Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

DRMSpell: dynamically reweighting multimodality for Chinese spelling correction*

Yinghao LI[1], Heyan HUANG[1,2], Baojun WANG[3], Yang GAO[‡1,2]

[1]*School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China*

[2]*Southeast Academy of Information Technology, Beijing Institute of Technology, Putian 351100, China*

[3]*Huawei Noahs Ark Lab, Shenzhen 518129, China*

E-mail: yhli@bit.edu.cn; hhy63@bit.edu.cn; puking.w@huawei.com; gyang@bit.edu.cn

Received Dec. 1, 2023; Revision accepted May 6, 2024; Crosschecked Feb. 11, 2025

Abstract: Chinese spelling correction (CSC) is a task that aims to detect and correct the spelling errors that may occur in Chinese texts. However, the Chinese language exhibits a high degree of complexity, characterized by the presence of multiple phonetic representations known as pinyin, which possess distinct tonal variations that can correspond to various characters. Given the complexity inherent in the Chinese language, the CSC task becomes imperative for ensuring the accuracy and clarity of written communication. Recent research has included external knowledge into the model using phonological and visual modalities. However, these methods do not effectively target the utilization of modality information to address the different types of errors. In this paper, we propose a multimodal pretrained language model called DRMSpell for CSC, which takes into consideration the interaction between the modalities. A dynamically reweighting multimodality (DRM) module is introduced to reweight various modalities for obtaining more multimodal information. To fully use the multimodal information obtained and to further strengthen the model, an independent-modality masking strategy (IMS) is proposed to independently mask three modalities of a token in the pretraining stage. Our method achieves state-of-the-art performance on most metrics constituting widely used benchmarks. The findings of the experiments demonstrate that our method is capable of modeling the interactive information between modalities and is also robust to incorrect modal information.

Key words: Chinese spelling correction; Multimodality; Masking strategy
https://doi.org/10.1631/FITEE.2300816 **CLC number:** TP391.1

1 Introduction

Chinese spelling correction (CSC) is a crucial task in the field of natural language processing, as it plays a vital role in ensuring the accuracy and clarity of written communication in Chinese (Cheng et al., 2020; Xu et al., 2021). CSC aims to detect and correct spelling errors that may occur in Chinese texts, which can be particularly challenging due to the nature of Chinese characters represented as pictographs (Liu et al., 2021). For instance, Chinese characters may possess multiple phonetic representations known as pinyin, where each pinyin comprises distinct tonal variations. These tonal variations can also correspond to different characters (Sun et al., 2021). Given these complexities, texts in the Chinese language frequently encounter spelling errors that are both phonetically and visually similar, as illustrated in Fig. 1. Additionally, the meaning of a sentence is modified dramatically when some characters are incorrect. This is a scenario wherein the given context is also impacted to a great extent (Huang et al., 2021). Given these problems arising in Chinese texts, CSC is challenging and important

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. U21B2009)

ORCID: Yinghao LI, https://orcid.org/0000-0002-9439-8544; Yang GAO, https://orcid.org/0000-0002-2422-0548

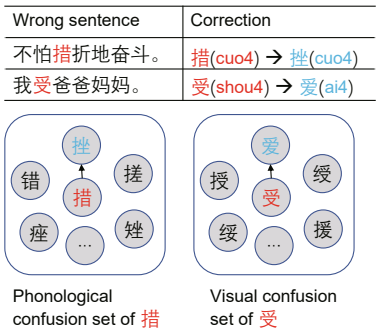

Fig. 1 Examples of phonological errors and visual errors in Chinese spelling errors. Pinyin with its tone is in brackets

for some downstream tasks such as optical character recognition (OCR) and automatic speech recognition (ASR) (Bhardwaj et al., 2022; Kim et al., 2022).

Some works developed some confusion sets containing phonologically or visually similar character pairs (Wang et al., 2018; Cheng et al., 2020; Ma et al., 2023), aiming to model the relationship between these characters. However, these confusion sets possess characters that are limited in scope and used by heuristic rules, which results in poor use of phonological and visual information. Recent works have introduced external knowledge into the pretrained language model (Guo et al., 2021; Zhang RQ et al., 2021; Liang et al., 2023) to overcome the shortcomings of the confusion set. However, due to Chinese spelling errors often involving homophones or similarly-looking characters, the appropriateness of the modal information used would vary across the different error types. Therefore, it is essential to differentiate the contributions made by different modal features to error correction. The existing approaches have not adequately modeled the semantic connections between the different modalities. They engage in merely introducing multimodal information through simple fusion operations, such as summation, thereby reducing the contribution of the different modalities.

In this paper, we propose DRMSpell, a multimodal pretrained language model for CSC. A dynamically reweighting multimodality (DRM) component is proposed for the reweighting of different modal inputs for each character in the sequence. DRM can dynamically determine which modality to rely on more according to the context. To make the model better benefit from DRM, we introduce a masking strategy for multimodal interaction. We independently mask the textual, phonological, and visual modalities of a token, so that each modality may correspond to different characters. The proposed independent-modality masking strategy (IMS) provides more examples that require multimodal interaction and thus also render the model to be more robust.

Experiments are conducted on three open benchmarks. The results demonstrate that DRMSpell fulfills the CSC task satisfactorily, outperforming all competitor models by a large margin. A further experiment on an OCR dataset shows that our method performs better than the other methods concerning OCR errors. In summary, our contributions are as follows:

1. We propose a novel pretrained language model called DRMSpell to build relationships among textual, phonological, and visual modalities, where DRM is specially designed to dynamically extract information by reweighting different modalities.

2. We design a masking strategy named IMS for multimodal interactions, which helps DRM to model the relationships among the different modalities.

3. Our method achieves state-of-the-art (SOTA) performance on three benchmark datasets and an OCR dataset.

2 Related works

With the development of self-supervised language models such as BERT (Devlin et al., 2019), pretrained language models have achieved excellent performance on many downstream tasks (Lin et al., 2019; Yang W et al., 2019). Recent works have applied a pretrained language model to the CSC task and observed that it can achieve good results with fine-tuning alone. FASPell (Hong et al., 2019) designs a denoising autoencoder–decoder paradigm with BERT as the encoder. Soft-masked BERT (Zhang SH et al., 2020) uses an added gated recurrent unit (GRU) (Bahdanau et al., 2015) layer as a detection network to generate soft-masked embedding. These methods achieve some performance improvement mainly by modifying the internal structure of BERT. An effective n-gram masking strategy and a dot-product gating mechanism were introduced by Yang HY (2023) to enhance Chinese spell checking models. Curriculum learning has also been introduced for CSC, leveraging contextual information to improve performance and highlighting the

importance of contextual similarity (Zhang D et al., 2023). A confusion set guided decision network for spoken CSC was introduced by Ma et al. (2023) to address the limitations of n-gram models and neural networks in handling uneven distributions of correct and incorrect characters.

