AI Open 4 (2023) 183–192

Contents lists available at ScienceDirect

AI Open

journal homepage: www.keaipublishing.com/en/journals/ai-open

Full Length Article

Is Chinese Spelling Check ready? Understanding the correction behavior in real-world scenarios

Liner Yang [a,b,*], Xin Liu [a,b], Tianxin Liao [a,b], Zhenghao Liu [c], Mengyan Wang [a,b], Xuezhi Fang [a,b], Erhong Yang [a,b]

[a] *National Language Resources Monitoring and Research Center for Print Media, Beijing Language and Culture University, Beijing, China*
[b] *School of Information Science, Beijing Language and Culture University, Beijing, China*
[c] *School of Computer Science and Engineering, Northeastern University, Shenyang, China*

ARTICLE INFO

Keywords:
Chinese Spelling Check
YACSC
Real-world scenarios

ABSTRACT

The task of Chinese Spelling Check (CSC) is crucial for identifying and rectifying spelling errors in Chinese texts. While prior work in this domain has predominantly relied on benchmarks such as SIGHAN for evaluating model performance, these benchmarks often exhibit an imbalanced distribution of spelling errors. They are typically constructed under idealized conditions, presuming the presence of only spelling errors in the input text. This assumption does not hold in real-world scenarios, where spell checkers frequently encounter a mix of spelling and grammatical errors, thereby presenting additional challenges. To address this gap and create a more realistic testing environment, we introduce a high-quality CSC evaluation benchmark named YACSC (Yet Another Chinese Spelling Check Dataset). YACSC is unique in that it includes annotations for both grammatical and spelling errors, rendering it a more reliable benchmark for CSC tasks. Furthermore, we propose a hierarchical network designed to integrate multidimensional information, leveraging semantic and phonetic aspects, as well as the structural forms of Chinese characters, to enhance the detection and correction of spelling errors. Through extensive experiments, we evaluate the limitations of existing CSC benchmarks and illustrate the application of our proposed system in real-world scenarios, particularly as a preliminary stage in writing assistant systems.

1. Introduction

The primary objective set forth in the field of Chinese Spelling Check (CSC) is to discover and rectify spelling errors contained within Chinese written material. In the realm of Natural Language Processing (NLP), CSC has emerged as a significant research area due to common spelling issues that occur during manual writing, mistouches on typing devices, automatic speech recognition (Hartley and Reich, 2005), and optical character recognition (Afli et al., 2016). It has become a crucial component for numerous NLP tasks, including automatic essay scoring (Dong and Zhang, 2016), search query correction (Martins and Silva, 2004; Gao et al., 2010), and optical character recognition.

Traditionally, the evaluation of CSC model performance relied on benchmarks, among which SIGHAN (Wu et al., 2013; Yu et al., 2014; Tseng et al., 2015) is frequently considered as a reference. However, these benchmarks generally present an idealized distribution, wherein the input text has been pre-polished for grammatical errors, leaving only spelling errors for rectification. In real-life scenarios, Chinese spelling checkers often grapple with a mixture of spelling and grammatical errors, thereby creating additional complexities. Table 1 showcases the improved results when a CSC is executed prior to a grammatical error correction, signifying the importance of considering both grammatical and spelling errors during the CSC process.

To enhance the efficacy of spelling checks in a more realistic environment where both grammatical and spelling errors co-exist, we propose a novel approach to the CSC task. Whereas previous methodologies concentrated solely on spelling errors, our approach, by taking into account grammatical errors, aims to create more alignment with real-world spelling check scenarios.

In this study, we introduce a new paradigm for evaluating Chinese Spelling Check (CSC) models and present an evaluation dataset named **YACSC** (**Y**et **A**nother **C**hinese **S**pelling **C**heck Dataset), derived from the **YACLC** (**Y**et **A**nother **C**hinese **L**earner **C**orpus) (Wang et al., 2021). We meticulously selected 2550 sentence pairs apt for CSC, annotated them, and conducted comprehensive analyses to glean deeper insights.

* Corresponding author at: National Language Resources Monitoring and Research Center for Print Media, Beijing Language and Culture University, Beijing, China.

E-mail address: lineryang@gmail.com (L. Yang).

https://doi.org/10.1016/j.aiopen.2023.10.004
Received 5 October 2023; Received in revised form 26 October 2023; Accepted 30 October 2023
Available online 3 November 2023

Table 1
Different results of Chinese spelling check before or after grammatical error correction. The word "好象" is a grammatical error, and the character "象" is a spelling error. **GEC only** and **CSC only** refer to performing either grammatical error correction or spelling check individually. **CSC → GEC** indicates performing spelling check before grammatical error correction, while **GEC → CSC** indicates performing grammatical error correction before spell check.

Input	北京与西安有很多好象一样的地方 běi jīng yǔ xī ān yǒu hěn duō hǎo xiàng yī yàng de dì fāng
GEC only	北京与西安有很多好的地方 běi jīng yǔ xī ān yǒu hěn duō hǎo de dì fāng
CSC only	北京与西安有很多好像一样的地方 běi jīng yǔ xī ān yǒu hěn duō hǎo xiàng yī yàng de dì fāng
GEC→CSC	北京与西安有很多好的地方。 běi jīng yǔ xī ān yǒu hěn duō hǎo de dì fāng
CSC→GEC	北京与西安有很多相似的地方 běi jīng yǔ xī ān yǒu hěn duō xiāng sì de dì fāng

We provide a thorough exposition of the annotation specifications grounded in linguistic theories. In a novel approach, we annotated the YACSC dataset with simplified Chinese text, deliberately preserving grammatical errors to mimic real-world scenarios. Table 1 exemplifies this, showcasing a sentence with a spelling error ("像" instead of "象") and a grammatical error ("好像" incorrectly following "很多"). Additionally, we curated a subset of the dataset with corrected grammatical errors to evaluate existing CSC models in ideal conditions, akin to the SIGHAN benchmarks.

Contrastingly, prior works in this domain operated under ideal conditions, neglecting the influence of grammatical errors on spelling checks. In response, we introduce a cutting-edge Chinese spelling checker, a hierarchical network that integrates multidimensional information, serving as a baseline for our innovative paradigm. This model leverages BERT (Devlin et al., 2019) to capture contextual semantic nuances and employs a confusion encoder to encode both phonetic and visual aspects of Chinese characters, discerning the subtle relationships between them, including phonetic proximity and graphic proximity. Through a hierarchical encoder and a parameterized fusion mechanism, the model synthesizes information into a concise representation, facilitating word relationship establishment.

