RACQC: Advanced Retrieval-Augmented Generation for Chinese Query Correction

Anonymous ACL submission

Abstract

In web search scenarios, erroneous queries frequently degrade user experience by leading to irrelevant results. This underscores the critical role of Chinese Spelling Check (CSC) systems in maintaining search quality. Conventional approaches typically employ domain-specific models trained on limited corpora. While effective for frequent errors, these models exhibit two key limitations: (1) poor generalization to rare entities in open-domain searches,and (2) inability to adapt to temporal entity variations due to static training paradigms. With the advent of Large Language Models(LLMs), a potential solution has been provided for these problems. However, LLMs have serious over-correction issues and struggle to handle long-tail entities. 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 handle rare entities 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.[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.

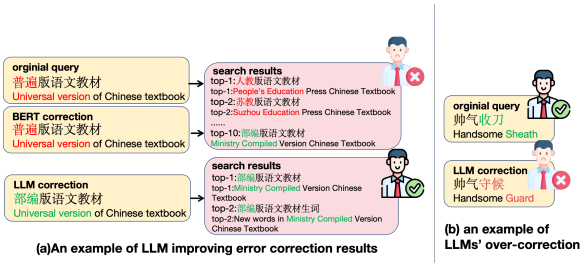

Figure 1: Examples of query correction,where the red characters represents the errors and green represents the correct result. The LLM is GPT-4.

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), which is due to the lack of such entity information in the pretraining data and the hallucination of LLMs. 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, thereby seriously disrupting the user's

[1]Our MDCQC benchmark and training data can be accessed at https://anonymous.4open.science/r/RACQC_v1

search experience. Currently, 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, we have discerned that the genesis of this issue is twofold: (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 believe that the error correction capability can be primarily measured through five distinct tasks, including error detection, error correction scoring, error correction generation, error correction re-ranking and error correction chain of thought(CoT)(Wei et al., 2022). We hypothesize 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 will search for relevant information from the corpus. The retrieved information will be used to enhance the model's response, thereby addressing the over-correction issues generated by LLMs with out-of-distribution entities. Experiments on the medical-domain dataset MCSC(Jiang et al., 2022) and multi-domain dataset LEMON(Wu et al., 2023) show that the performance of RACQC transcends existing baselines, achieving state-of-the-art (SOTA) performance.

Furthermore, owing to the substantial discrepancies between the existing mainstream CSC datasets and search scenarios, they do not present enough challenges to model's ability in correcting entity errors. The LEMON dataset is deficient in entity correction samples, while the MCSC dataset lacks errors like word addition or omission. To ameliorate this situation, we present MDCQC, a **M**ulti-**D**omain **C**hinese **Q**uery **C**orrection benchmark, which includes more than 4000 examples across 10 different domains from actual online user queries that online system struggles to handle, with human-annotated entity information. Notably, we still achieved the SOTA performance in the MDCQC dataset. The contributions of this work can be summarized as follows:

- We propose the RACQC framework, innovatively building an entity-title corpus to introduce RAG into the CSC task, which enhances the capability of LLMs for entity error correction.
- We propose multi-task training for error correction, introducing five error correction training tasks in instruction fine-tuning. They mutually enhance each other, improving the model's performance on the CSC task.
- Based on online search scenarios, we release MDCQC, a more challenging multi-domain Chinese query correction benchmark.

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 fed 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 variety of fields such as code generation(Islam et al., 2024; Wang et al., 2024c), open-domain QA(Wang et al., 2023, 2024d), table understanding(Chen et al., 2024a), and so on. However, according to our research, almost no work has applied RAG to CSC tasks.

2.2 LLMs in Chinese Spelling Check(CSC)

CSC is an important task in Natural Language Processing. Previous research primarily used BERT-style models due to their contextual awareness and transfer learning capabilities(Devlin et al., 2019; Wu et al., 2023; Cheng et al., 2020).To improve their error correction abilities, strategies like data synthesis(Wang et al., 2024b), incorporating error detection modules(Zhang et al., 2020a), and specific character masking strategies(Liu et al., 2010) have been used. However, due to the lack of knowledge and the inherent parameter limitations of BERT-style models, they struggle to handle long-tail entity queries.