Almost all recent works leverage the phonological and visual information of Chinese characters in different ways. SpellGCN (Cheng et al., 2020) uses the graph convolution network (GCN) (Kipf and Welling, 2017) to classify the output vectors of BERT. SpellGCN outperformed the earlier works significantly by incorporating phonological and visual similarities into BERT, which demonstrates the importance of external knowledge. ReaLiSe (Xu et al., 2021) introduces a phonetic encoder with Transformers and a graphic encoder with ResNet (He et al., 2016) to encode pinyin and graphic information. The representation vectors of different modalities are processed by a selective modality fusion module and output layer. Similar to ReaLiSe, PHMOSpell (Huang et al., 2021) encodes three modalities with respective encoders. Differently, PHMOSpell leverages the audio information by Tacotron2 (Shen et al., 2018) additionally. These methods fuse the extra modality information to text embedding during the fine-tuning stage, although the phonological encoder and the morphological encoder require to be pretrained.

Furthermore, some methods integrate multimodal information into the pretraining stage. MLM-phonetics (Zhang RQ et al., 2021) replaces words with phonetic features and corresponding sound-alike words during the pretraining stage and proposes a detection network by a pretrained encoder. PLOME (Liu et al., 2021) not only integrates phonological information, but also employs phonic and shape GRU networks to generate phonic embedding and shape embedding, respectively. These embeddings are used alongside word embedding as input for BERT. While PLOME encodes visual information using the strokes of Chinese characters, ChineseBert learns glyph information in the form of an image, although the CSC task results have not been reported. ECOPO (Li YH et al., 2022) is a training framework that narrows the gap between pretrained language models and CSC, offering the potential for further enhancements and application to other tasks. ECSpell (Lv et al., 2023) leverages an error-consistent masking strategy during pretraining to generate data that better align with real-world input scenarios. DORM (Liang et al., 2023) disentangles textual and phonetic features, enabling direct interaction between the characters and pinyin, resulting in improved performance compared to the baselines.

Multimodal information is widely used in different aspects of models such as ChineseBert (Sun et al., 2021), whose input embeddings are directly concatenated and fed into a fusion layer. However, the internal relationships among the different modalities have not been researched sufficiently. To study multimodality interactions, we propose DRMSpell for CSC.

3 Proposed pretraining framework

We introduce the framework of DRMSpell as illustrated in Fig. 2, which follows the pretraining and fine-tuning paradigm. We first introduce the proposed masking strategy IMS for multimodal pretraining, which involves varied noises for each modality to make the model more robust. Then, we describe the structure of DRMSpell, of which the DRM module can reweight the input modalities by computing the modality-side attention. Finally, we illustrate the fine-tuning procedure consisting of training and testing.

3.1 Independent-modality masking strategy

The masking strategies in ChineseBert and PLOME mask all modal information of a token, meaning that once a character is chosen to be masked, all its original modal information is removed and represented by a special token. However, these masking strategies assume that different modalities represent the same characters, which may be rather different for the multimodal inputs in some scenarios.

To handle this situation, the proposed IMS does not always mask all the modalities of a character with the same masking token, but masks different modalities independently. In other words, the three modalities chosen to be masked employ the same replacement strategy independently, and different modalities may represent different characters. The masking process is actually equivalent to replacing the masked tokens with noises, and these noises will be reconstructed into the original tokens in the output layer. In this process, IMS introduces highly varied

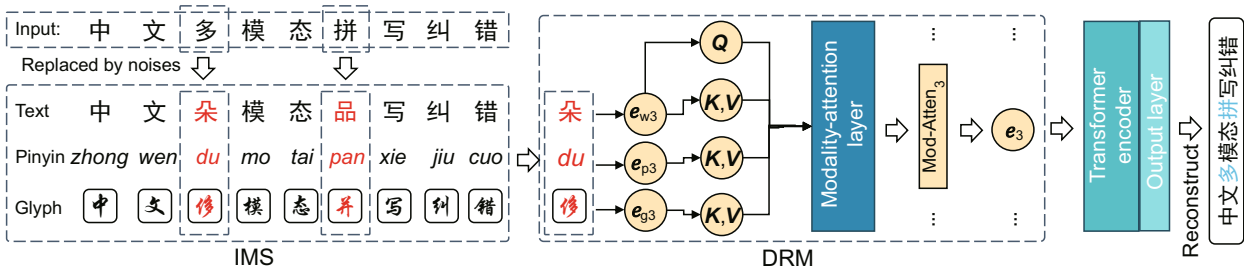

Fig. 2 Pretraining procedure of the proposed DRMSpell. Left: the input sentence "中文多模态拼写纠错" is transformed to text, pinyin, and glyph modalities by the input layer. Characters "多" and "拼" are chosen to be masked, of which the modalities are replaced independently by the independent-modality masking strategy (IMS). Middle: taking the character "多" as an example, the dynamically reweighting multimodality (DRM) module dynamically reweights the three modalities. Right: the Transformer encoder and output layer process the final input embedding to predict the correct sentence. The red font denotes masking tokens and the blue font denotes reconstructed tokens. References to color refer to the online version of this figure

noises for a masked token, which can make the model greatly robust.

Specifically, we randomly mask 15% of the tokens in the input sentence and use a dynamic masking strategy to avoid generating the same masked sequence as BERT. While using the same masking probability as in PLOME, we independently apply IMS to each of the three modalities. The detailed masking probabilities are as follows: 15% of the characters are kept unchanged, 60% are replaced by a character with a similar phonology, 15% are replaced by a character with a similar glyph, and 10% are replaced by a random character from the vocabulary. The replacement operation is applied to the three modalities text, pinyin, and glyph, separately. This allows for generating a diverse set of masked sequences, which improves the model's robustness to different forms of input variations. The difference between IMS and the confusion set based masking strategy (CMS) for PLOME is shown in Table 1.

3.2 Input layer

The input layer processes each token in the input sequence $X = \{x_1, x_2, ..., x_n\}$ to obtain the modality representation consisting of word embedding, glyph embedding, and pinyin embedding, the sum of which is the final input embedding.

For each token in the sequence X, word embedding $\boldsymbol{E}_{\mathrm{w}}$ with a dimension of d_k is obtained from an embedding table by looking up the operation the same as BERT. We follow ChineseBert (Sun et al., 2021) to obtain glyph embedding $\boldsymbol{E}_{\mathrm{g}}$ and pinyin embedding $\boldsymbol{E}_{\mathrm{p}}$, both of which have the same shape as word embedding. Glyph embedding is obtained from a 24×24 image with three types of Chinese fonts by convolutional neural networks (CNNs) and a fully connected layer. Pinyin embedding is obtained from the pinyin sequence of a Chinese character by a CNN model. The input layer is initialized with the parameters of ChineseBert. Refer to Sun et al. (2021) for details.

3.3 Dynamically reweighting multimodality

After obtaining the different modality representation vectors of a token, the proposed DRM module handles the vectors to a reweighted embedding e as the final input embedding, as shown in the middle part of Fig. 2.

Before introducing DRM, we make a brief introduction to the self-attention mechanism (Vaswani et al., 2017), which computes attention only once among all the characters in a sequence.