We subjected our model to rigorous evaluation on both the established SIGHAN benchmarks and our YACSC evaluation set. The results affirm that our model delivers competitive performance under ideal conditions on the SIGHAN benchmarks, showcasing its proficiency in spelling error correction. Notably, it surpasses other baseline models on the YACSC set, which is tailored to reflect real-world scenarios inclusive of grammatical errors. This underscores our model's capability to navigate the intricacies of such scenarios, leveraging the rich information embedded in Chinese characters. These outcomes underscore the practical applicability of our model in CSC tasks.

To encapsulate, the contributions of this work are threefold:

- We introduce a novel paradigm for CSC, enabling spelling checks in contexts marred by grammatical errors.
- We develop YACSC, a high-quality CSC evaluation set annotated with ungrammatical sentences from YACLC, providing a robust framework to assess CSC models under our paradigm.
- We propose an innovative Chinese spelling checker that concurrently encodes the phonetic and graphic dimensions of Chinese characters, demonstrating superior performance in both ideal and real-world scenarios compared to baseline models.

2. A real-world scenarios Chinese Spelling Check evaluation set: YACSC

Early Chinese spelling check datasets such as SIGHAN are usually built under ideal conditions, assuming the input contains spelling errors only. However, in actual application scenarios, the spelling check task will inevitably encounter sentences with grammatical errors. To test the effectiveness of existing CSC models in real-world scenarios and their potential to improve grammar correction, we propose a new evaluation set called YACSC. It is based on the Chinese learner corpus, YACLC (Wang et al., 2021), which includes multidimensional crowd-sourcing annotations for sentences written by Chinese language learners.

YACSC is a two-stage corpus for evaluating the performance of CSC models. The corpus consists of sentences with both spelling and grammatical errors, which are then corrected in two stages. The first stage corrects the spelling errors, while the second stage corrects grammatical errors. We also provide two subsets of YACSC, YACSC_w/_GE, and YACSC_w/o_GE, which evaluate the performance of the CSC model in real-world and ideal scenarios, respectively. The instances of YACSC can be found in Fig. 1.

2.1. Annotation specification

The goal of the CSC task is to identify and correct spelling mistakes that arise from phonetic and shape similarity. Previous research has created spelling check datasets and sets of confused Chinese characters, including phonological and morphological information. However, these studies have not established a clear criterion for judging phonetic similarity and shape similarity. From the perspective of linguistics, this paper attempts to summarize the principles for determining a group of Chinese characters as phonetic-like or shape-like characters during the tagging process, for reference.

In the annotating process, it is necessary to mark the words that have the relationship of *phonetic proximity* and *graphic proximity*.

Phonetic proximity. Modern Chinese syllables are composed of three components: initials, finals, and tones, each of which serves a unique function in differentiating between syllables. To ensure the accuracy of the labeling process, combined with the knowledge of Chinese phonology, we summarized the consonants proximity table and vowels proximity table. Additionally, we have established the following six principles to judge phonetic proximity (exclude initials, finals, and tones are totally the same):

- The finals are the same, and the initials are similar; Such as 决(jué)-确(què), which have the same finals *üe* and similar initials *j* and *q*.
- The initials are the same, and the finals are similar; Such as 既(jì)-经(jīng), which have the same initials *j* and similar finals *i* and *ing*.
- The initials are similar, the finals are similar, and the tones are the same; Such as 蘸(zhàn)-涮(shuàn), which have similar initials *zh* and *sh*, similar finals *an* and *uan*, tones are both fourth tones.

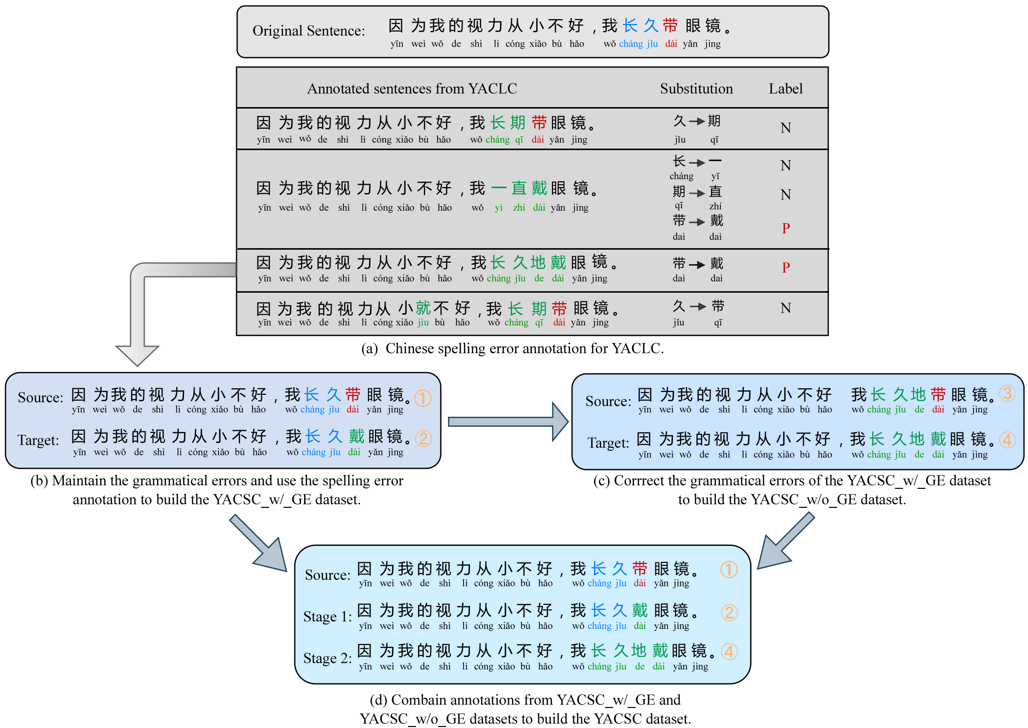

Fig. 1. The construction stages of our YACSC dataset. We first use the Levenshtein distance tool to annotate the substitution between the original sentences and annotated sentences and annotate the spelling errors (a). Then, we conduct two versions of YACSC datasets, YACSC_w/_GE (b) and YACSC_w/o_GE (c), which maintain the grammatical errors and correct the grammatical errors. Finally, both YACSC_w/_GE and YACSC_w/o_GE are combined to build the YACSC dataset (d).

- The initials are the same, the finals are the same, and the tones are different; Such as 招(zhāo)-照(zhào), which have the same initials *zh*, same finals *ao*, tones are first tones and fourth tones.
- Zero initial and non-zero initial syllables must have the same finals (Syllables starting with *y*, *w*, or *yu* are zero initial syllables); Such as 物(wù)-筑(zhù), which have Zero initial *w* and none-zero initial *zh* with same final *u*.
- The comparison between zero initials and zero initial syllables follows the principle of vowel similarity. Such as 应(yīng)-因(yīn), following the principle of vowel similarity, *in* and *ing* are similar.

