INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 21, Issue: 3, Month: June, Year: 2026
Article Number: 7213, https://doi.org/10.15837/ijccc.2026.3.7213

Multi-granularity Curriculum Learning for Chinese Spelling Correction in Legal Texts

Shijin Zhou, Yabo Liu

Shijin Zhou
School of Electronic Information Engineering
Hebei University of Technology
300401, Tianjin
202321902001@stu.hebut.edu.cn

Yabo Liu*
School of Electronic Information Engineering
Hebei University of Technology
300401, Tianjin
*Corresponding author: 202111901015@stu.hebut.edu.cn

Abstract

Chinese spelling correction (CSC) in legal texts presents unique challenges due to specialized terminology, complex error patterns, and the high accuracy requirements of legal documentation. To address these issues, we propose Multi-granularity Curriculum Learning (mgCL), a novel two-level adaptive training framework integrating batch-level and instance-level curricula. At the batch-level, mgCL dynamically prioritizes training samples based on cross-entropy-derived difficulty, ensuring the model is exposed to increasingly complex examples as its competence evolves. At the instance-level, it leverages Monte Carlo Dropout to quantify prediction uncertainty and adopts a weighted cross-entropy loss function with both sentence level weights and token level weights (weighted pinyin and glyph similarity) to guide the model in adapting its learning to individual samples, directing greater learning focus to ambiguous characters—especially domain-specific legal terms. To support research in legal-domain CSC, we also introduce CNLAW, a novel benchmark dataset featuring diverse error patterns and extensive legal terminology. Experimental results confirm mgCL's effectiveness: on CNLAW, it achieves a 98.02% F1 score (outperforming the strong Rephrasing Language Model (ReLM) baseline of 96.75%) and dramatically reduces the False Positive Rate (FPR) from 1.60% to 0.16%. Additionally, robust performance on the general-domain SIGHAN15 benchmark validates its cross-domain generalization. These findings demonstrate mgCL's value as an effective, scalable framework for specialized-domain CSC, with potential extensions to medical and financial text processing.

Keywords: Chinese Spelling Correction, Curriculum Learning, Legal Texts.

1 Introduction

The task of CSC involves automatically identifying and correcting spelling mistakes in the input Chinese sentences. With the proliferation of digital communication and user-generated content, Chinese spelling errors—such as homophone mix-ups, character omissions, and typographical mistakes—have become pervasive in social media, education, and professional writing [1, 2]. CSC research not only addresses practical needs for automated text quality control but also contributes to understanding Chinese orthography and supporting language learning for non-native speakers [3, 4].

While significant progress has been made in general-domain CSC, its application to specialized domains such as legal texts present unique challenges that have received insufficient attention. Legal documents contain domain-specific terminology (e.g., "litigation representative" vs. "legal representative"), formulaic expressions, and precise numerical/date references where errors could have serious consequences. Unlike social media or educational contexts where spelling errors may cause comprehension difficulties, mistakes in legal contracts or court documents could lead to substantive misinterpretations with real-world ramifications [5, 6]. Furthermore, the formal register and structural complexity of legal texts render many general-domain CSC approaches ineffective, as they fail to account for the specialized vocabulary of Chinese legal language, the need for absolute precision in key legal terms, and the contextual dependencies in lengthy legal provisions. This paper addresses these gaps by developing the first dedicated CSC framework for Chinese legal texts, combining domain adaptation techniques with a novel curriculum learning approach to handle both technical terminology and complex sentence structures characteristic of legal documentation.

Methods based on pre-trained language models have long been the mainstream of the CSC task. Most researchers can build upon pre-trained language models (e.g., BERT, RoBERTa) [7, 8] to develop customized solutions for their specific needs, particularly those with limited computational resources. A small number of researchers also employ domain-adaptive distillation on large models (e.g., Qwen, GPT) [9, 10] to guide the training of smaller models. Existing methods based on Pre-Trained Language Models (PLMS) for CSC can be categorized into four categories: feature enhancement-based [11], character-level information-based [2], architecture-based [12], and rewriting language model-based [13] approaches.

Feature enhancement-based approaches refine PLMs representations by integrating convolutional operations for local orthographic patterns and attention mechanisms for long-range semantic dependencies, strengthening detection-correction of phonetic-semantic ambiguity errors. Character-level approaches expand training corpora via confusion set injection (e.g., homophone substitutions or typos) to improve generalization to unseen errors. Unlike these input-oriented methods, architecture-based models use pointer networks to reuse original characters, constraining output space and reducing over-generation. Rewriting language models adopt sequence-to-sequence architectures to regenerate corrected sentences while preserving semantics, handling complex errors needing global context. However, existing CSC approaches have a critical limitation: they neglect varying spelling error complexity across training instances. Equal treatment of such heterogeneous data harms model performance by failing to prioritize hard cases.