Table 1 Comparison between CMS and IMS for the sentence "我很(hen)喜欢打棒球"

Masking strategy	Masking token	Textual modality	Phonological modality	Visual modality
CMS	狠(hen)	狠	hen	狠
IMS	狠(hen) 跟(gen) 很(hen)	狠	gen	很

CMS: confusion set based masking strategy; IMS: independent-modality masking strategy. CMS denotes the CMS in PLOME. The character "很(hen)" is chosen to be masked. While CMS selects "狠(hen)" as the masking token for the three modalities, IMS selects the three characters independently for the three modalities

The formulation of self-attention on an n-length sequence with only word embedding $\boldsymbol{E}_{\mathrm{w}} = (\boldsymbol{e}_{\mathrm{w}_1}^{\mathrm{T}}; \boldsymbol{e}_{\mathrm{w}_2}^{\mathrm{T}}; ...; \boldsymbol{e}_{\mathrm{w}_n}^{\mathrm{T}}) \in \mathbb{R}^{n\times d_k}$ is defined as

$$\text{Self-Atten}_X = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V}, \tag{1}$$

$$\begin{cases} \boldsymbol{Q} = \boldsymbol{E}_{\mathrm{w}}\boldsymbol{W}_{\boldsymbol{Q}} \in \mathbb{R}^{n\times d_k}, \\ \boldsymbol{K} = \boldsymbol{E}_{\mathrm{w}}\boldsymbol{W}_{\boldsymbol{K}} \in \mathbb{R}^{n\times d_k}, \\ \boldsymbol{V} = \boldsymbol{E}_{\mathrm{w}}\boldsymbol{W}_{\boldsymbol{V}} \in \mathbb{R}^{n\times d_k}, \end{cases} \tag{2}$$

where $\{\boldsymbol{W}_{\boldsymbol{Q}}, \boldsymbol{W}_{\boldsymbol{K}}, \boldsymbol{W}_{\boldsymbol{V}}\} \in \mathbb{R}^{d_k\times d_k}$ are weight matrices and d_k is the dimension of $\boldsymbol{E}_{\mathrm{w}}$.

Differently, DRM computes the attention called modality-attention (Mod-Atten) among the three modalities for each character in X. Mod-Atten will be applied n times in a sequence with a length of n. Mod-Atten of the n-length sequence X with input embeddings $\{\boldsymbol{E}_{\mathrm{w}}; \boldsymbol{E}_{\mathrm{g}}; \boldsymbol{E}_{\mathrm{p}}\}$ is defined as

$$\begin{aligned} \text{Mod-Atten}_X = (&\text{Mod-Atten}_1; \\ &\text{Mod-Atten}_2; ...; \text{Mod-Atten}_n), \end{aligned} \tag{3}$$

$$\text{Mod-Atten}_i = \text{softmax}\left(\frac{\boldsymbol{Q}_i^{\mathrm{m}}\left(\boldsymbol{K}_i^{\mathrm{m}}\right)^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V}_i^{\mathrm{m}}, \tag{4}$$

$$\begin{cases} \boldsymbol{Q}_i^{\mathrm{m}} = \boldsymbol{E}_i^{\mathrm{m}}\boldsymbol{W}_{\boldsymbol{Q}_i}^{\mathrm{m}} \in \mathbb{R}^{3\times d_k}, \\ \boldsymbol{K}_i^{\mathrm{m}} = \boldsymbol{E}_i^{\mathrm{m}}\boldsymbol{W}_{\boldsymbol{K}_i}^{\mathrm{m}} \in \mathbb{R}^{3\times d_k}, \\ \boldsymbol{V}_i^{\mathrm{m}} = \boldsymbol{E}_i^{\mathrm{m}}\boldsymbol{W}_{\boldsymbol{V}_i}^{\mathrm{m}} \in \mathbb{R}^{3\times d_k}, \end{cases} \tag{5}$$

$$\boldsymbol{E}_i^{\mathrm{m}} = (\boldsymbol{e}_{\mathrm{w}_i}^{\mathrm{T}}; \boldsymbol{e}_{\mathrm{g}_i}^{\mathrm{T}}; \boldsymbol{e}_{\mathrm{p}_i}^{\mathrm{T}}) \in \mathbb{R}^{3\times d_k}, \tag{6}$$

where $i \in \{1, 2, ..., n\}$ and $\{\boldsymbol{W}_{\boldsymbol{Q}_i}^{\mathrm{m}}, \boldsymbol{W}_{\boldsymbol{K}_i}^{\mathrm{m}}, \boldsymbol{W}_{\boldsymbol{V}_i}^{\mathrm{m}}\} \in \mathbb{R}^{d_k\times d_k}$ are weight matrices.

DRM-Atten$_X$ contains all the interactive information among the three modalities. As shown in Eqs. (4)–(6), the three modality vectors are used as queries, respectively, to calculate the attention. At last, we retrieve the embedding with the textual modality as query, which is the final reweighted input embedding $\boldsymbol{E}$:

$$\begin{aligned} \boldsymbol{E} =&\text{Mod-Atten}_X(\text{text}) \\ =&(\text{Mod-Atten}_1(\text{text}); \text{Mod-Atten}_2(\text{text}); ...; \\ &\text{Mod-Atten}_n(\text{text})) \in \mathbb{R}^{n\times d_k}. \end{aligned} \tag{7}$$

The reason for selecting text modality to extract the final embedding is that while the model can process information from all three modalities simultaneously, it requires a baseline modality to compute the relationship between the three modalities and this baseline. Since the pinyin and glyph modalities are added as additional information, the text modality is chosen as the query to extract the final embedding information.

3.4 Transformer encoder

The reweighted embedding is then fed into a Transformer encoder with 12 Transformer layers, 12 self-attention heads, and a 768-dimension hidden size, which is the same as BERT$_{\text{base}}$ (Devlin et al., 2019). The output of the last layer contains rich modality information and is used as the final representation of the input token.

3.5 Output layer and learning

The output layer is a linear layer with vocabulary-size nodes for token classification. In the pretraining stage, the output layer predicts the masked tokens in the input sequence X and outputs the corresponding probability for the target sequence $Y = \{y_1, y_2, ..., y_n\}$. The output probability is defined as

$$p(\hat{y}_i|X) = \text{softmax}(\boldsymbol{W}\boldsymbol{h}_i + \boldsymbol{b}), \tag{8}$$

where v is the vocabulary size, $\boldsymbol{W} \in \mathbb{R}^{v\times d_k}$ and $\boldsymbol{b} \in \mathbb{R}^v$ are learnable parameters, and $\hat{y}_i$ is the i^{th} masked character in the input sequence.

The loss function is the maximum likelihood of the probability distribution computed as

$$\mathcal{L} = -\sum_{i=1}^{n} \ln p(\hat{y}_i = y_i|X), \tag{9}$$

where $\mathcal{L}$ denotes the learning objective.