The consonants proximity table and vowels proximity table are attached to Appendix.

Graphic proximity. The shape proximity is mainly determined according to the Chinese character components. Generally speaking, characters with the same main character-forming components are graphically similar. The main component refers to the part that occupies a more significant proportion of the area in the word formation. Following are some examples:

(1) 他一直[相|想]当修理工。
(2) 但是我不知道为什[幺|么]。
(3) 我的亲戚[太|大]多住在那。

Graphically similar characters are in []. The former character is the wrong character, whereas the latter is correct. "相" and "想" have the same component "相" in example (1); "幺" and "么" have the same component "厶" in example (2); "太" and "大" have the same component "大" in example (3).

2.2. Annotation process

Fig. 1 shows the construction process of YACSC. We built YACSC based on YACLC (Wang et al., 2021), a large-scale, multidimensional annotated Chinese learner corpus. YACLC contains 32,124 sentences written by Chinese as a Second Language (CSL) learners, and each sentence is annotated by ten annotators. We annotate on the published version of YACLC,[1] which contains the training, validation and test sets, to construct YACSC.

Initially, we employed the Levenshtein distance tool to identify substitutions between the original and corrected sentences from YACLC, subsequently requesting annotators to verify whether these substitutions rectified spelling errors according to the annotation specification in 2.1. We used classifications P, G, B, and *N* to differentiate specific connections in the substitutions based on phonetic similarity, graphic similarity, a blend of both, and instances without any spelling mistakes, correspondingly. Subsequently, we pinpointed 1275 sentences verified to have spelling errors, and for balance in our research, we randomly chose an equal number of sentences, confirmed to be free of spelling mistakes. This random sampling ensured the preservation of the targeted ratio of sentences with spelling errors in our study. Then, we applied the confirmed spelling corrections to the original sentence, ensuring grammatical errors remained, resulting in the YACSC_w/_GE dataset. Progressing from this, we rectified grammatical errors in both the source and target sentences of the YACSC_w/_GE dataset, culminating in the YACSC_w/o_GE dataset. The final step involved combining the source and target sentences from the YACSC_w/_GE dataset with the target sentences from the YACSC_w/o_GE dataset to construct the complete YACSC dataset.

Finally, we acquired the two-stage dataset YACSC, where the first stage focuses on correcting spelling errors, and the second stage addresses grammatical mistakes. We also created two subsets of YACSC based on whether the original sentence contained grammatical errors:

[1] https://github.com/blcuicall/YACLC

Table 2
Detail statistics of YACSC dataset.

Description	Statistics
Total sentences	2550
Sentences with spelling errors (%)	1275 (50%)
Sentences with grammatical errors (%)	1735 (68%)
Average length	22.95
Phonetic errors	967
Graphic errors	210
Both phonetic and graphic errors	312
Total errors	1489

- YACSC_w/o_GE: The grammatical errors in original sentences are corrected to validate how existing models perform under ideal conditions.
- YACSC_w/_GE: The original sentences have both grammatical and spelling errors, which are used to validate how existing models perform in real-world scenarios.

2.3. Statistics and analysis

Table 2 provides an overview of the statistics for YACSC, which consists of 2550 samples, with 50% of them being annotated for spelling mistakes. 68% of the sentences in YACSC_w/_GE have grammatical errors. In terms of error types, 65% of the typos are due to pure phonetic similarity, 20% is a result of both phonetic and graphic similarity, and the remaining typos are caused by pure graphic similarity.

Compared with the existing open-source evaluation sets, YACSC has several advantages. Firstly, different from SIGHAN benchmarks, in which the original data is in traditional Chinese and needs to be converted to simplified Chinese before use, the annotations are directly based on Simplified Chinese text, eliminating the noise introduced by the conversion between simplified and traditional Chinese or regional differences, as shown in Table 4. Secondly, spelling mistakes in the original sentences are annotated without modification, allowing for the presence of grammatical errors. This is more aligned with practical application scenarios in NLP tasks, as CSC models are often used in preprocessing stages and, therefore, encounter more complex situations than just misspelled sentences.

3. Methodology

In this section, we provide an in-depth overview of our proposed model. Our model is built on an observation that confusion between two characters can come from both their phonetics and shape. Liu et al. (2010) have shown that over a third of spelling errors are due to both pronunciation and shape. As shown in Fig. 2, "象(xiàng, elephant)" and "像(xiàng, seemingly)" not only have similar pronunciations but also the parts of "象" are the same. Common late fusion methods (Cheng et al., 2020; Xu et al., 2021) cannot capture these relationships as they integrate the result of encoding phonetics and shape as independent modules. Intuitively, by combining phonetic and shape information into a single input sequence, the attention mechanism can simultaneously perceive the sound and shape proximity to make a more accurate judgment. It makes it easier for the model to learn such confusing information.

3.1. Model architecture

Our model utilizes semantic, phonetic, and graphic information to distinguish the similarities between Chinese characters and correct spelling errors. As shown in Fig. 2, we first employ a BERT-based semantic encoder and a Transformer-based confusion encoder to capture valuable contextual, phonetic, and graphic information. The outputs of the two encoders are combined using a fusion module to generate context-aware representations of character confusion. Finally, the output layer predicts the probability of correction.

Semantic encoder. Judging the correctness of typos in text involves considering the difference in semantics between similar phonetics or glyphs. To accurately identify spelling errors, it is crucial to assess the fluency of the current character in the context of the sentence. Context-free static word embedding is unable to take into account the current textual information, so it is a better choice to use a context-dependent dynamic pre-training model.

We adopt BERT (Devlin et al., 2019) to develop the semantic encoder. BERT provides rich contextual word representation with unsupervised pre-training on large corpora. Note that, as we show in Fig. 2, using only the PTM-based semantic encoder, the model tends to modify spelling errors into more frequent words.

Confusion encoder. To effectively integrate phonetic and glyph information, we utilize the Transformer encoder, encompassing both a character-level encoder and a sequence-level encoder, forming a hierarchical relationship as depicted in Fig. 2. The input for the confusion encoder is a rich information sequence of the character, which is composed of three key elements. The initial element is the Chinese character itself, followed by the constituent components of the word,[2] and finally the pinyin.[3] To establish a unique identity for glyphs and phonemes, we also incorporate type encodings. Specifically, the type index for the glyph segment is set to 0, while for the phonetic segment, it is designated as 1. Both the type encodings and the input sequence are transformed into vectors through random initialization.

