:
{"思考过程是": "", "纠正错误后的query应该是": ""}
English translation:
You are a search engine query text correction expert, your tasks are:
Determine whether the query has grammatical or factual errors;
After providing the corrected query, please supplement your thought process.
Now, the original query is:{orginal_query}
The corrected query is:{correct_query}
Please output in the following format:
{"Thought process": "", "Corrected query should be": ""}

D More detailed discussion on the CoT task.

Previous studies have not attempted to introduce CoT tasks in the CSC task, often because the CSC

	LEMON	MCSC	MDCQC
Ratio	6%	19%	**37%**

Table 9: Detailed Analysis of Long-Tail and Temporal Entity Distributions. Here, Ratio denotes the percentage of these two entity types within the sample.

task is generally considered to be relatively local and surface-level, and does not require complex reasoning. In fact, regarding the CoT data, we aim to guide the model in learning complex corrections for real search scenarios. A key challenge involves resolving logical inconsistencies in user queries - for instance, correcting "春花厌电视剧"(Chun Hua Yan TV series) to "春花焰电视剧"(Kill Me Love Me TV series). The model needs to first understand that "春花厌" is a novel, while "春花焰" is a TV series. Then, based on the user's query, it should reason that the user's intent is to search for a TV series, and use this as the basis for correction. By introducing CoT, we aim to teach the model to correct high-difficulty erroneous queries like this, thereby enhancing the model's performance ceiling.

E Entity Corpus Construction Process

To construct the entity corpus, we first employ a pre-trained Named Entity Recognition (NER) model to extract entities from the titles. To ensure the accuracy of the extracted entities, we perform a secondary verification process using the title information to validate the entities. Specifically, for each extracted $entity_i$, we retrieve the top 10 titles with the highest similarity to it. Subsequently, we calculate the frequency k at which $entity_i$ appears within these ten titles. Finally, we retain all entities with $k \geq 5$, which are verified by titles, thus obtaining a high-quality entity corpus.

F Further analysis of the MDCQC benchmark

To further demonstrate the importance of the MDCQC dataset, we conducted additional statistical analyses on MDCQC, LEMON, and MCSC. From a random sample of 100 entries, we defined the following two categories:

- **Temporal entities**: Entities that emerged after 2023.
- **Long-tail entities**: Entities with fewer than or equal to 10 monthly searches.

We computed the proportions of both entity types within the dataset and report the results in Table 9. The results indicate persistent discrepancies in long-tail and temporal entity distributions between conventional benchmark datasets (MCSC, LEMON) and search-oriented MDCQC dataset.

G Prompts for calling GPT and Ernie-4.0

Prompts for calling GPT and Ernie-4.0

你是一个搜索引擎query文本纠错专家,你的任务是:
1)判断query是否有语法或者事实性错误;
2)如果有错误,给出纠错后的query,并且要求改动最小。
如果有错请在query是否有错字段输出是，否则输出否。
如果query没有错误，把纠正错误后的query字段设为空；否则给出你的纠错结果。
请按照如下格式输出:
{"query是否有错": "", "纠正错误后的query应该是": ""}
现在，query是:{query}

English translation:

You are a search engine query text correction expert, your tasks are:
1.Determine whether the query has grammatical or factual errors;
2.If there are errors, provide the corrected query with minimal changes.
If there is an error, output "yes" in the "Does the query have errors?" field, otherwise output "no".
If the query is correct, the "Corrected query should be" field should be left blank; otherwise, provide your correction
Please output in the following format:
{"Does the query have errors?": "", "Corrected query should be": ""}
Now, the query is: {query}

H Prompts for calling RACQC

Prompts for calling RACQC

你是一个搜索引擎query文本纠错专家,你的任务是:
1)判断query是否有语法或者事实性错误;
2)如果有错误,给出纠错后的query,并且要求改动最小。
当前搜索引擎排名top的展现结果为[{titles}]
请按照如下格式输出:
{"query是否有错": "", "纠正错误后的query应该是": ""}
现在，query是:{query}

English translation:

You are a search engine query text correction expert, your tasks are:
Determine whether the query has grammatical or factual errors;
If there are errors, provide the corrected query with minimal changes.
The current top-ranked display results of the search engine are [titles]
Please output in the following format:
{"Does the query have errors?": "", "Corrected query should be": ""}
Now, the query is: {query}

I Further analysis on mitigating over-correction

To further investigate how our method addresses over-correction, we conducted a more detailed analysis of the results on the MDCQC test set. First, we identified two key manifestations of over-correction:

- **Correct-to-Incorrect**: The model modifies originally accurate sentences into incorrect ones.
- **Incorrect-to-Worse**: The model exacerbates already erroneous sentences.

The experimental results are presented in the table 10. The results indicate that RACQC reduces over-correction instances by 20.3% compared to the SFT baseline, demonstrating its effectiveness in mitigating this type of error.

Model	Over-correction	Lift Rate
qwen2-1.5B+SFT	359	-
RACQC	286	20.3%

Table 10: Detailed Analysis of Over-Correction Phenomena.

J Online optimization mechanisms

In our production deployment, we apply INT-8 quantization to the RACQC model. Online experimental results demonstrate that this quantization approach maintains acceptable accuracy with minimal degradation. What's more, we implement a caching mechanism for entity-title corpus retrieval results. This model's processing time can meet the requirements of our online system, as it can be executed in parallel with other query analysis operators to enable real-time service.

To further optimize latency, we have implemented an additional caching and large-small model collaboration strategy: upon the first occurrence of a query, it is processed through the RACQC and online small model cascade system via a consumption queue, and the computed results are stored in a monthly cache. If the same query is submitted again and a corresponding result exists in the cache, it is directly retrieved and served. This approach enables highly efficient online serving while preserving near-perfect user experience, as over 99.9% of queries occur at least twice per month. For the retrieval process, we also use a similar caching strategy, where queries can directly access the cache to obtain the necessary entity-title corpus information.