3.6 Fine-tuning procedure

The fine-tuning procedure of DRMSpell aims to detect and correct erroneous input sentences. Given a Chinese sequence X containing n characters with errors, the CSC task aims to transform X to a correct sequence Y with the same length. In both the training and test stages, the input sequence is still processed to obtain the three modality representations by the input layer, but the masking strategy is not applied. In the fine-tuning procedure, the output layer predicts all tokens in the input sequence for correction by maximizing the conditional probability

$\sum_{i=1}^{n} p(y_i|X)$.

4 Experiments

4.1 Datasets and parameters

For the pretraining, we collect 57.3 million sentences from Weibo, Zhihu, and Wikipedia. Considering that the CSC task deals with relatively short sentences, each sentence is processed with a maximum length of 126 characters, while the model's maximum sentence length capacity is 128 characters. The parameters of the Transformer encoder are the same as those of BERT-base-Chinese (Devlin et al., 2019) with 12 Transformer layers, 12 self-attention heads, and a 768-dimension hidden size. We pretrain the model using the AdamW (Xie et al., 2020) optimizer for 6 epochs with 8 Tesla V100 graphics processing units (GPUs), a learning rate of 5e-5, and a batch size of 2048. The duration of the pretraining process is approximately 1 month.

For the fine-tuning, we use the SIGHAN training datasets with 6455 samples from SIGHAN13 (Wu et al., 2013), SIGHAN14 (Yu et al., 2014), and SIGHAN15 (Tseng et al., 2015). Following the previous works, we use the artificially generated pseudo data with 271 310 samples from Wang et al. (2018). We evaluate our model on the test datasets of SIGHAN13, SIGHAN14, and SIGHAN15. The statistics of fine-tuning datasets are shown in Table 2. Additionally, we use the OCR dataset from FASPell (Hong et al., 2019), which contains 3575 samples in the training set and 1000 samples in the test set, to validate the performance of visually similar characters. We fine-tune the model using the AdamW (Xie et al., 2020) optimizer for 10 epochs with a Tesla V100 GPU, a learning rate of 5e-5, and a batch size of 256, which takes approximately 12 h. We conduct all the experiments for 3 runs and report the averaged metrics.

Table 2 Statistics of the training and test datasets in the fine-tuning phase

Training set	Number of sentences	Avg. length	Number of errors
SIGHAN13	700	41.9	344
SIGHAN14	3423	49.2	5114
SIGHAN15	2332	30.9	3026
Wang et al. (2018)'s	271 310	42.3	381 920
OCR	3575	43.6	4850
Total	281 340	207.9	395 254

Test set	Number of sentences	Avg. length	Number of errors
SIGHAN13	1000	74.2	1221
SIGHAN14	1062	50.0	770
SIGHAN15	1100	30.3	701
OCR	1000	45.7	754
Total	4162	200.2	3446

4.2 Evaluation metrics

Following the previous works (Liu et al., 2021; Xu et al., 2021), the evaluation metrics include accuracy, precision, recall, and F1 score. For the detection and correction subtasks, the results are reported not only at the character level, but also at the sentence level, which is stricter than at the character level. For the detection subtask, it should be noted that the detected position is obtained from the correction position of the model, no matter whether the correction is appropriate or not. The evaluation script (https://github.com/liushulinle/PLOME) is used to compute these metrics. We also provide the sentence-level metrics evaluated by the official evaluation tool (https://github.com/NYCU-NLP/SIGHAN-CSC), which gives a false positive rate (FPR) additionally.

4.3 Baseline models

We compare DRMSpell with the following SOTA methods: PLOME (Liu et al., 2021) is a pretrained masked language model using CMS to learn semantics and misspelled knowledge, and it incorporates the pinyin and stroke information of a Chinese character in word embedding. MLM-phonetics (Zhang RQ et al., 2021) is an end-to-end framework incorporating phonetic features into language representation and uses a pretrained encoder to detect the wrong character. ReaLiSe (Xu et al., 2021) uses different sizes of Transformers to encode the semantic, phonetic, and graphic information of a Chinese character and predict the final correct characters by a selective modality fusion module. MDCSpell (Zhu et al., 2022) combines visual and phonological features of misspelled characters while minimizing their impact on the context. CoSPA (Yang SJ and Yu, 2022) uses an alterable copy mechanism to prevent over-correction and an attention mechanism to incorporate phonetic and visual features. ECOPO (Li YH et al., 2022) presents

a straightforward training framework that strives to achieve error-driven optimization. NM (Yang HY, 2023) is a masking strategy based on n-grams, designed to address the challenges of label leakage and error disturbance in training models. CL (Zhang D et al., 2023) leverages contextual information for enhanced performance, through training of the existing CSC models from easy to difficult. ECSpell (Lv et al., 2023) is a competitive general speller that uses an error-consistent masking strategy for pretraining. DORM (Liang et al., 2023) disentangles textual and phonetic features, enabling direct interaction between them through a phonetics-aware input. BERT* is a baseline model implemented by using the BERT-base-Chinese (Devlin et al., 2019) model. ChineseBert* represents a refined version, achieved through the fine-tuning of the ChineseBert (Sun et al., 2021) model.

4.4 Main results

Table 3 shows the evaluation scores of the proposed method and the baseline models on SIGHAN test datasets at both the character level and the sentence level. Most methods provide the precision, recall, and F1 score on three SIGHAN test sets at both the detection level and correction level. We also report the results of SIGHAN15 evaluated by the official tool at the sentence level as shown in Table 4. The proposed model again achieves the best scores on most metrics. Compared to the SOTA model DORM, DRMSpell uses the multimodalities of a Chinese character more directly and achieves an improvement of 1 percentage point (pp) for the F1 score at the sentence-correction level on SIGHAN15, indicating the effectiveness of DRMSpell.

Table 5 shows the performance of models on the OCR dataset. FASPell is a classical model yet without any fine-tuning on the OCR dataset, obtaining the worst performance at the sentence level. For fairness, we compare and fine-tune both BERT and ChineseBert on the OCR dataset. DRMSpell improves the correction F1 score by 5.3 pp and 2.4 pp at the sentence level when compared to BERT* and ChineseBert*, respectively.

5 Analysis

5.1 Effects of multimodality interaction

Different from the fusion method of ChineseBert, which is simply concatenating the input modalities, the DRM module computes the attention not only at the sequence side but also at the modality side. As shown in Table 3, the F1 score at the sentence-correction level of DRMSpell on SIGHAN15 is improved by 5.2 pp compared with the result of ChineseBert*, demonstrating that the embedding fusion way of DRM is more effective than the way of concatenating.

5.2 Ablation study

To further analyze the results of DRMSpell, we explore the effectiveness of DRM and the performance of different masking strategies at the sentence level, as shown in Table 6.