In contrast to the Transformer encoder, which employs a self-attention mechanism, we adopt a local attention mechanism within the character-level confusing information encoder. As depicted in Fig. 2, we illustrate a scenario where we calculate the attention scores for all the pinyin and Chinese character components of the input character "象" (xiàng, elephant). In contrast to self-attention, where the input character itself often receives the highest attention score, we mask itself when calculating the attention score. This clever adjustment makes the local attention mechanism more adept at directing the model's focus toward pronunciation or glyph information, which frequently underlies the perplexing relationships leading to spelling errors.

By utilizing the transformer encoder and a local attention mechanism, the model adeptly captures confusion information between characters. For instance, the characters "象" (xiàng, elephant) and "像" (xiàng, similar) share the same components, including "刀", "口", "|", "豕", and the identical pinyin "xiàng". These relationships are effectively captured by the model.

Fusion module. With previously mentioned semantic and confusion encoders, we get the representation vectors $\mathbf{H}^s$ and $\mathbf{H}^c$ to encode contextual information and confusion information. We develop a selective fusion module to integrate these two embeddings to predict the correct Chinese characters. A selective gate unit is employed to mix semantic information and confusing relationships. Parameters of the gating mechanism are computed by a fully connected layer followed by a sigmoid function. Following the method of ReaLiSe (Xu et al., 2021), the inputs include the character representation from the confusion encoder and the mean of the semantic encoder output $\mathbf{H}^s$ to get the overall semantic information of the input text. Formally, we denote the parameters of the gating unit for contextual and confusing information as g^s and g^c. The mixed representation is computed as follows:

$$\overline{h}^s = \frac{1}{N}\sum_{i=1}^{N} h_i^s \tag{1}$$

$$g_i^s = \sigma(\mathbf{W}^s \cdot [h_i^s, h_i^c, \overline{h}^s] + b^s) \tag{2}$$

$$g_i^c = \sigma(\mathbf{W}^c \cdot [h_i^s, h_i^c, \overline{h}^s] + b^c) \tag{3}$$

$$\widetilde{h}_i = g_i^s \cdot h_i^s + g_i^c \cdot h_i^c \tag{4}$$

[2] https://github.com/kfcd/chaizi
[3] https://github.com/hotoo/pinyin

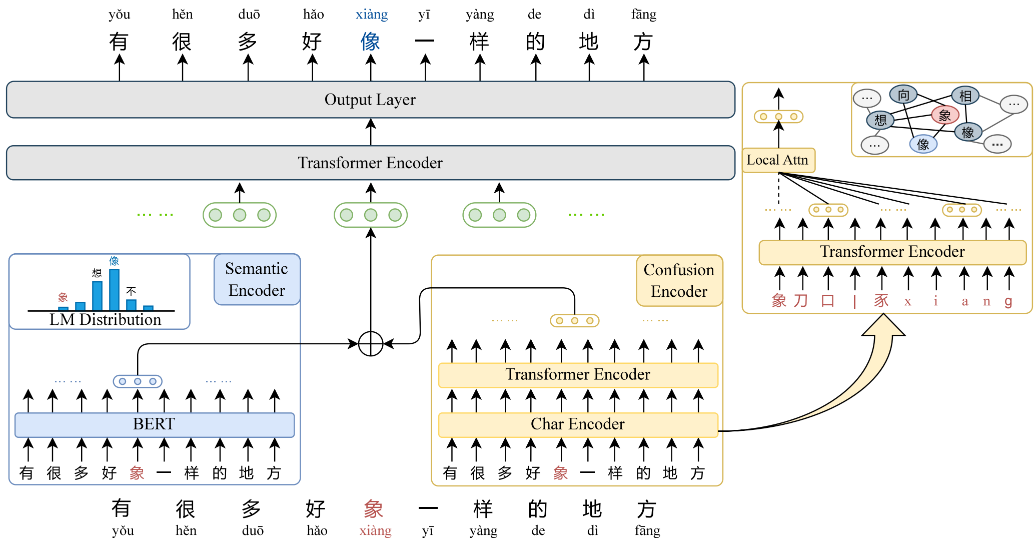

Fig. 2. The architecture overview of our proposed model. The semantic/confusion encoders are used to capture the contextual/phonetic and graphic information. We adopt a local attention mechanism within the confusion encoder to encode the phonetic and glyph information. And a gating mechanism is employed to selectively integrate the information from two encoders.

where $\mathbf{W}^s$, $\mathbf{W}^c$, b^s, b^c are learnable parameters, σ is the sigmoid function, and $[\cdot]$ means the concatenation of vectors. Then, at the sentence level, Transformer is applied to fully learn the semantic and confusing information. The mixed representations of the whole sentence are packed together into $\mathbf{H}_0 = [\widetilde{h}_1, \widetilde{h}_2, \ldots, \widetilde{h}_N]$, and the probability distribution $\hat{y}_i$ of what the ith character should be represented as:

$$\mathbf{H}_l = \mathbf{Transformer}_l(\mathbf{H}_{l-1}), l \in [1, L'] \tag{5}$$

$$\hat{y}_i = \mathbf{softmax}(\mathbf{W}^o h_i + b^o), h_i \in \mathbf{H}_{L'} \tag{6}$$

where L' is the number of Transformer layers, $\mathbf{W}^o$ and b^o are learnable parameters.

4. Experiments

In this section, we introduce experimental details and results on the SIGHAN benchmarks (Wu et al., 2013; Yu and Li, 2014; Tseng et al., 2015) and our newly constructed YACSC evaluation set. Then, we conduct detailed analyses, follow-up experiments and ablation studies to verify the contribution of our method.

4.1. Datasets and baseline methods

Train data. We use the same training data following previous works (Xu et al., 2021; Zhang et al., 2020), including the SIGHAN training samples (Wu et al., 2013; Yu and Li, 2014; Tseng et al., 2015) and the pseudo training samples automatically generated by OCR-based and ASR-based methods (Wang et al., 2018). Statistics of the datasets are shown in Table 3. All the training data are merged to train our proposed model and other baseline models.

Table 3
Statistics of the used public training datasets.