The core concept of Curriculum Learning [14](CL) is to enhance model training by progressively introducing samples from simpler to more complex instances, a strategy proven effective in improving CSC systems. While traditional curriculum learning methods for CSC rely on coarse batch-level difficulty scoring or global token loss weighting (as in SSCL) [15], our instance-level approach introduces Monte Carlo Dropout to quantify prediction uncertainty at specific [MASK] positions, enabling fine-grained focus on correcting error-prone characters. It adopts a weighted cross-entropy loss function with both sentence level weights and token level weights (weighted pinyin and glyph similarity) to guide the model in adapting its learning to individual samples. The local sensitivity of this approach aligns better with the unique challenges of CSC, particularly capturing subtle character-level errors that global curriculum strategies may overlook. This uncertainty-aware weighting mechanism [16] automatically prioritizes challenging cases like homophone errors and visually similar characters while demonstrating greater robustness to training noise compared to deterministic loss-based methods. The local sensitivity of our approach better captures the unique challenges of CSC, particularly for subtle character-level errors that global curriculum strategies might overlook. A multi-granularity

CL framework is proposed for CSC task, where samples are dynamically ranked by training loss and introduced from easy to hard. Evaluated on ReLM (a state-of-the-art PLM) using the CNLAW and SIGHAN15 benchmark, this approach outperforms conventional methods, demonstrating its effectiveness in enhancing CSC performance. The key innovations of this research include:

1. A Novel Multi-granularity Framework: We introduce the Multi-granularity Curriculum Learning (mgCL) framework, which features a dual-level curriculum design to handle hierarchical error complexity. At the batch-level, training samples are dynamically sorted from easy to hard according to cross-entropy loss. At the instance-level, Monte Carlo Dropout is utilized to quantify prediction uncertainty at [MASK] positions for fine-grained correction of error-prone characters. Besides, a weighted cross-entropy loss function combining sentence-level weights and token-level weights (integrating pinyin and glyph similarity) is adopted for adaptive sample learning. Benefiting from superior local sensitivity, this method is well-suited for CSC tasks and can capture subtle character-level errors ignored by conventional global curriculum strategies.
2. A New Legal-Domain Benchmark (CNLAW): We construct CNLAW, a new benchmark dataset specifically created for Chinese legal-domain spelling correction. This dataset provides a critical resource, featuring a diverse collection of authentic error patterns and specialized terminology necessary for training and rigorously evaluating models in this specific domain.
3. Significant Empirical Insights: Our extensive experiments establish that the mgCL framework achieves state-of-the-art performance on the CNLAW benchmark. Critically, it also yields a substantial reduction in the False Positive Rate, fulfilling a key requirement for reliable legal text processing. Furthermore, the framework's strong performance on the general-domain SIGHAN15 benchmark demonstrates its effective cross-domain generalization, validating its applicability beyond the specialized legal context.

2 Related Work

2.1 Chinese Spelling Correction

Research on CSC has witnessed a series of advancements over the years, which can be categorized into four key aspects presented in the introduction section.

Character-level approaches in CSC have traditionally employed confusion sets as a core technique. These sets, built upon phonetic and visual resemblances between characters, provide an efficient mechanism for detecting potential errors. Previous work [17, 18] has successfully integrated confusion sets in pretraining stages. While constrained by finite coverage and limited flexibility, this method offers computational advantages by predefining character relationships rather than calculating similarities dynamically.

For feature enhancement strategies, phonetic representation through pinyin sequences remains predominant in CSC systems. The field exhibits varied implementations, with certain models [19] processing pinyin holistically while others [17] adopt a finer-grained analysis of initials, finals, and tones for improved phonetic modeling. Tacotron2 [20] presents an alternative architecture in this domain. Visual feature extraction has similarly seen diverse approaches, ranging from direct stroke-based representations [21] to more sophisticated methods employing ideographic description sequences [22] that capture character structure through hierarchical decomposition. Contemporary research leverages deep learning architectures including ResNet and VGG19 [20, 23] for visual feature extraction, notwithstanding empirical observations that visually-induced errors constitute a minority of cases in practical applications.

The Detector-Corrector (D-C) framework represents a widely adopted architectural paradigm in CSC, operating through a sequential detection-correction pipeline. In this framework, the detection module, often implemented via Bi-GRU or efficient Transformer architectures, performs binary classification to identify erroneous characters. Subsequently, the correction module generates replacement candidates from a probability-weighted character set. While effective, conventional D-C architectures face several limitations, prompting various enhancements:

(1) **Enhanced Detection Mechanisms**: Soft-Masked BERT [24] introduces a probabilistic masking approach, where a Bi-GRU detector modulates the masking intensity based on error likelihood, enabling more nuanced error handling.

(2) **Feature Preservation Strategies**: MDCSpell [25] implements a late fusion technique that maintains typo-relevant visual and phonetic features while reducing error propagation through coordinated detection-correction hidden states.

(3) **Multi-Stage Processing**: DR-CSC [26] extends the pipeline with a dedicated reasoning phase between detection and correction, optimizing candidate generation through structured search.

(4) **Bidirectional Optimization**: Bi-DCSpell [27] establishes dynamic cross-task interaction, allowing continuous mutual refinement between detection and correction processes.

While sequence tagging models dominate current research, ReLM represents a significant paradigm shift, demonstrating superior performance across multiple benchmarks.

The transformative impact of Large Language Models (LLMs) on NLP has yet to be fully realized in CSC applications. Current research remains primarily focused on PLM-based approaches, leaving substantial opportunities for exploring LLM capabilities in spelling correction tasks. This represents an important frontier for future investigation, particularly regarding: 1) LLM adaptation strategies for CSC-specific challenges. 2) Comparative effectiveness between PLM and LLM approaches. 3) Resource-efficient implementation of LLMs for spelling correction.

2.2 Curriculum Learning

CL has proven to be an effective training paradigm across multiple domains of artificial intelligence. In computer vision research, several studies [28, 29, 30] have empirically demonstrated the benefits of CL for model optimization. In natural language processing, this approach has shown particular promise for complex tasks such as sentiment analysis [31] and response generation [32], where its progressive learning strategy helps address optimization challenges through systematic data sequencing.

The application of CL to Neural Machine Translation (NMT) was first proposed by researchers [33], who identified two key components: difficulty metrics for training samples and corresponding learning schedules. Follow-up studies [34, 35, 36, 37, 38] successfully extended these principles to specialized translation domains.