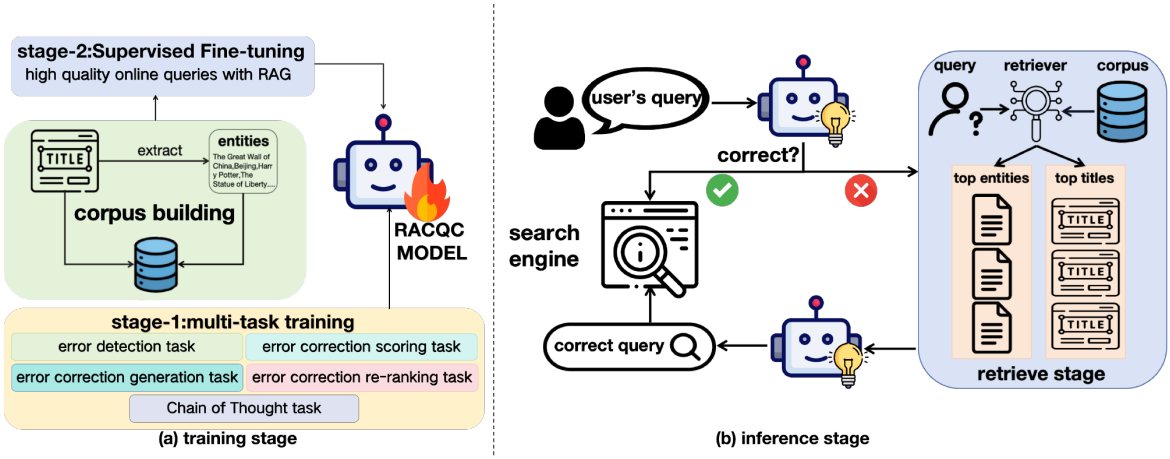

Figure 2: Overview of RACQC.RACQC introduce multi-task training and RAG into CSC tasks.

With the advent of Large Language Models (LLMs) like ChatGPT(Achiam et al., 2023), some work has begun to explore their application to the CSC context. However, previous research indicates that LLMs tend to over-correct, resulting in an underperformence to the baseline BERT-style models(Qu and Wu, 2023).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.And $trigger^3$(Zhang et al., 2024) proposed a correction scheme based on the cooperation of large and small models. Nonetheless, these works disregard training tasks that need to be introduced to enhance the model's ability. Therefore, 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 will generate 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. Each type of error correction task corresponds to a distinct error correction capability, and these five kinds of abilities supplement and amplify each other, endowing the model with robust error correction abilities. Due to space constraints, the complete training instructions are provided in Appendix A. Additionally, we provide examples of training data in our anonymous GitHub repository.

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 Eq 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

Algorithm 1 Get error correction negative samples

Input: Orginial error query S , Corrected query S^+ , Confusion set ST
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^-

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 Eq 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 gave 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 B

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, we find 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 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. 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

LEMON(Wu et al., 2023) is a popular CSC benchmark at present. However, it does not pose sufficient challenges for the entity correction scenario. Meanwhile, MCSC benchmark(Jiang et al., 2022), a correction set in the medical field, lacks types of common errors in practical search scenarios, such as adding and omitting characters. This makes

type	NE	NPE	NEE	A&O
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:number of examples,NPE:number of positive examples,NEE:number of entity errors,A&O:adding and omitting numbers)

them far from real search scenarios, so a Chinese error correction dataset based on actual open-domain search scenarios is necessary.

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. Compared with constructing CSC data based on error patterns, this collection method can be closer to the input habits of real human users.

At the same time, since the data comes from real online scenarios, it will involve a large number of long-tail and temporal queries, which brings challenges to the correction model at the entity level.This requires much external information. The distribution of entities between different fields also has significant differences, posing challenges to the content of the external corpus that it relies on. The overview of MDCQC is reported in Table 1.

4 Experiments

4.1 Experimental Settings

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

Datasets. Previous studies in the general CSC field often use SIGHAN(Tseng et al., 2015; Wu et al., 2013; Yu et al., 2014) as a benchmark. However, over time, the SIGHAN benchmark has become increasingly challenging to simulate current Chinese input habits. Moreover, there is a significant difference between the existing general field CSC dataset and the query under the actual search scenario. We need search domain datasets for experimentation. In summary, we used the following three datasets for our experiments.

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). Compared to SIGHAN, it shows better text quality and annotation accuracy.

MCSC(Jiang et al., 2022) is a Medical Chinese Spelling Correction Dataset, a large-scale and specialist-annotated dataset for Chinese spelling correction in the medical domain. It is collected from a large-scale query log dataset from a real-world medical application.

MDCQC is a multi-domain chinese query correction benchmark, it comes from the actual online user query logs of a popular Chinese search engine. After careful manual annotation and filtering, high-quality Chinese error correction data is 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 for comparison with our method. For traditional models, we selected the n-gram LM implemented based on KenLM(Heafield, 2011). For BERT-style models, 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). Due to space constraints, for the specific prompts used when calling GPT-4 and ERNIE-4.0, please refer to Appendix C.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, to simulate the real environment, we use qwen2-1.5B(Yang et al., 2024), which has

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 result 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 result 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.

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 D.

4.2 Implementation Details

Our code is based on LLaMA-Factory(Zheng et al., 2024). We used real online search scenario logs for multi-task training, selecting samples over 90% correction probability as positive samples and random correct queries. Furthermore, we utilized the method proposed in Algorithm 1 to construct all s^-.Finally, we constructed 40 million samples, maintaining a 1:1 ratio of positive to negative samples for five training 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 title-entity 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, in multi-task training stage, we fine-tune the entire qwen2-1.5B with Adam optimizer, setting the initial learning rate to 1e-5, the batch size to 64, and apply a cosine learning schedule for one epoch. In the SFT stage, we apply a cosine learning schedule for three epochs. Adopt cross-entropy for all training loss functions. Our retriever always uses bge-large-zh-v1.5(Xiao et al., 2023). All experiments are performed on 8xNVIDIA A100 80GB GPUs.