Training set	# Sent	Avg. length	# Errors
SIGHAN13	700	41.8	343
SIGHAN14	3437	49.6	5122
SIGHAN15	2338	31.3	3037
Wang271K	271,329	42.6	381,962
Total	277,804	42.6	390,464

Test data. We evaluate our model on the SIGHAN benchmarks (Wu et al., 2013; Yu and Li, 2014; Tseng et al., 2015) and our newly constructed YACSC evaluation set. Originally, the SIGHAN datasets are in the traditional Chinese. Most prior studies (Wang et al., 2019; Zhang et al., 2020; Cheng et al., 2020; Xu et al., 2021) are based on the converted simplified Chinese version by openCC.[4] However, the conversion to simplified Chinese introduces a lot of noise. Multiple traditional Chinese characters may correspond to a single simplified Chinese character after conversion. Thus, a confusion pair may become invalid after simplification. For example, as shown in Table 4, "復習(review)" and "複習(review)" could have constituted an error pair "復-複", but both "復" and "複" are converted to "复" in Simplified Chinese. And we also observed that there are some unreasonable annotations in the SIGHAN benchmarks. Some of the main reasons include similarity in sound ("的"/"地"/"得"), similarity in form ("约"/"钓"), and grammatical errors ("摄影机"/"影机"). These issues undermine the reliability of model evaluation, to a certain extent.

To address the above-mentioned issues in the SIGHAN test sets, including mislabeling and omission, etc., we conduct a revision to improve the confidence of evaluation results. The before-and-after revision statistics of the SIGHAN test sets are shown in Table 5. From the statistics, we can see that, as we mentioned above, the original SIGAHN test sets have a significant number of unmarked errors. So, the results tested on the original benchmark inevitably have some distortion.

We evaluate our model on the original/revised SIGHAN test sets and YACSC test sets. The test sets are used separately to evaluate the model performance.

Metrics. Results are reported at the detection level and the correction level. At the detection level, a sentence is considered to be correct if and only if all the spelling errors are detected successfully. At the correction level, the model must not only detect but also correct all the erroneous characters to the right ones. We report the precision, recall, and F1 scores on both levels. Note that Xu et al. (2021) proposed to remove all the detected and corrected "的", "地", and "得" characters

[4] https://github.com/BYVoid/OpenCC

Table 4
A bad case in SIGHAN15 test set. “復習” and “複習” are both converted to “ 复习”.

Traditional input	明天要期末考，我需要復習。 míng tiān yào qī mò kǎo , wǒ xū yào fù xí .
Traditional target	明天要期末考，我需要複習。 míng tiān yào qī mò kǎo , wǒ xū yào fù xí .
Simplified input	明天要期末考，我需要复习。 míng tiān yào qī mò kǎo , wǒ xū yào fù xí .
Simplified target	明天要期末考，我需要复习。 míng tiān yào qī mò kǎo , wǒ xū yào fù xí .

Table 5
Statistics of SIGHAN test sets before and after revision.

Original	# Sent	# Erroneous sent	# Errors
SIGHAN13	1000	1000	1224
SIGHAN14	1062	531	771
SIGHAN15	1100	550	703
Revised	# Sent	# Erroneous sent	# Errors
SIGHAN13	1000	977	1483
SIGHAN14	1062	602	932
SIGHAN15	1100	618	858

from the model output on SIGHAN13 test set because of the poor annotation quality. However, we refrain from this process to ensure a fair comparison of model performance on the original and revised SIGHAN test sets.

Baseline methods. To evaluate the performance of our method, we select several advanced strong baseline methods: *BERT* (Devlin et al., 2019) is to directly fine-tune the PTM with the training data. *ReaLiSe* (Xu et al., 2021) captures and mixes multimodal knowledge to improve CSC performance, which is the previous state-of-the-art method on SIGHAN benchmarks.

4.2. Experimental setup

Our method is implemented using PyTorch framework (Paszke et al., 2019) with the Transformer library (Wolf et al., 2020). Microsoft's deepspeed (Rasley et al., 2020) is employed for parallel training and mixed-precision acceleration. We chose the whole-word-masking pretraining model chinese-bert-wwm-ext (Cui et al., 2021) as the backbone of the semantic encoder. For the confusion encoder, a two layers transformer encoder with two attention heads is applied for character-level and a four layers transformer encoder with eight attention heads for sentence-level. The fusion module has two transformer layers with 8 attention heads. All the embeddings and hidden states have a dimension of 768. We train our model with the AdamW (Loshchilov and Hutter, 2017) optimizer for ten epochs. The learning rate is set to 1e−5, the batch size is set to 96, and the accumulation step is set to 2. We would shuffle all the training data before training.

4.3. Main results

Table 6 shows the evaluation results at detection and correction levels on the original/revised SIGHAN test sets and our constructed YACSC evaluation sets.

When comparing the performance of models on the original and revised SIGHAN test sets, all models exhibit an average drop of 4.26% F1 score on the correction level. This demonstrates that the revised SIGHAN test sets are more challenging than the original version, highlighting the unreliability of the original SIGHAN test sets.

Our proposed model shows competitive ability against the baseline models under ideal conditions on the revised SIGHAN test sets and YACSC_w/o_GE test set. Moreover, on the real-world scenario evaluation set YACSC_w/_GE that we proposed, our model performs significantly better than BERT and ReaLiSe (Xu et al., 2021). By integrating the rich information of Chinese characters instead of just using context information, our model demonstrates excellent performance with a large margin against BERT. By jointly encoding character components and pinyin using the Transformer encoder, the experimental results demonstrate that our method utilizes the phonetic and graphic proximity information more effectively than ReaLiSe (Xu et al., 2021).

The results in the last two sections of Table 6 show that the performance of models on the YACSC-w/o_GE set is significantly higher than on the YACSC_w/_GE set. This highlights the difficulty of the CSC task in real-world scenarios and suggests that the old ideal building benchmarks may not be suitable for these scenarios. At the same time, the performance of our proposed model drops the letter than ReaLiSe (Xu et al., 2021) when faced with scenes with grammatical errors and achieves the best results, proving better robustness and capacity of our method in this more realistic evaluation scenario.

As the Spelling Check system is commonly used in conjunction with a Grammar Error Correction (GEC) model in the text correction process, we further investigate different strategies for enhancing the performance of text correction on the YACSC_w/_GE evaluation set. Given that GEC models (Omelianchuk et al., 2020; Liu et al., 2021; Yang et al., 2022; Bryant et al., 2023) now use pre-trained language models like BERT as the semantic information extractor, spelling errors in texts will bring significant noise to the output of the pre-trained language model, which will negatively impact the subsequent grammar error correction. We believe that correcting spelling errors in sentences before they are sent to the GEC model can significantly improve the text correction performance. Table 7 shows the comparison of different strategies of text correction. The results show that using a CSC model for preprocessing achieves better performance overall, which also proves that a CSC dataset without pre-modification of its original input is more suitable in real-world scenarios. Moreover, our method's superiority in handling real-world scenarios is demonstrated by the significant improvement in the $F_{0.5}$ score achieved through the use of our model for preprocessing.