For general translation tasks, prior work [39] conducted comprehensive evaluations of various difficulty metrics, including linguistically intuitive measures such as lexical complexity and syntactic length. The results showed significant performance variations across different language pairs and model architectures. More recently, [40] Another study proposed using the cross-entropy loss of individual training samples as a direct measure of instance difficulty for curriculum learning, where higher loss values indicate more challenging samples that the model has not yet mastered. This approach enables a data-driven difficulty assessment that naturally adapts as the model evolves during training, eliminating the need for manually defined heuristic metrics.

3 Proposed Framework

This section presents our technical framework in two phases. First, we formally define the CSC task and its key characteristics. Second, we introduce our novel mgCL framework specifically designed for CSC, which incorporates dual-level optimization mechanisms operating at different granularities.

3.1 Problem Definition

The CSC task can be formally characterized as follows: For an input character sequence $X = \{x_1, x_2, \cdots, x_N\}$ containing potential errors, the system aims to produce a corrected output sequence $Y = \{y_1, y_2, \cdots, y_N\}$, where each y_i corresponds to the ground-truth correction of x_i. This transformation process is mathematically formulated as a conditional probability maximization problem $P(Y|X)$. For each character position i, if x_i is identified as erroneous, the correction probability is computed as $P(y_i|X)$, where y_i represents the correct character at position i. More specifically, the learning objective is to minimize negative log-likelihood loss:

$$\mathcal{L}(X,Y) = -\frac{1}{N}\sum_{i=1}^{N}\log P(y_i \mid x_{<i}, X; \theta) \tag{1}$$

where N represents the length of the input sequence (or can be specifically the number of actual error characters). $x_{<i}$ signifies the contextual characters preceding position i. $P(y_i \mid x_{<i}, X; \theta)$ denotes the probability that the model predicts the correct character y_i at position i leveraging the preceding context $x_{<i}$ and the input X. θ denotes the trainable parameters of our CSC model.

3.2 Batch-Level Curriculum Learning for CSC

In CSC tasks, cross-entropy loss can reliably quantify sample difficulty, based on which the dataloader arranges training batches in an easy-to-hard order for curriculum learning. Traditional difficulty metrics (e.g., word frequency, sentence length, linguistic complexity, and edit distance) are insufficient for domain-specific CSC scenarios such as legal texts. By contrast, cross-entropy loss fits the essence of spelling correction: it reflects the model's error-recognition confidence without relying on fixed annotated references, thus adapting to ambiguous legal terms. Linguistic complexity tends to misjudge professional expressions and long correct sentences as difficult, while edit distance requires unique standard references and merely measures correction steps rather than model recognition difficulty. Empirical results further verify that samples with lower cross-entropy loss achieve higher correction accuracy, demonstrating the rationality of adopting this loss for difficulty quantification.

Given a trained CSC model and a dataset with N sentence pairs as:

$$D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\} \tag{2}$$

The cross-entropy loss function is calculated to obtain the difficulty of each instance. Then, the Cumulative Density Function (CDF) is applied to transform the distribution of instance difficulties into the interval $(0, 1]$:

$$\tilde{d}\langle(x_i, y_i)\rangle \in (0,1] = \mathrm{CDF}\left(\left\{\tilde{d}\langle(x_i, y_i)\rangle\right\}_{i=1}^{N}\right)_i \tag{3}$$

Specifically, the score of the difficult instance tends to be 1, while that of the easy instance tends to be 0.

For CL in the CSC task, the next challenge is organizing all training samples into a sequential curriculum based on their difficulty scores. This process defines the complexity of samples that the CSC model can absorb within a specific training batch.

We utilize the concept of model competence—a function that takes the training batch step t as input and outputs a value ranging from 0 to 1. This function regulates the batch-level training data loading schedule.

$$C(t) = \min\left(1, \sqrt{k \cdot t \cdot \frac{1 - c_0^k}{T} + c_0^k}\right) \tag{4}$$

where c_0 denotes the initial competence at the training onset, k acts as a coefficient to control the growth rate of model competence, and T serves as a hyperparameter determining the duration of the batch-level curriculum for CSC.

3.3 Instance-Level Curriculum Learning for CSC

Although batch-level CL guides the model to learn from easy samples to hard ones via ordered dataloader scheduling, individual sentences and tokens within a single batch still exhibit heterogeneous difficulty. Accordingly, we further assign instance-specific learning weights based on fine-grained difficulty estimation. Generally, frequent tokens are easier for model prediction, while rare ones pose greater correction challenges. Longer sentences and grammatically defective texts also raise correction difficulty. To address this issue, we adopt Monte Carlo Dropout to estimate model uncertainty and dynamically adjust the loss weight for each instance.