DRMSpell achieves the best C-F1 score of 80.6% by using IMS and DRM. In the absence of DRM, we use only IMS to pretrain and fine-tune DRMSpell. The C-F1 score of IMS is 77.6% and is reduced by 3 pp, which indicates the effectiveness of DRM. Then we pretrain and fine-tune DRMSpell with both DRM and CMS to explore the effectiveness of CMS. The C-F1 score of DRM + CMS is reduced by 2.6 pp compared with DRM + IMS, indicating the effectiveness of IMS. We also conduct experiments with only CMS, the result of which is 77.3%, reduced by 0.3 pp compared with only IMS. The C-F1 score of CMS is reduced by 3.3 pp compared with DRM + IMS, which is notable, highlighting that DRM and IMS have significant effects on the performance of the model.

5.3 Comparison of model parameter size

The number of parameters of pretrained language models is typically quite large. Scaling parameters always have a significant impact on both the model's performance and the inference delay of downstream tasks. In Table 7, we summarize parameter sizes of the core modules in the overall CSC models. Without multimodal information, the parameter size of BERT is significantly smaller than those of other models. Equipped with fewer parameters, DRMSpell still outperforms PHMOSpell and RealLiSe, whose parameter sizes are almost 0.8 and

Table 3 Performance of our method, SOTA methods, and baseline models on SIGHAN13, SIGHAN14, and SIGHAN15 test datasets (%)

Model	Character level						Sentence level							
	D-P	D-R	D-F1	C-P	C-R	C-F1	D-Acc	D-P	D-R	D-F1	C-Acc	C-P	C-R	C-F1
PLOME (Liu et al., 2021)	85.0	89.3	87.1	**98.7**	89.1	93.7	–	–	–	–	–	–	–	–
MLM-phonetics (Zhang RQ et al., 2021)	–	–	–	–	–	–	–	82.0	78.3	80.1	–	79.5	77.0	78.2
ReaLiSe (Xu et al., 2021)	–	–	–	–	–	–	82.7	88.6	82.5	85.4	81.4	87.2	81.2	84.1
ECOPO (Li YH et al., 2022)	–	–	–	–	–	–	**83.3**	89.3	83.2	86.2	**82.1**	88.5	82.0	**85.1**
NM (Yang HY, 2023)	–	–	–	–	–	–	–	88.3	82.8	85.4	–	86.6	81.3	83.9
CL (Zhang D et al., 2023)	–	–	–	–	–	–	76.3	**99.3**	75.7	**85.9**	75.8	**99.2**	73.8	84.6
DORM (Liang et al., 2023)	–	–	–	–	–	–	–	87.9	**83.7**	85.8	–	86.8	**82.7**	84.7
BERT*	79.0	88.4	83.5	98.0	86.7	92.0	70.8	72.1	70.6	71.3	69.0	70.3	68.8	69.6
ChineseBert*	81.3	92.7	86.6	98.0	90.8	94.2	70.3	72.9	70.1	71.5	69.1	71.6	68.9	70.2
DRMSpell	**90.1**	**94.5**	**92.2**	98.3	**92.9**	**95.5**	82.8	86.0	82.6	84.2	81.7	84.8	81.5	83.1

Model	Character level						Sentence level							
	D-P	D-R	D-F1	C-P	C-R	C-F1	D-Acc	D-P	D-R	D-F1	C-Acc	C-P	C-R	C-F1
PLOME (Liu et al., 2021)	88.5	79.8	83.9	**98.8**	78.8	87.7	–	–	–	–	–	–	–	–
MLM-phonetics (Zhang RQ et al., 2021)	–	–	–	–	–	–	–	66.2	73.8	69.8	–	64.2	73.8	68.7
ReaLiSe (Xu et al., 2021)	–	–	–	–	–	–	**78.4**	67.8	71.5	69.6	77.7	66.3	70.0	68.1
ECOPO (Li YH et al., 2022)	–	–	–	–	–	–	**78.4**	68.8	72.1	70.4	**78.5**	67.5	71.0	69.2
NM (Yang HY, 2023)	–	–	–	–	–	–	–	66.1	69.8	67.9	–	64.1	67.7	65.9
CL (Zhang D et al., 2023)	–	–	–	–	–	–	76.1	**79.7**	62.4	70.0	75.0	**79.0**	61.4	69.1
DORM (Liang et al., 2023)	–	–	–	–	–	–	–	69.5	73.1	**71.2**	–	68.4	71.9	**70.1**
BERT*	88.7	74.8	81.2	95.9	71.8	82.1	66.1	61.4	64.8	63.1	64.9	58.9	62.1	60.5
ChineseBert*	**89.8**	80.3	84.8	96.6	77.6	86.0	71.2	64.7	68.7	66.6	69.3	62.1	66.0	64.0
DRMSpell	89.1	**88.5**	**88.8**	97.9	**86.7**	**92.0**	**78.4**	66.1	**75.8**	70.6	77.8	64.9	**74.4**	69.4

Model	Character level						Sentence level							
	D-P	D-R	D-F1	C-P	C-R	C-F1	D-Acc	D-P	D-R	D-F1	C-Acc	C-P	C-R	C-F1
PLOME (Liu et al., 2021)	94.5	87.4	90.8	97.2	84.3	90.3	–	77.4	81.5	79.4	–	75.3	79.3	77.2
MLM-phonetics (Zhang RQ et al., 2021)	–	–	–	–	–	–	–	77.5	83.1	80.2	–	74.9	80.2	77.5
ReaLiSe (Xu et al., 2021)	–	–	–	–	–	–	84.7	77.3	81.3	79.3	84.0	75.9	79.9	77.8
MDCSpell (Zhu et al., 2022)	–	–	–	–	–	–	–	80.8	80.6	80.7	–	78.4	78.2	78.3
CoSPA (Yang SJ and Yu, 2022)	**95.9**	88.6	92.1	**98.5**	85.3	91.4	–	79.0	82.4	80.7	–	76.7	80.0	78.3
ECOPO (Li YH et al., 2022)	–	–	–	–	–	–	85.0	77.5	82.6	80.0	84.2	76.1	81.2	78.5
NM (Yang HY, 2023)	–	–	–	–	–	–	–	78.3	82.1	80.1	–	77.3	81.0	79.1
CL (Zhang D et al., 2023)	–	–	–	–	–	–	80.9	**85.8**	75.4	80.3	79.3	**84.7**	73.0	78.4
ECSpell (Lv et al., 2023)	–	–	–	–	–	–	86.9	82.2	80.2	81.2	86.1	80.5	78.6	79.5
DORM (Liang et al., 2023)	–	–	–	–	–	–	–	77.9	**84.3**	81.0	–	76.6	**82.8**	79.6
BERT*	92.5	85.3	88.8	96.0	81.9	88.4	83.5	74.3	79.2	76.7	82.1	71.5	76.3	73.8
ChineseBert*	93.2	88.2	90.6	95.6	84.2	89.6	84.1	75.0	81.3	78.0	83.5	72.4	78.6	75.4
DRMSpell	93.9	**90.5**	**92.2**	97.3	**88.0**	**92.4**	**87.2**	79.0	83.7	**81.3**	**86.6**	79.3	81.9	**80.6**