4.4. Ablation study

We analyze the effect of several fusion strategies, including the semantic encoder, the concatenation of the semantic encoder and confusion encoder, the sum of the semantic encoder and confusion encoder, and the gating mechanism. The results are illustrated in Table 8, proving that the scheme using the gating mechanism is the optimal fusion method. Compared with the strategy using only the semantic encoder, the gating mechanism also shows certain advantages, which also reflects the effectiveness of joint modeling with confusing information.

Table 6
The performance of our model and all baseline models on the original/revised SIGHAN test sets and our newly constructed YACSC evaluation set. We show the ΔF1 score on the correction level for the original and revised SIGHAN test sets, YACSC-w/o_GE and YACSC-w/_GE, respectively.

Dataset	Method	Detection level			Correction level		
		Precision	Recall	F1	Precision	Recall	F1
SIGHAN13	BERT	77.75	**74.15**	**75.91**	**76.78**	**73.22**	**74.96**
	ReaLiSe	**78.77**	72.61	75.56	76.42	70.44	73.31
	Ours	75.90	69.72	72.67	75.22	69.10	72.03
SIGHAN14	BERT	65.03	68.65	66.79	63.75	**67.31**	65.48
	ReaLiSe	62.86	**76.69**	65.19	61.43	66.15	63.70
	Ours	**67.12**	66.73	**66.92**	**66.34**	65.96	**66.15**
SIGHAN15	BERT	76.14	80.22	78.13	74.04	78.00	75.97
	ReaLiSe	76.08	**81.15**	78.53	74.52	**79.48**	**76.92**
	Ours	**76.65**	77.08	76.87	**76.10**	76.52	76.31
SIGHAN13_REVISED	BERT	72.57	68.78	70.63	71.06	67.35	69.15 (↓ 5.81)
	ReaLiSe	**74.86**	**70.73**	**72.74**	71.94	**67.96**	69.89 (↓ 3.42)
	Ours	74.44	68.27	71.22	**73.21**	67.14	**70.04** (↓ 1.99)
SIGHAN14_REVISED	BERT	64.56	59.30	61.82	62.57	57.48	59.91 (↓ 5.57)
	ReaLiSe	65.59	**60.80**	**63.10**	63.62	**58.97**	61.21 (↓ 2.49)
	Ours	**67.18**	58.13	62.33	**66.03**	57.14	**61.26** (↓ 4.89)
SIGHAN15_REVISED	BERT	75.44	69.58	72.39	73.16	67.48	70.20 (↓ 5.77)
	ReaLiSe	75.78	**70.87**	**73.24**	74.39	**69.58**	71.91 (↓ 5.01)
	Ours	**77.72**	68.28	72.70	**76.98**	67.64	**72.01** (↓ 4.30)
YACSC_w/o_GE	BERT	63.67	41.10	49.95	56.38	36.39	44.23
	ReaLiSe	67.93	**47.69**	**56.04**	60.34	**42.35**	49.77
	Ours	**71.62**	42.35	53.23	**68.30**	40.39	**50.76**
YACSC_w/_GE	BERT	56.43	38.20	45.56	49.59	33.57	40.04 (↓ 4.19)
	ReaLiSe	57.71	**43.14**	**49.37**	51.31	**38.35**	43.90 (↓ 5.87)
	Ours	**63.94**	39.92	49.15	**61.06**	38.12	**46.93** (↓ 3.83)

Table 7
Results of different strategies for text correction. GEC-only indicates that we only use the GEC system to correct errors. The third and fourth rows refer to a GEC system followed by a CSC system for error correction, and the CSC model here is our proposed model or ReaLiSe, and the GEC model is GECToR (Omelianchuk et al., 2020). The fifth and sixth rows mean the opposite.

Strategy	Precision	Recall	$F_{0.5}$
GEC-only	31.48	16.00	26.38
GEC→ReaLiSe	32.90	19.89	29.09
GEC→Ours	34.00	19.77	29.72
ReaLiSe→GEC	35.08	21.23	31.03
Ours→GEC	**36.76**	**21.44**	**32.16**

5. Related work

The SIGHAN13, 14, and 15 benchmarks (Wu et al., 2013; Yu et al., 2014; Tseng et al., 2015) are predominant in the field, exclusively focusing on spelling errors while excluding other syntactical inaccuracies or inappropriate co-occurrences. Wang et al. (2018) addressed the scarcity of training data by generating synthetic datasets through automated methods. A significant portion of recent data-driven research utilizes the SIGHAN datasets and the data from Wang et al. (2018) as training material (Zhang et al., 2020; Cheng et al., 2020; Xu et al., 2021; Li et al., 2022), showcasing the positive impact of expanded data size on Chinese Spelling Correction (CSC) performance. Hu et al. (2022) introduced CSCD-IME, a dataset tailored for CSC errors stemming from pinyin Input Method Editors (IME), and proposed a method for pseudo-data generation by simulating pinyin IME inputs. This approach is specifically designed for CSC in the native Chinese language domain. Additionally, Jiang et al. (2022) presented MCSCSet, a comprehensive dataset annotated by experts for CSC within the medical domain.

Historically, CSC relied on unsupervised language models (Liu et al., 2013; Yu and Li, 2014), rule-based techniques (Chang et al., 2015; Chu and Lin, 2015) and conventional machine learning approaches (Wang and Liao, 2015; Xiong et al., 2015). The advent of deep learning in Natural Language Processing (NLP) has led to significant advancements in CSC through neural-based methodologies. Wang et al. (2018) utilized discriminative sequence tagging and a Bi-LSTM network for CSC tasks. Treating CSC as a translation task within the same language, Wang et al. (2019) applied the seq2seq model for correction purposes. The emergence of pre-trained language models (PLMs) like BERT (Devlin et al., 2019), has established PLMs as encoders as a prevalent strategy in CSC. FASPell, a model proposed by Hong et al. (2019) combines a DAE-Decoder structure with a BERT-based encoder. Zhang et al. (2020) introduced a Bi-GRU detection network to produce masking vectors for a BERT-based correction network, while Lin et al. (2022) proposed a reverse contrastive learning approach for CSC.

The integration of external character similarity knowledge has recently garnered attention. This knowledge is often structured into confusion sets containing pairs of similar characters. Yu and Li (2014) initially utilized manually created confusion sets to identify potential errors. Wang et al. (2019) incorporated a pointer network (Vinyals et al., 2015) to directly copy characters from the confusion set to the target sentence. Cheng et al. (2020) introduced SpellGCN, a model that utilizes Graph Convolution Networks (Kipf and Welling, 2017) to represent pronunciation and shape similarities within the confusion set. Moving away from static confusion sets, Nguyen et al. (2021) applied TreeLSTM (Tai et al., 2015; Zhu et al., 2015) to filter correction candidates based on learned confusion relations, while Xu et al. (2021) explored multimodal knowledge to discern subtle similarities between Chinese characters.