Given a mini-batch with M sentence pairs and model parameters θ, we perform Q stochastic sampling inference. Each sample yields Q probability outputs, whose variance indicates the model's predictive confidence. Based on such uncertainty statistics, we quantitatively evaluate the complexity of sentences and tokens for subsequent weighted optimization. Specially, for each token $w_{i,j}$ in sentence $\mathbf{x}_i$ we obtain Q conditional probabilities $p_{i,j}^{(q)}$ from the Q MC-dropout forward passes, where $p_{i,j}^{(q)}$ is the probability of correcting $w_{i,j}$ to the target token $\hat{w}_{i,j}$ in the q-th pass, where $\bar{p}_{i,j} = \frac{1}{Q}\sum_{q=1}^{Q} p_{i,j}^{(q)}$. The model uncertainty for this token is quantified by the variance of these probabilities:

$$U_{i,j} = \mathrm{Var}(P_{i,j}) = \frac{1}{Q}\sum_{q=1}^{Q}(p_{i,j}^{(q)} - \bar{p}_{i,j})^2. \tag{5}$$

To incorporate objective linguistic difficulty, we define token level pinyin similarity Sim_{ij}^{pinyin} and glyph similarity Sim_{ij}^{glyph} between the original token $w_{i,j}$ and the target $\hat{w}_{i,j}$.

$$Sim_{ij}^{\text{pinyin}} = 1 - \frac{\sum_{k=1}^{K_{i,j}} \mathrm{EditDist}(p(w_{i,j}), p(\hat{w}_{i,j}))}{\sum_{k=1}^{K_{i,j}} \max\left(\mathrm{len}(p(w_{i,j})), \mathrm{len}(p(\hat{w}_{i,j}))\right)} \in (0,1) \tag{6}$$

$$Sim_{ij}^{\text{glyph}} = 1 - \frac{1}{K}\sum_{k=1}^{K_{i,j}} \frac{\|g_{i,j} - \hat{g}_{i,j}\|}{\|g_{i,j}\| + \|\hat{g}_{i,j}\|} \in (0,1) \tag{7}$$

where $K_{i,j}$ denotes the maximum number of characters between $w_{i,j}$ and $\hat{w}_{i,j}$, $p(\cdot)$ denotes the pinyin sequence of a character, $\mathrm{EditDist}(\cdot)$ denotes the edit distance between two pinyin sequences, and $g_{i,j}$ and $\hat{g}_{i,j}$ represent the glyph feature vectors of individual characters. Both similarities lie in $(0,1)$ and decrease as the phonetic or visual difference grows, indicating higher correction difficulty.

A token-level difficulty score is constructed by fusing model uncertainty with linguistic features:

$$d_{i,j} = U_{i,j} + 1 - \alpha\, Sim_{ij}^{\text{pinyin}} - (1-\alpha)\, Sim_{ij}^{\text{glyph}}, \tag{8}$$

where $U_{i,j}$ denotes the model uncertainty for token j in sentence i, Sim_{ij}^{pinyin} and Sim_{ij}^{glyph} represent the phonetic and glyph similarities respectively, and $\alpha \in (0,1]$ is a weighting coefficient balancing these two linguistic modalities.

Sentence-level difficulty D_i is then obtained by aggregating token-level difficulties, weighted by attention weights $\beta_{i,j}$ that reflect each token's importance within the sentence:

$$D_i = \sum_{j=1}^{L_i} \beta_{i,j}\, d_{i,j}, \tag{9}$$

where L_i is the length of sentence i, and the weights satisfy $\sum_{j=1}^{L_i} \beta_{i,j} = 1$.

To avoid extreme weighting differences, both token-level and sentence-level difficulties are normalized via a sigmoid function:

$$\tilde{d}_{i,j} = \frac{1}{1 + e^{-\lambda(d_{i,j} - \mu_d)}}, \tag{10}$$

$$\tilde{D}_i = \frac{1}{1 + e^{-\lambda(D_i - \mu_D)}}, \tag{11}$$

where λ controls the sensitivity of the normalization, and μ_d, μ_D are shifting constants that center the difficulty distributions.

The learning weights for individual tokens and sentences are then defined based on the normalized difficulties:

$$\omega_{i,j} = 1 + \gamma\, \tilde{d}_{i,j}, \tag{12}$$

$$\Omega_i = 1 + \gamma\, \tilde{D}_i, \tag{13}$$

where γ is a hyperparameter that adjusts the overall influence of difficulty on the learning objective.

Finally, the instance-level curriculum loss combines sentence-level and token-level weights into a weighted cross-entropy objective:

$$\mathcal{L}_\theta = -\frac{1}{M}\sum_{i=1}^{M}\Omega_i\left(\frac{1}{L_i}\sum_{j=1}^{L_i}\omega_{i,j}\log P_{i,j}\right), \tag{14}$$

where M is the total number of sentences, L_i is the length of sentence i, and $P_{i,j}$ denotes the model's predicted probability of the correct target token $\hat{w}_{i,j}$.

This formulation encourages the model to focus more on tokens and sentences that exhibit higher uncertainty and greater linguistic complexity, thereby realizing fine-grained, instance-aware curriculum learning.

3.4 Multi-granularity combination of Batch-Level and Instance-Level learning

Guided by the principles introduced in the preceding sections, we develop Algorithm 1 and present the overall framework for reproducibility. The architectural schematic is shown in Figure 1, while an operational demonstration of the initial phases is provided in Figure 2.