D: detection; C: correction; P: precision; R: recall; F1: F1 score; Acc: Accuracy. DRM: dynamically reweighting multimodality. Bold font denotes the best result. "–" denotes that relevant data are not available from the reference. Test datasets for the upper, middle, and lower tables are SIGHAN13, SIGHAN14, and SIGHAN15, respectively

Table 4 Performance of models on SIGHAN15, where the results are evaluated by the official tool at the sentence level (%)

Model	FPR	D-Acc	D-P	D-R	D-F1	C-Acc	C-P	C-R	C-F1
PHMOSpell (Huang et al., 2021)	–	82.6	**90.1**	72.7	80.5	80.9	**89.6**	69.2	78.1
TtT (Li PJ and Shi, 2021)	–	82.7	85.4	78.1	81.6	81.5	85.0	75.6	80.0
PLOME (Liu et al., 2021)	10.9	85.0	87.9	80.9	84.3	83.7	87.6	78.3	82.7
BERT*	12.2	82.6	86.0	77.1	81.3	81.6	85.7	75.1	80.0
ChineseBert*	**13.2**	83.2	85.3	79.5	82.3	81.9	84.9	76.9	80.7
DRMSpell	10.4	**85.8**	88.4	**81.9**	**85.0**	**84.6**	88.1	**79.3**	**83.5**

FPR: false positive rate; DRM: dynamically reweighting multimodality. D: detection; C: correction; P: precision; R: recall; F1: F1 score; Acc: Accuracy. "–" denotes that relevant data are not available from the reference. Bold font denotes the best result

Table 5 Overall performance of models on the OCR dataset (%)

Model	Character level						Sentence level							
	D-P	D-R	D-F1	C-P	C-R	C-F1	D-Acc	D-P	D-R	D-F1	C-Acc	C-P	C-R	C-F1
FASPell (Hong et al., 2019)	–	–	–	–	–	–	18.6	78.5	18.6	30.1	17.4	**73.4**	17.4	28.1
BERT*	92.6	84.4	88.3	75.3	63.5	68.9	77.5	84.9	76.6	80.6	58.9	63.3	57.2	60.1
ChineseBert*	**95.1**	79.9	86.8	**79.6**	63.5	70.6	74.8	**87.4**	73.9	80.1	59.7	68.8	58.1	63.0
DRMSpell	92.4	**85.7**	**88.9**	79.3	**68.0**	**73.2**	**79.5**	86.3	**78.9**	**82.4**	**63.9**	68.5	**62.6**	**65.4**

DRM: dynamically reweighting multimodality; OCR: optical character recognition. D: detection; C: correction; P: precision; R: recall; F1: F1 score; Acc: Accuracy. "–" denotes that relevant data are not available from the reference. Bold font denotes the best result

0.5 times larger, respectively. Compared with the models having similar parameters, DRMSpell performs the best.

5.4 Model robustness

This subsection analyzes the robustness of the model, including the significance of multimodality when a specific modality is missing and the impact of modality noise on the model.

As shown in Table 8, performances of the models are both dropped when any one of the modalities is absent. The performance degradation of the model with DRM + CMS is larger than that of the model with DRM + IMS. This indicates that the model with IMS is more robust when some modalities are lost. The performance worsens when pinyin and glyph are removed simultaneously, highlighting the value of multimodalities. Furthermore, the impact of a missing glyph is significantly greater than the impact of a missing pinyin, which indicates that the glyph of a Chinese character is more important than its pinyin. The reason of the results is that there are a lot of heteronyms in Chinese, which leads to semantic ambiguity. A Chinese character may have multiple pronunciations, while there is only one glyph for a character. When lacking the pinyin modality, the model can still obtain certain information from the glyph modality. However, if the glyph modality is dropped, the model will lack a definitive source of semantic information, leading to worse results.

We also explore how the model can benefit from the exact pinyin or glyph of the characters. Table 8 shows that when either the correct character's pinyin or glyph is given, the performance of the model is greatly improved. It shows that when different

Table 6 Ablation study on DRM and different masking strategies at the sentence level (%)

Setting	D-P	D-R	D-F1	C-P	C-R	C-F1
DRMSpell (DRM + IMS)	79.0	83.7	81.3	79.3	81.9	80.6
DRMSpell (IMS)	75.8	80.1	77.9	74.8	80.6	77.6
DRMSpell (DRM + CMS)	77.2	82.9	79.9	75.9	80.2	78.0
DRMSpell (CMS)	75.9	79.2	77.5	75.0	79.8	77.3

DRM: dynamically reweighting multimodality; IMS: independent-modality masking strategy. DRM + IMS means using DRM and IMS to pretrain and fine-tune DRMSpell; IMS means using IMS without DRM to pretrain and fine-tune DRMSpell; CMS denotes the confusion set based masking strategy in PLOME

Table 7 Comparison of parameter sizes for various models on SIGHAN15

Model	Number of parameters ($\times 10^6$)	Core module	C-F1 score (%)
BERT	102	Trans.	73.8
ChineseBert	147	Trans.+CNN×2	75.4
PLOME	122	Trans.+GRU×2	77.2
PHMOSpell	268	Trans.+Tacotron2+VGG19	77.5
ReaLiSe	220	Trans.×3+ResNet5+GRU	77.8
MDCSpell	204	Trans.×2	78.3
CoSPA	112	Trans.+GRU	78.3
ECOPO	102	Trans.	78.5
NM	316	Trans.×3+GRU	79.1
CL	102	Trans.	78.4
ECSpell	102	Trans.	79.5
DORM	204	Trans.×2	79.6
DRMSpell	150	Trans.+CNN×2+DRM	80.6

Trans.: Transformers; CNN: convolutional neural network; GRU: gated recurrent unit; VGG19: an architecture introduced by Simonyan and Zisserman (2014)

Table 8 C-F1 score on SIGHAN15 of different DRMSpell when lacking some modalities (%)

Setting	DRMSpell (DRM + IMS)	DRMSpell (DRM + CMS)
+text pinyin glyph	80.6	78.0
−pinyin	79.4	76.7
−glyph	77.3	74.4
−both	76.9	74.1
+golden pinyin	89.6	89.4
+golden glyph	99.6	99.2
+both	99.7	99.4

"−pinyin" means dropping the pinyin modality embedding. "+golden pinyin" means using the ground truth of pinyin as the input. Other settings are similar. CMS: confusion set based masking strategy; DRM: dynamically reweighting multimodality; IMS: independent-modality masking strategy

modalities represent different characters, the correct modality plays an important role in the process of correction. Furthermore, when the correct glyph is given, errors are almost completely corrected. It verifies that the information entropy of the glyph is much larger than that of the pinyin due to the uniqueness of representation.

Jin et al. (2022) mentioned that the introduction of additional Chinese modalities may introduce noise and analyzed the reasons behind it: (1) Inaccurate image algorithms may introduce noise into the Chinese character modality. (2) The morphological encoding of Chinese may ignore subtle differences in characters and introduce noise. Therefore, we discuss the impact of modality noise introduction on the robustness of the model.