Despite the impressive capabilities of Large Language Models (LLMs) like GPT-3 in various NLP tasks, Li et al. (2023) conducted an in-depth analysis of LLMs' performance on CSC tasks, uncovering challenges and areas for improvement in their application to CSC.

6. Conclusion

In this paper, We propose a new paradigm for the Chinese Spelling Check (CSC) task, which conducts spell checking directly in scenarios where grammatical errors exist. This paradigm is more in line with the demands of real-world scenarios. We also introduce YACSC, a comprehensive CSC evaluation set comprising 2550 sentences from Chinese as a Second Language (CSL) learners. YACSC offers a more

Table 8
The ablation results for fusion strategies (%). Concatenation means the concatenation of the semantic encoder and confusion encoder. Sum denotes the sum of the semantic encoder and confusion encoder.

Dataset	Fusion strategy	Detection level			Correction level		
		Precision	Recall	F1	Precision	Recall	F1
YACSC_w/o_GE	Semantic encoder	63.02	40.24	49.11	57.99	37.02	45.19
	Sum	68.49	42.27	52.28	64.55	39.84	49.27
	Concatenation	58.51	35.84	44.46	54.16	33.17	41.15
	Gating mechanism	**71.62**	**42.35**	**53.23**	**68.30**	40.39	**50.76**
YACSC_w/_GE	Semantic encoder	53.44	38.43	44.71	48.09	34.59	40.24
	Sum	61.96	39.22	48.03	57.62	36.47	44.67
	Concatenation	50.29	34.27	40.76	45.68	31.14	37.03
	Gating mechanism	**63.94**	**39.92**	**49.15**	**61.06**	**38.12**	**46.93**

dependable evaluation compared to existing benchmarks, thanks to two key features: (1) It highlights spelling errors in the original sentences while preserving any grammatical mistakes, ensuring alignment with real-world use cases. (2) It focuses on a simplified corpus from Chinese language learners, minimizing the noise from simplified-traditional Chinese conversions. Additionally, we provide a subset with corrected grammatical errors, akin to the SIGHAN benchmark, to assess how current models fare in optimal conditions. We also propose a new CSC model that effectively integrates contextual, phonetic, and graphic information for error detection and correction. Our comprehensive experiments and analyses demonstrate that our model not only excels in ideal conditions but also exhibits remarkable resilience in practical settings. The dataset and evaluation scripts are accessible at https://github.com/blcuicall/yacsc.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Natural Science Foundation of China (No. 62206042) and the MOE Project of Humanities and Social Sciences, China (No. 23YJCZH264). We sincerely appreciate all of the reviewers for their valuable suggestions.

Appendix. Consonants proximity table and vowels proximity table

We synthesize the phonetic proximity types and summarize them in the following consonant proximity phonetics table. As illustrated in Fig. A.1 the consonants marked as 1 in the table have a close relationship, and a null value indicates dissimilarity.

	b	p	m	f	z	c	s	d	t	n	l	zh	ch	sh	r	j	q	x	g	k	h
b	1	1	1					1													
p	1	1							1												
m	1	1																			
f				1																	1
z					1	1	1					1				1					
c					1	1	1						1				1				
s					1	1	1							1							
d	1							1	1	1	1										
t		1						1	1	1	1										
n								1	1	1	1										
l								1	1	1	1				1						
zh					1							1	1	1							
ch						1						1	1	1							
sh							1					1	1	1							
r												1	1	1	1						
j					1											1	1	1			
q						1										1	1	1			
x							1									1	1	1			
g																			1	1	1
k																			1	1	1
h				1															1	1	1

Fig. A.1. Consonants Proximity Table. The consonants marked as 1 in the table have a close relationship, and a null value indicates dissimilarity.

During the labeling process, the phonemic variants of the vowels in the finals need to be considered. Based on the phonetic error rules of the finals in Song (2000), we have summed up the vowels proximity table, as shown in Fig. A.2.

a	ia	ua	ai	ao	an	ang	ə	**uo**	o	e	ua	ao	ə	
o	uo	ou						**uai**	ai	uei	ei	ua		
e	ê	er	uo	ei	en	eng		**uei**	ei	uai	ai			
ê	e	iê	üe					**üe**	ê	e	üan	ie		
er	e							**an**	a	en	ian	uan	üan	ang
i	ü	in	ing	ə				**en**	e	an	uen	eng		
-i(front)	-i(back)							**ang**	a	an	iang	uang	eng	
-i(back)	-i(front)							**eng**	e	en	ang	ueng	ə	
u	iu	ong	ə					**ian**	an	ia	üan	in	iang	uan
ü	i	ün	iong					**in**	i	ing	ian	ün		
ai	a	ei	uai	uei				**iang**	ia	ang	ing	ian		
ei	e	ai	uei	uai				**ing**	i	in	iang	eng	iong	
ao	a	iao	ou	iou	uo			**uan**	an	uang	uen	ua	ian	
ou	o	ao	iou	iao				**uen**	en	uan	ueng			
ia	a	ie						**uang**	ua	ang	uan	ueng	ong	
ie	ê	ia	üe					**ueng**	uen	eng	ong			
iao	ao	ou	iou	ia				**ong**	uang	ueng	eng			
iou	ou	ao	iao					**üan**	an	ian	ün			
ua	a	uo	uang	uan				**ün**	in	üan	iong			
								iong	ün	ing				

Fig. A.2. Vowels Proximity Table. There are similarity between the bolded vowel and the vowels following it.

References

Afli, H., Qiu, Z., Way, A., Sheridan, P., 2016. Using SMT for OCR error correction of historical texts. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation.

Bryant, C., Yuan, Z., Qorib, M.R., Cao, H., Ng, H.T., Briscoe, T., 2023. Grammatical error correction: A survey of the state of the art. Comput. Linguist. 1–59.

Chang, T.-H., Chen, H.-C., Yang, C.-H., 2015. Introduction to a proofreading tool for Chinese spelling check task of SIGHAN-8. In: Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing.

Cheng, X., Xu, W., Chen, K., Jiang, S., Wang, F., Wang, T., Chu, W., Qi, Y., 2020. SpellGCN: Incorporating phonological and visual similarities into language models for Chinese spelling check. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

Chu, W.-C., Lin, C.-J., 2015. NTOU Chinese spelling check system in sighan-8 bake-off. In: Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing.

Cui, Y., Che, W., Liu, T., Qin, B., Yang, Z., 2021. Pre-training with whole word masking for Chinese BERT. IEEE/ACM Trans. Audio Speech Lang. Process.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.