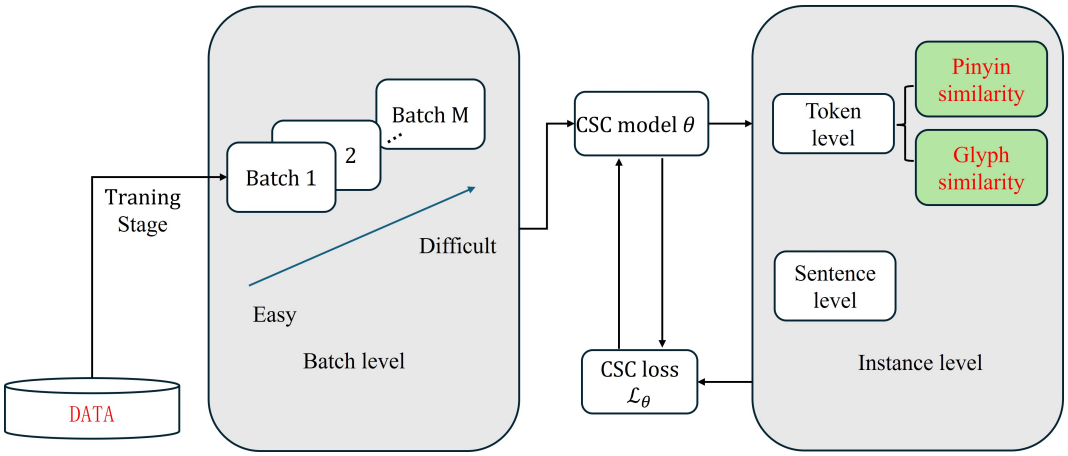

Figure 1: A comprehensive schematic of multi-granularity Curriculum Learning architecture. Data is batched from easy to difficult, fed into the CSC model θ optimized $\mathcal{L}_\theta$, and evaluated at token and sentence levels.

Algorithm 1 Multi-granularity Curriculum Learning (mgCL) Algorithm

Input: Dataset $\mathcal{D} = \{s_i\}_{i=1}^{M}$ containing M batch samples, each with i samples; CSC model trainer $\mathcal{T}$; difficulty function d; competence function c; token-level weight ω; sentence-level weight Ω; sample loss $\mathcal{L}$; initial parameters of the spelling error correction model θ.

Compute the difficulty score $d(s_i)$ for each $s_i \in \mathcal{D}$.
Compute the cumulative distribution function (CDF) of the difficulty scores, obtaining the normalized difficulty CDF score $\bar{d}(s_i) \in (0, 1]$ for each sample.
for each training step $t = 1, 2, \ldots$ **do**
    Compute the model competence $c(t)$.
    Uniformly sample a data batch B_t from all $s_i \in \mathcal{D}$ such that $\bar{d}(s_i) \leq c(t)$.
    Invoke the trainer $\mathcal{T}$ with B_t as input.
end for
Perform Monte Carlo Dropout processing.
Compute token-level pinyin and glyph similarity.
Compute token-level and sentence-level weights ω and Ω.
Update the model θ using the sample loss $\mathcal{L}$ calculated from ω and Ω.
return The optimized spelling error correction model θ.

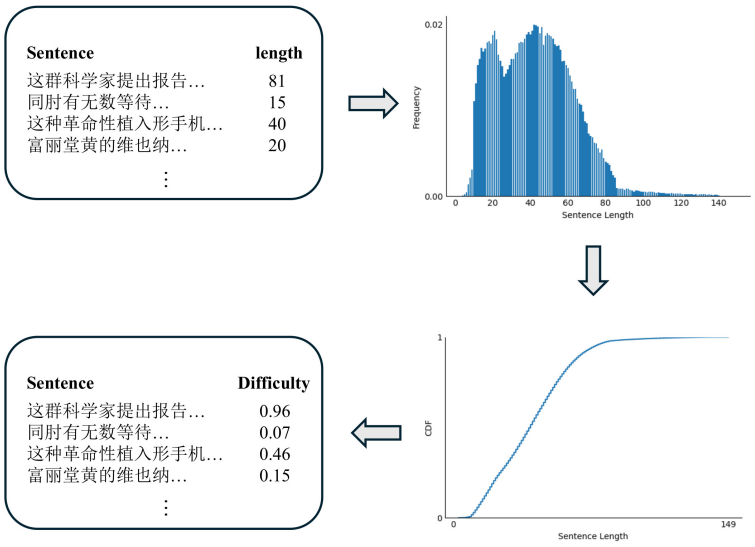

Figure 2: Example visualization of the preprocessing pipeline employed in our framework.

4 Experiments

4.1 CNLAW Dataset Construction

Raw data is collected from the Chinese Legal Network, comprising 625 sub-legal documents. Initially, we performed normalization on this raw data, removing non-functional statements and single-character words. Subsequently, we segmented the processed data into 157,244 sentences based on punctuation marks such as periods, semicolons, and question marks, while discarding samples with lengths shorter than four characters.

In the initial phase, we recruited ten volunteers with expertise in deep learning to manually introduce errors intothe original sentences. All volunteers adhered to a unified annotation protocol to guarantee consistency ofmanual error introduction, lt is worth noting that the samples used for manual annotation were not randomlycreated; they were selected from the error record database of a legal document proofreading softwareassociated with a company's project, Owing to potential impacts on the company's interests, specific details otthis database cannot be publicly shared. However, due to the substantial labor costs involved in manual errorintroduction, we later shifted to a method of generating errors using confusion sets. Unlike previous work, weenriched the confusion sets with characters that are visually similar based on the Four-Corner Encoding systemThis approach significantly diversified our dataset, making it more representative of real-world spelling errorsand enhancing the robustness of our training data.

Finally, Our rules for confusion set replacement are as follows: (1) We randomly replace 50% of the samples, unlike previous publicly available datasets, we increased the proportion of negative samples. This is because a higher number of negative samples allows for a more effective evaluation of the model's false positive rate, which refers to the rate at which the model incorrectly modifies originally correct sentences, also known as over-correction. (2) There is a 40% probability of replacing with homophonic characters (same pinyin), a 30% probability of replacing with phonetically similar characters (similar pinyin), a 20% probability of replacing with characters that are visually similar in stroke structure, and a 10% probability of replacing with a random commonly used character.

4.2 Datasets

Following previous works, we evaluate the model performance on CSC datasets, CNLAW and SIGHAN15. We use the 271K automatically generated corpus and 10K manually annotated samples from SIGHAN and CNLAW as our training data. The details of the datasets are provided in Table 1.