First, this paper focuses on how to more fully use the information between different modalities to enhance model performance. Based on ChineseBert, we use CNN to capture glyph modality and do not explore more precise encoders for glyph modality. Therefore, there may indeed be some noise in the glyph information. However, compared to the benefits brought by the introduction of glyph modality information, the errors caused by this noise can be considered negligible. In subsequent work, we can specifically investigate the amount of noise introduced by different glyph encoders and how the noise affects the final performance of the model, such as by applying the more granular glyph information utilization method proposed by Jin et al. (2022) to our study. The same applies to the pinyin modality, where more detailed modal information extraction algorithms can be explored in future research.

5.5 Case study and similarity analysis

Fig. 3 demonstrates a case study of the DRM module. "热" and "然" share very similar glyph, but "热" is the erroneous character that should be corrected to "然." To mimic the potential inputs of the model, we design three different input scenarios when DRMSpell is employed. In the first scenario, the input items of text modality, pinyin modality, and glyph modality are all from the erroneous input character "热." We can observe that the glyph modality is given the most weight, which highlights its significance. The glyph modality has a greater impact in this situation to fix the mistakes since "热" and "然" are glyphically similar. In the second scenario, the pinyin modality is replaced by a random noise item, "ren." The outcomes remain correct, indicating that DRMSpell is robust enough to tolerate certain errors. In the third scenario, the pinyin modality is replaced by the ground-truth item, "ran." In this case, DRM corrects the erroneous character mainly by pinyin modality, indicating that DRM can automatically reweight different modality weights to correct the erroneous character.

In this case, although the input pinyin for the three scenarios is inconsistent, the model ultimately outputs the correct content. Then, we explore the impact of pinyin similarity on the model's correction ability by analyzing the similarities among the pinyin "re," "ren," and "ran." However, if the measurement scales or data dimensions used to compare the similarity between two groups of data are inconsistent, it can lead to variation issues (Weigang et al., 2024).

Therefore, we use three different similarity calculation methods to analyze the data from different perspectives.

We first use the national standard GB 18030-2022 (SAMR and SAC, 2022) to calculate the cosine similarities among these three pinyin instances, which are as follows: CosSim_GB(re, ren)=0.9535, CosSim_GB(re, ran)=0.4887, CosSim_GB(ren, ran)=0.5048. As we can see, only "re" and "ren" have a relatively high similarity, while the similarity results between "re" and "ran," and "ren" and "ran" are lower. Then, we use the augmentation method of pinyin numerical code (Weigang et al., 2024) to perform the calculation: CosSim_Aug(re, ren)=0.8264, CosSim_Aug(re, ran)=0.8210, CosSim_Aug(ren, ran)= 0.9804. We can observe that the similarity between "ren" and "ran" significantly increases compared to the results using the GB code, which aligns with the actual closeness of their pronunciations. Both of the above coding methods directly encode the pinyin letters. Lastly, we extract embeddings for these three pinyin from the model, all with a length of 768, to calculate their similarity: CosSim_Emb(re, ren)=0.8619, CosSim_Emb(re, ran)=0.7835, CosSim_Emb(ren, ran)=0.9527. It can be observed that the similarity results from pinyin embeddings are similar to those from the augmentation method, especially with "ren" and "ran" having the highest similarity. In the case study mentioned earlier, even though the second and third scenarios have pinyin "ren" and "ran" with high similarity, DRM can still distinguish that "ren" should contribute less weight (0.36), while "ran" should have a major contribution (0.58). This further highlights the usefulness of DRM from the perspective of pinyin similarity.

6 Conclusions

In this paper, we propose DRMSpell, a multimodal pretrained language model, for CSC. Phonological information and visual information are demonstrated to be essential to correct a mistaken Chinese character. We introduce DRM to build the internal relationships among different modalities of a Chinese character. Furthermore, the proposed IMS makes full use of the modal information in the pretraining stage. Experiments show that DRMSpell has achieved SOTA performance for the CSC task. In future work, we will explore the difference between the textual and visual representations of a Chinese character, as our experimental results show the potential representation capability of a character's glyph.

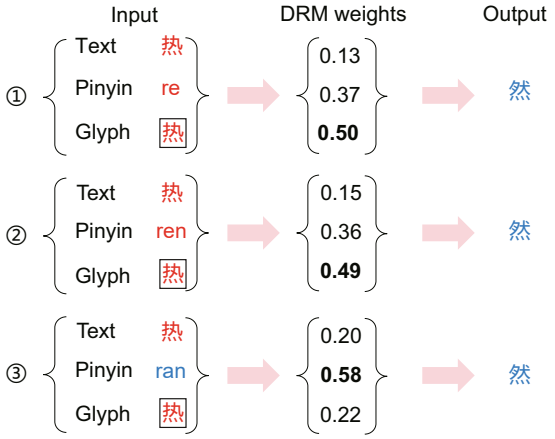

Fig. 3 A case study of the DRM module. Suppose that an input sentence is "热后我应该回到家," which includes an error character "热." We demonstrate three possible scenarios in which our model could be used to execute the correction. Bold font denotes the best DRM weights, red font denotes the erroneous cases, and blue font denotes the correct outcomes "然." References to color refer to the online version of this figure

Contributors

Yinghao LI and Baojun WANG designed the research. Yinghao LI processed the data and drafted the paper. Heyan HUANG and Yang GAO helped organize the paper. Yinghao LI, Heyan HUANG, and Yang GAO revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data are not available.

References

Bahdanau D, Cho K, Bengio Y, 2015. Neural machine translation by jointly learning to align and translate. Proc 3rd Int Conf on Learning Representations.

Bhardwaj V, Ben Othman MT, Kukreja V, et al., 2022. Automatic speech recognition (ASR) systems for children: a systematic literature review. *Appl Sci*, 12(9):4419. https://doi.org/10.3390/app12094419

Cheng XY, Xu WD, Chen KL, et al., 2020. SpellGCN: incorporating phonological and visual similarities into language models for Chinese spelling check. Proc 58th Annual Meeting of the Association for Computational Linguistics, p.871-881. https://doi.org/10.18653/v1/2020.acl-main.81

Devlin J, Chang MW, Lee K, et al., 2019. BERT: pre-training of deep bidirectional Transformers for language understanding. Proc Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, p.4171-4186. https://doi.org/10.18653/v1/N19-1423

Guo Z, Ni Y, Wang KQ, et al., 2021. Global attention decoder for Chinese spelling error correction. Proc Findings of the Association for Computational Linguistics, p.1419-1428. https://doi.org/10.18653/v1/2021.findings-acl.122

He KM, Zhang XY, Ren SQ, et al., 2016. Deep residual learning for image recognition. Proc IEEE Conf on Computer Vision and Pattern Recognition, p.770-778. https://doi.org/10.1109/CVPR.2016.90

Hong YZ, Yu XG, He N, et al., 2019. FASPell: a fast, adaptable, simple, powerful Chinese spell checker based on DAE-decoder paradigm. Proc 5th Workshop on Noisy User-Generated Text, p.160-169. https://doi.org/10.18653/V1/D19-5522

Huang L, Li JJ, Jiang WW, et al., 2021. PHMOSpell: phonological and morphological knowledge guided Chinese spelling check. Proc 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing, p.5958-5967. https://doi.org/10.18653/v1/2021.acl-long.464

Jin H, Zhang ZB, Yuan PP, 2022. Improving Chinese word representation using four corners features. *IEEE Trans Big Data*, 8(4):982-993. https://doi.org/10.1109/TBDATA.2021.3106582

Kim G, Hong T, Yim M, et al., 2022. OCR-free document understanding Transformer. Proc 17th European Conf on Computer Vision, p.498-517. https://doi.org/10.1007/978-3-031-19815-1_29

Kipf TN, Welling M, 2017. Semi-supervised classification with graph convolutional networks. Proc 5th Int Conf on Learning Representations.