Dong, F., Zhang, Y., 2016. Automatic features for essay scoring – an empirical study. In: Proceedings of the Empirical Methods in Natural Language Processing.

Gao, J., Quirk, C., et al., 2010. A large scale ranker-based system for search query spelling correction. In: Proceedings of the 23rd International Conference on Computational Linguistics.

Hartley, M.W., Reich, D.E., 2005. Method and System for Speech Recognition Using Phonetically Similar Word Alternatives. Google Patents.

Hong, Y., Yu, X., He, N., Liu, N., Liu, J., 2019. FASPell: A fast, adaptable, simple, powerful Chinese spell checker based on DAE-decoder paradigm. In: Proceedings of the 5th Workshop on Noisy User-Generated Text.

Hu, Y., Meng, F., Zhou, J., 2022. CSCD-IME: Correcting spelling errors generated by pinyin IME. arXiv preprint arXiv:2211.08788.

Jiang, W., Ye, Z., Ou, Z., Zhao, R., Zheng, J., Liu, Y., Liu, B., Li, S., Yang, Y., Zheng, Y., 2022. Mcscset: A specialist-annotated dataset for medical-domain Chinese spelling correction. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management.

Kipf, T.N., Welling, M., 2017. Semi-supervised classification with graph convolutional networks. In: Proceedings of International Conference on Learning Representations.

Li, Y., Huang, H., Ma, S., Jiang, Y., Li, Y., Zhou, F., Zheng, H., Zhou, Q., 2023. On the (in)effectiveness of large language models for Chinese text correction. arXiv preprint arXiv:2307.09007.

Li, Y., Zhou, Q., Li, Y., Li, Z., Liu, R., Sun, R., Wang, Z., Li, C., Cao, Y., Zheng, H.-T., 2022. The past mistake is the future wisdom: Error-driven contrastive probability optimization for Chinese spell checking. In: Findings of the Association for Computational Linguistics.

Lin, N., Fu, S., Lin, X., Jiang, S., Yang, A., 2022. A Chinese spelling check framework based on reverse contrastive learning. arXiv preprint arXiv:2210.13823.

Liu, X., Cheng, K., Luo, Y., Duh, K., Matsumoto, Y., 2013. A hybrid Chinese spelling correction using language model and statistical machine translation with reranking. In: Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing.

Liu, C.-L., Lai, M.-H., Chuang, Y.-H., Lee, C.-Y., 2010. Visually and phonologically similar characters in incorrect simplified Chinese words. In: Proceedings of the 23rd International Conference on Computational Linguistics.

Liu, Z., Yi, X., Sun, M., Yang, L., Chua, T.-S., 2021. Neural quality estimation with multiple hypotheses for grammatical error correction. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.

Loshchilov, I., Hutter, F., 2017. Decoupled weight decay regularization. In: Proceedings of International Conference on Learning Representations.

Martins, B., Silva, M.J., 2004. Spelling correction for search engine queries. In: Proceedings of International Conference on Natural Language Processing.

Nguyen, M., Ngo, G.H., Chen, N.F., 2021. Domain-shift conditioning using adaptable filtering via hierarchical embeddings for robust Chinese spell check. IEEE/ACM Trans. Audio Speech Lang. Process..

Omelianchuk, K., Atrasevych, V., Chernodub, A., Skurzhanskyi, O., 2020. GECToR – grammatical error correction: Tag, not rewrite. In: Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library. In: Proceedings of Advances in Neural Information Processing Systems.

Rasley, J., Rajbhandari, S., Ruwase, O., He, Y., 2020. DeepSpeed: System optimizations enable training deep learning models with over 100 billion parameters. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.

Song, X., 2000. A Practical Handbook for Mandarin Proficiency Testers. The Commercial Press.

Tai, K.S., Socher, R., Manning, C.D., 2015. Improved semantic representations from tree-structured long short-term memory networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing.

Tseng, Y.-H., Lee, L.-H., Chang, L.-P., Chen, H.-H., 2015. Introduction to SIGHAN 2015 bake-off for Chinese spelling check. In: Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing.

Vinyals, O., Fortunato, M., Jaitly, N., 2015. Pointer networks. In: Proceedings of Advances in Neural Information Processing Systems.

Wang, Y., Kong, C., Yang, L., Wang, Y., Lu, X., Hu, R., He, S., Liu, Z., Chen, Y., Yang, E., Sun, M., 2021. YACLC: a Chinese learner corpus with multidimensional annotation. arXiv preprint arXiv:2112.15043.

Wang, Y.-R., Liao, Y.-F., 2015. Word vector/conditional random field-based Chinese spelling error detection for SIGHAN-2015 evaluation. In: Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing.

Wang, D., Song, Y., Li, J., Han, J., Zhang, H., 2018. A hybrid approach to automatic corpus generation for Chinese spelling check. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing.

Wang, D., Tay, Y., Zhong, L., 2019. Confusionset-guided pointer networks for Chinese spelling check. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al., 2020. Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations.

Wu, S.-H., Liu, C.-L., Lee, L.-H., 2013. Chinese spelling check evaluation at SIGHAN bake-off 2013.. In: Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing.

Xiong, J., Zhang, Q., Zhang, S., Hou, J., Cheng, X., 2015. Hanspeller: A unified framework for Chinese spelling correction. Int. J. Comput. Linguist. Chin. Lang. Process. Special Issue Chin. Foreign Lang.

Xu, H.-D., Li, Z., Zhou, Q., Li, C., Wang, Z., Cao, Y., Huang, H., Mao, X.-L., 2021. Read, listen, and see: Leveraging multimodal information helps Chinese spell checking. In: Findings of the Association for Computational Linguistics. ACL-IJCNLP.

Yang, L., andYun Chen, C.W., Du, Y., Yang, E., 2022. Controllable data synthesis method for grammatical error correction. Front. Comput. Sci.

Yu, L.-C., Lee, L.-H., Tseng, Y.-H., Chen, H.-H., 2014. Overview of SIGHAN 2014 bake-off for Chinese spelling check. In: Proceedings of the Third CIPS-SIGHAN Joint Conference on Chinese Language Processing.

Yu, J., Li, Z., 2014. Chinese spelling error detection and correction based on language model, pronunciation, and shape. In: Proceedings of the Third CIPS-SIGHAN Joint Conference on Chinese Language Processing.

Zhang, S., Huang, H., Liu, J., Li, H., 2020. Spelling error correction with soft-masked BERT. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

Zhu, X., Sobihani, P., Guo, H., 2015. Long short-term memory over recursive structures. In: Proceedings of International Conference on Machine Learning.