4.3 Main Results

The main results on the MCSC and MCDQC test sets are presented in Table 2, and the results on the LEMON test set are presented in Table 3. From these results, we can draw the following conclusions:(1)Our RACQC method achieved SOTA per-

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 datasets.The boldface indicates the best performance.

formance on all three datasets, affirming the effectiveness of our multi-task training and introduction of RAG information to enhance LLMs' ability in CSC task. (2)RACQC consistently outperforms direct SFT on LLMs across all test sets. This underscores that the introduction of our multi-task training is necessary. Through this training paradigm, LLM not only learned various error correction capabilities but also demonstrated that these capabilities synergistically reinforce each other. (3)In the two datasets, MDCQC and MCSC, which are based on actual search scenarios, the introduction of RAG information yields significant performance improvements. This performance disparity indicates that in actual search scenarios, the problem of long-tail entities does exist and highlights the effectiveness of our approach in overcoming this problem. On the LEMON dataset, the introduction of RAG information did not significantly 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 such as GPT-4 exhibit superior zero-shot performance on the MCSC dataset compared to on the MDCQC dataset. This performance disparity suggests that the MDCQC dataset is more challenging for models in real-world entity correction tasks, and the entities MDCQC involves are more difficult for the pre-training knowledge of the model to cover.

5 Analysis

5.1 Ablation Study

During the multi-task training phase, our RACQC training tasks mainly consist of the following five parts: 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 perform a series of ablation experiments to verify the individual contribution of these five tasks on the CSC task. Specifically, we remove one task from the five tasks each time to evaluate the impact on the model's performance.The ablation results on MDCQC dataset are presented in Table 4. Based on the experimental results, we have the following observations:

Ablation of the ec gene 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 the five tasks because it directly gives the model the ability to correct errors and further enhances its ability to detect errors.

Ablation of ec 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 ed 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

As highlighted in the introduction and methods, the quality of the text corpus plays a crucial role in the effectiveness of the RAG system. In this section, we mainly discuss the impact of different text corpus settings on the effectiveness of RACQC. We mainly considered three settings: the first utilizes only web page titles(title only), the second employs only entities extracted from titles(entity only), and the third includes both entity and title information(entity-title). Furthermore, in order to align with the actual online deployment scenario, the title and entity data in this session no longer come from the WuDao dataset but from our real online entities and titles.

Based on the experimental results in Table 5, we have the following analysis: (1)In the entity-only scenario, RACQC's performance has declined on the MCSC and MDCQC datasets. The primary reason for this is that a corpus containing only entity names does not enable the model to comprehend the specific information about the entities, nor does it allow for direct error correction based on the re-

dataset	data sourse	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.The boldface indicates the best performance.

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.The boldface indicates the best performance.

trieved entity name information.(2)In the title-only scenario, model performance tends to decline due to noise in the real-world title data. Consequently, the noise within the titles themselves adversely impacts the model's effectiveness when relying solely on title information. Therefore, we ultimately integrate entity and title information to construct our 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 the Table 6. From the results, it can be observed that our five training tasks consistently deliver robust 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.

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.The red text means that there is an error in this word, and the green text means that the error has been corrected correctly.

5.4 Case studies

We have selected two representative samples from the MDCQC dataset for analysis, displayed in Table 7.In the first case,乙骨忧太(Okkotsu Yūta) is a character from anime Jujutsu Kaisen, which has been serialized since 2018. The user erroneously entered his name 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 the title. However, the entity information is correct, enabling RACQC to correct the correction. In the second case, we can make similar observations.

6 Conclusion

This paper points out that LLMs exhibit significant over-correction issues in real-world CSC scenarios. We find that the root cause of the problem lies in the insufficient error correction capability of the LLMs and the lack of relevant knowledge, making it difficult for the model to deal with complex online scenarios. 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 high-quality entity-title corpus to employ the RAG methodology to resolve the problem of the model lacking external knowledge. Experimental results indicate that RACQC achieves state-of-the-art performance on both search and general datasets, including on MDCQC, a multi-domain Chinese query correction dataset we proposed.

7 Limitations

Our work is designed for error correction in the chinese domain, so it may struggle with english error correction. Also, introducing RAG information adds extra query time for each correction, posing a challenge for practical online deployment. 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.

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*.

Zihao Wang, Anji Liu, Haowei Lin, Jiaqi Li, Xiaojian Ma, and Yitao Liang. 2024c. 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. 2024d. 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. Trigger^3 : $Refiningquerycorrectionviaadaptivemodelselector$.

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.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. 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 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": ""}

C 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}

D 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}