Table 1: Statistical data of the dataset, including the number of sentences (#Sent), average sentence length (#Avg.Length), and total number of errors (#Errors).

Data	**#Sent**	**#Avg.Length**	**#Errors**
Training Data			
271K automatically generated corpus	271,329	44.4	382,704
SIGHAN 2013	350	49.2	350
SIGHAN 2014	6,526	49.7	10,087
SIGHAN 2015	281,379	44.4	397,378
CNLAW	126,180	41.9	62,884
Test Data			
SIGHAN 2015	1,100	30.5	550
CNLAW	15,720	41.8	7,860

4.3 Evaluation Metrics

The sentence-level F1 score and FPR are reported as the evaluation metrics.

(1) In fact, precision and recall are a pair of conflicting evaluation indicators. If the model wants to correct more wrong texts, its correction performance is not qualitatively improved, it can only rely on making corrections in more text areas. The increase in the correct texts misjudged as wrong texts will affect the precision rate, and vice versa. In order to comprehensively consider the accuracy rate and recall rate, avoid being limited to a single correction index, the F1 score is used as a comprehensive index. The formula is as follows:

$$\mathrm{F1} = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{15}$$

(2) In practical deployments of CSC systems, the False Positive Rate (FPR) serves as a crucial performance metric. This indicator quantifies the system's tendency to incorrectly alter properly spelled text segments, a phenomenon commonly referred to as over-correction. The mathematical formulation for FPR calculation is presented below:

$$\mathrm{FPR} = \frac{FP}{FP + TN} \tag{16}$$

where TP (true positives): Correctly corrected error texts, FP (false positives): Erroneously modified correct texts, FN(false negatives): Uncorrected error texts, TN (true negatives): Correctly retained correct texts.

4.4 Experimental Setup

This study adopts ReLM as the backbone model, optimized using AdamW with a learning rate of 5e-5. The training configuration employs a batch size of 64 and processes sequences up to 128 tokens in length over 10 epochs. Key curriculum learning parameters include an initial competence of 0.4, a growth rate of 2, 5 Monte Carlo dropout samples, and a pinyin-glyph similarity weight α of 0.8, with all hidden layers maintaining 768-dimensional representations. Experiments were conducted on a Linux cloud server equipped with 16 Intel Xeon vCPUs (2.5GHz), 40GB RAM, and an NVIDIA RTX 4090 GPU (24GB VRAM) to accelerate training, using Python 3.7 and PyTorch 2.0 for model development and optimization.

4.5 Baseline Models

Our proposed model is evaluated against several leading approaches in CSC:

- **Soft-Masked BERT**: Implements a dual-stage framework that first identifies potential errors and subsequently applies masking before feeding the modified input into BERT for correction.
- **DCN**: Leverages a dynamically connected architecture to capture sequential dependencies between neighboring characters, enhancing contextual modeling.

- **CRASpell**: Addresses over-correction and contextual noise through a hybrid approach, incorporating a copy mechanism and dedicated noise modeling.
- **MDCSpell**: Integrates visual and phonetic features through BERT embeddings, combining detector and corrector hidden states via late fusion to mitigate error propagation.
- **ReLM**: Adopts a sentence-level rephrasing strategy with slot infilling, departing from conventional character-wise tagging methods.

4.6 Main Results

Table 2 and 3 demonstrates the performance of our CSC model integrated with the mgCL method, in comparison to baseline models. The proposed mgCL framework consistently outperforms all baselines by strategically prioritizing challenging samples during training. The mgCL framework adopted here prompts the CSC model to progressively shift its focus toward more difficult samples, ultimately boosting its performance.

Table 2: The Performance of mgCL and All Baselines on CNLAW

Dataset	Model	Prec.(%)	Rec.(%)	F1(%)	FPR(%)
CNLAW	Soft-Masked BERT	85.87	82.99	84.40	8.31
	DCN	87.11	86.45	86.78	5.27
	MDCSpell	92.88	88.51	90.64	4.73
	CRASpell	89.37	87.10	88.22	4.22
	ReLM	97.12	96.38	96.75	1.60
	ReLM (mgCL)	98.28	97.76	98.02	0.16

Table 3: The Performance of mgCL and All Baselines on SIGHAN15

Dataset	Model	Prec.(%)	Rec.(%)	F1(%)	FPR(%)
SIGHAN15	Soft-Masked BERT	64.89	70.97	68.30	19.67
	DCN	65.91	73.43	69.46	19.30
	MDCSpell	66.77	75.04	70.67	18.96
	CRASpell	69.30	76.34	72.65	15.92
	ReLM	73.01	81.67	77.10	15.00
	ReLM (mgCL)	75.21	82.04	78.48	13.03

4.7 Ablation Study

To verify the effectiveness of each module in our proposed mgCL framework for the CSC model, we conduct ablation studies with the following setups: 1) ReLM; 2) only integrating Batch-Level CL; 3) only integrating Instance-Level CL. When either Batch-Level CL or Instance-Level CL is removed, the model's performance diminishes—this highlights the efficacy of each module within our multi-granularity framework.

As shown in Table 4, every component of our framework significantly improves the CSC model's performance compared to ReLM. Batch-Level CL pushes the CSC model to gradually shift its focus toward complex samples, driving performance enhancements. Meanwhile, Instance-Level CL helps the CSC model automatically optimize in the right direction, further boosting its performance.

Table 4: Results For Ablation Studies. "ΔF1" and "ΔFPR" Indicate The Absolute F1 and FPR Improvements on CNLAW Dataset.