Li PJ, Shi SM, 2021. Tail-to-tail non-autoregressive sequence prediction for Chinese grammatical error correction. Proc 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing, p.4973-4984. https://doi.org/10.18653/v1/2021.acl-long.385

Li YH, Zhou QY, Li YN, et al., 2022. The past mistake is the future wisdom: error-driven contrastive probability optimization for Chinese spell checking. Proc Findings of the Association for Computational Linguistics, p.3202-3213. https://doi.org/10.18653/v1/2022.findings-acl.252

Liang ZH, Quan XJ, Wang QF, 2023. Disentangled phonetic representation for Chinese spelling correction. Proc 61st Annual Meeting of the Association for Computational Linguistics, p.13509-13521. https://doi.org/10.18653/v1/2023.acl-long.755

Lin C, Miller T, Dligach D, et al., 2019. A BERT-based universal model for both within- and cross-sentence clinical temporal relation extraction. Proc 2nd Clinical Natural Language Processing Workshop, p.65-71. https://doi.org/10.18653/v1/W19-1908

Liu SL, Yang T, Yue TC, et al., 2021. PLOME: pre-training with misspelled knowledge for Chinese spelling correction. Proc 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing, p.2991-3000. https://doi.org/10.18653/v1/2021.acl-long.233

Lv Q, Cao ZQ, Geng L, et al., 2023. General and domain-adaptive Chinese spelling check with error-consistent pretraining. *ACM Trans Asian Low-Resour Lang Inform Process*, 22(5):124. https://doi.org/10.1145/3564271

Ma CS, Hu M, Peng JJ, et al., 2023. Improving Chinese spell checking with bidirectional LSTMs and confusionset-based decision network. *Neur Comput Appl*, 35(21):15679-15692. https://doi.org/10.1007/s00521-023-08570-5

Shen J, Pang RM, Weiss RJ, et al., 2018. Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions. Proc IEEE Int Conf on Acoustics, Speech and Signal Processing, p.4779-4783. https://doi.org/10.1109/ICASSP.2018.8461368

Simonyan K, Zisserman A, 2014. Very deep convolutional networks for large-scale image recognition. https://doi.org/10.48550/arXiv.1409.1556

State Administration for Market Regulation (SAMR), Standardization Administration of the People's Republic of China (SAC), 2022. Information Technology - Chinese Coded Character Set, GB 18030-2022. National Standards of People's Republic of China (in Chinese).

Sun ZJ, Li XY, Sun XF, et al., 2021. ChineseBERT: Chinese pretraining enhanced by glyph and pinyin information. Proc 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing, p.2065-2075. https://doi.org/10.18653/v1/2021.acl-long.161

Tseng YH, Lee LH, Chang LP, et al., 2015. Introduction to SIGHAN 2015 Bake-off for Chinese spelling check. Proc 8th SIGHAN Workshop on Chinese Language Processing, p.32-37. https://doi.org/10.18653/v1/W15-3106

Vaswani A, Shazeer N, Parmar N, et al., 2017. Attention is all you need. Proc 31st Int Conf on Neural Information Processing Systems, p.6000-6010.

Wang DM, Song Y, Li J, et al., 2018. A hybrid approach to automatic corpus generation for Chinese spelling check. Proc Conf on Empirical Methods in Natural Language Processing, p.2517-2527. https://doi.org/10.18653/v1/D18-1273

Weigang L, Marinho MC, Li DL, et al., 2024. Six-writings multimodal processing with pictophonetic coding to enhance Chinese language models. *Front Inform Technol Electron Eng*, 25(1):84-105. https://doi.org/10.1631/FITEE.2300384

Wu SH, Liu CL, Lee LH, 2013. Chinese spelling check evaluation at SIGHAN Bake-off 2013. Proc 7th SIGHAN Workshop on Chinese Language Processing, p.35-42.

Xie ZK, Sato I, Sugiyama M, 2020. Stable weight decay regularization. https://arxiv.org/abs/2011.11152v2

Xu HD, Li ZL, Zhou QY, et al., 2021. Read, listen, and see: leveraging multimodal information helps Chinese spell checking. Proc Findings of the Association for Computational Linguistics, p.716-728. https://doi.org/10.18653/v1/2021.findings-acl.64

Yang HY, 2023. Block the label and noise: an n-gram masked speller for Chinese spell checking. https://arxiv.org/abs/2305.03314

Yang SJ, Yu L, 2022. CoSPA: an improved masked language model with copy mechanism for Chinese spelling correction. Proc 38$^{\text{th}}$ Conf on Uncertainty in Artificial Intelligence, p.2225-2234.

Yang W, Xie YQ, Lin A, et al., 2019. End-to-end open-domain question answering with BERTserini. Proc Conf of the North American Chapter of the Association for Computational Linguistics, p.72-77. https://doi.org/10.18653/v1/N19-4013

Yu LC, Lee LH, Tseng YH, et al., 2014. Overview of SIGHAN 2014 bake-off for Chinese spelling check. Proc 3$^{\text{rd}}$ CIPS-SIGHAN Joint Conf on Chinese Language Processing, p.126-132. https://doi.org/10.3115/v1/W14-6820

Zhang D, Li YH, Zhou QY, et al., 2023. Contextual similarity is more valuable than character similarity: an empirical study for Chinese spell checking. Proc IEEE Int Conf on Acoustics, Speech and Signal Processing, p.1-5. https://doi.org/10.1109/ICASSP49357.2023.10095675

Zhang RQ, Pang C, Zhang CQ, et al., 2021. Correcting Chinese spelling errors with phonetic pre-training. Proc Findings of the Association for Computational Linguistics, p.2250-2261. https://doi.org/10.18653/v1/2021.findings-acl.198

Zhang SH, Huang HR, Liu JC, et al., 2020. Spelling error correction with soft-masked BERT. Proc 58$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics, p.882-890. https://doi.org/10.18653/v1/2020.acl-main.82

Zhu CX, Ying ZQ, Zhang BY, et al., 2022. MDCSpell: a multi-task detector-corrector framework for Chinese spelling correction. Proc Findings of the Association for Computational Linguistics, p.1244-1253. https://doi.org/10.18653/v1/2022.findings-acl.98