Model	F1	FPR	ΔF1	ΔFPR
ReLM	96.75%	1.60%	—	—
ReLM+Only Batch-Level CL	97.93%	0.19%	+1.18%	−1.41%
ReLM+Only Instance-Level CL	97.96%	0.12%	+1.21%	−1.48%
ReLM (mgCL)	98.02%	0.16%	+1.27%	−1.44%

4.8 Parameter Study

As mentioned in the preceding content, we assign k to scale the gap between high-difficulty and low-difficulty sentences in the CSC task. We set c_0 as the initial competence at the training onset. We conduct Q Monte Carlo dropout sampling operations on the CSC model. Furthermore, we introduce α as the balancing weight to control the contribution of pinyin similarity and glyph similarity in token-level difficulty calculation. In this section, to investigate the impact of different k, c_0, Q, and α values, we conduct extensive experiments by varying these parameters on the CNLAW dataset.

Table 5 shows that when the value of k reaches a certain threshold, the performance of the CSC model no longer improves. We believe that an excessively large k leads to overfitting on hard instances while ignoring easy ones. Consequently, although using ReLM as the PLM brings consistent improvements for all k values ranging from 1 to 3, selecting the optimal $k = 2$ value is still essential.

Table 5: The F1 and FPR on CNLAW, Using Different Values of k in ReLM (Batch-level CL)

Model	k **Value**	$F1$	FPR
ReLM	—	96.75%	1.60%
ReLM(Batch-level CL)	1	94.87%	2.02%
ReLM(Batch-level CL)	**2**	**97.93%**	**0.19%**
ReLM(Batch-level CL)	3	95.62%	1.77%

Regarding c_0, Table 6 shows a clear performance trend. An excessively large c_0 makes the model tackle hard samples too early, disrupting the easy-to-hard learning order, causing overfitting and poor performance on simple cases. Conversely, an overly low c_0 leads to insufficient initial capability, inefficient learning, and delayed convergence. We find $c_0 = 0.4$ is optimal, which balances initial competence and gradual learning, ensuring stable performance on both simple and hard samples. Careful tuning of c_0 is critical for effective curriculum learning in CSC tasks.

Table 6: The F1 and FPR on CNLAW, Using Different Values of c_0 in ReLM (Batch-level CL)

Model	c_0 **Value**	$F1$	FPR
ReLM	—	96.75%	1.60%
ReLM(Batch-level CL)	0.3	97.85%	0.24%
ReLM(Batch-level CL)	**0.4**	**97.93%**	**0.19%**
ReLM(Batch-level CL)	0.5	97.55%	0.22%

Turning to the analysis of Monte Carlo samples Q, as shown in Table 7, performance varies significantly with this parameter. A small Q fails to capture sufficient stochastic information, harming generalization, while an excessively large Q increases computational costs without performance gains. Experiments show that $Q = 5$ achieves the best balance: it provides enough sampling diversity to enhance model robustness while maintaining efficiency, enabling the CSC model to effectively leverage Monte Carlo dropout.

Table 7: The F1 and FPR on CNLAW, Using Different Values of Q in ReLM (Instance-level CL)

Model	Q **Value**	$F1$	FPR
ReLM	—	96.75%	1.60%
ReLM(Instance-level CL)	3	97.03%	0.22%
ReLM(Instance-level CL)	**5**	**97.96%**	**0.12%**
ReLM(Instance-level CL)	7	96.84%	1.33%
ReLM(Instance-level CL)	10	95.88%	1.91%

The difficulty score derived from pinyin and glyph features affects the upper bound of instance-level curriculum learning through the weighted cross-entropy loss function. In particular, the value of the pinyin-glyph similarity weight parameter α directly determines the model's performance.as shown in Table 8, the model achieves the best performance when α is set to 0.8.

4.9 Case Study: Legal semantic errors scenarios

This comparative case study highlights the critical role of domain-specific knowledge in CSC for legal and regulatory texts. As shown in Figure 3, our model correctly identifies "ruling" (the precise

Table 8: The F1 and FPR on CNLAW, Using Different Values of α in ReLM (Instance-level CL)

Model	α **Value**	$F1$	FPR
ReLM	—	96.75%	1.60%
ReLM(Instance-level CL)	0.7	97.55%	0.19%
ReLM(Instance-level CL)	**0.8**	**97.96%**	**0.12%**
ReLM(Instance-level CL)	0.9	97.61%	0.19%

legal term) instead of the generic "judgment", while BERT erroneously suggests "judgment/decision". In a second case involving regulatory enforcement, our model accurately selects "testing qualification" (the appropriate technical term), whereas BERT proposes the incorrect "supervision qualification". These examples confirm that general-domain models like BERT often fail to capture nuanced terminology differences in specialized fields, sometimes introducing errors via inappropriate substitutions. In contrast, our domain-adapted framework consistently outperforms by maintaining terminological precision—correctly identifying both legal adjudication terms ("ruling") and technical regulatory terms ("testing qualification")—underscoring its effectiveness in meeting the vocabulary and contextual demands of professional domains.

Figure 3: Cases of semantic errors selected from CNLAW.

4.10 Case Study: Over-Correction scenarios

We then explore over-correction cases in CSC. As shown in Figure 4. In the first example, the source legal text states: "Judges must not substitute personal emotions for legal provisions when making judgments." Its Chinese version uses a legal-specific prohibition marker ("must not"). BERT mistakenly replaces this with a homophonous instruction marker ("be sure to"), reversing the sentence's

Figure 4: Cases of Over-Correction Selected from CNLAW.

semantic polarity and turning a legal prohibition into an erroneous directive. Our model, by contrast, retains the original prohibition marker, preserving correct legal intent.

In the second example, the source legal text reads: "The company shall establish special account books in accordance with the law to record revenue and expenditure." The Chinese term for "account books" uses a legally/accounting-specific character (Form A). BERT replaces Form A with a homophonous character (Form B)—a substitution that, while acceptable in general usage, deviates from standardized legal/accounting terminology and introduces an unnecessary error. Our model retains Form A, as its domain-specific knowledge recognizes Form A as the convention in relevant legal provisions.

These cases demonstrate that general-domain CSC models like BERT lack fine-grained understanding of specialized legal terminology and semantics. They may misapply general language rules to domain-specific texts, leading to over-correction that distorts legal meaning or introduces non-standard terms. Domain-adapted models, by leveraging specialized knowledge, better preserve the accuracy and authenticity of legal texts—critical for applications requiring precise legal text processing.

4.11 Case Study: Multiple errors scenarios

This case analyzes the performance of the BERT model and our proposed model in correcting Chinese texts with multiple spelling errors, focusing on two examples.

As shown in Figure5. In the first example related to legal text, the source text contains several misspellings: incorrect terms for "trustor", "trustee", and "beneficiary". The BERT model manages to correct some errors, such as identifying the correct "beneficiary", but fails to fully rectify all mistakes, leaving residual errors in the names of roles like "trustor". In contrast, our model accurately corrects

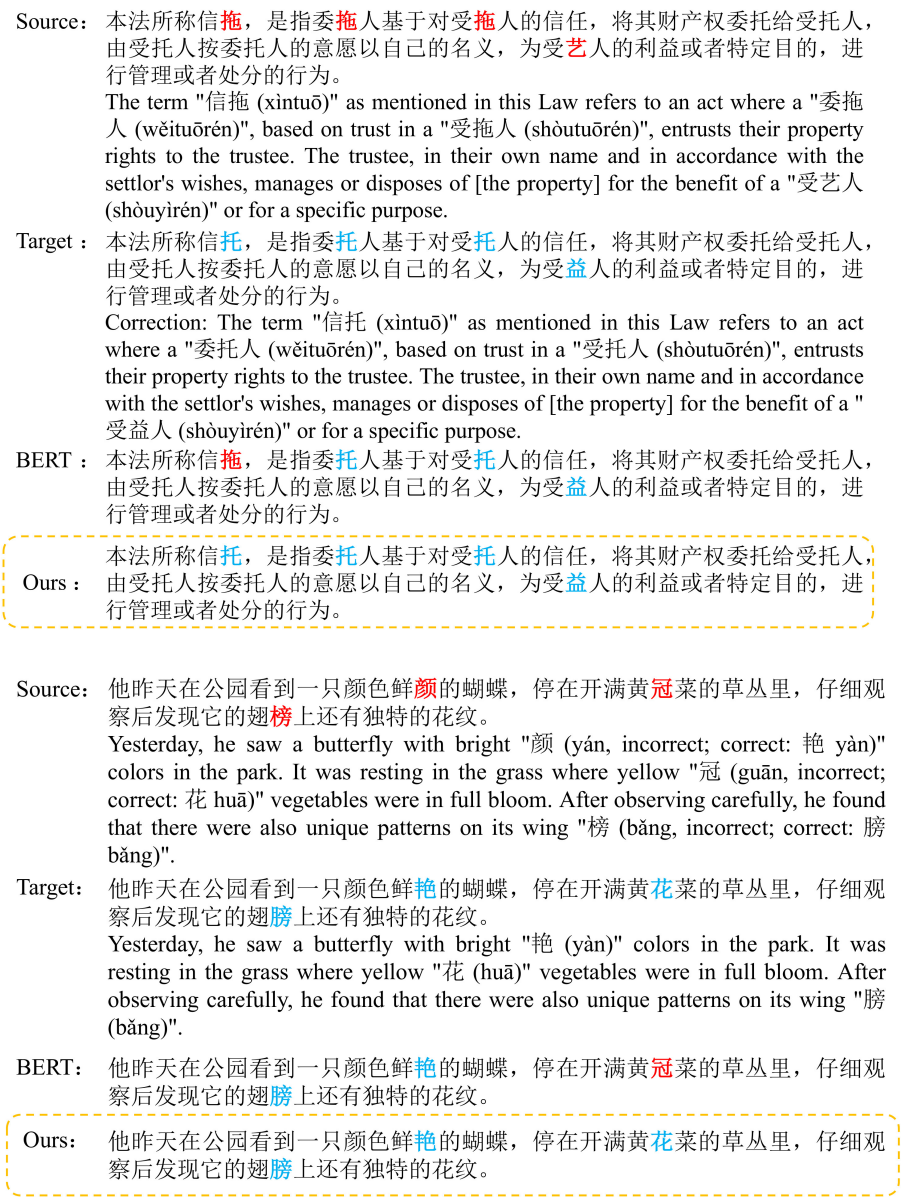

Figure 5: Cases of handling multiple erroneous samples.

all misspelled terms, ensuring the precise expression of legal concepts.

In the second example about a descriptive scene, the source text has misspellings in words describing the butterfly's color, the plant in the grass, and the butterfly's wing part. BERT corrects the butterfly's color term correctly but makes an error in the plant name. Our model, however, successfully corrects all these misspellings, restoring the text to its accurate and natural state.

These cases demonstrate that general-purpose models like BERT have limitations in handling multiple and domain-specific spelling errors, often failing to achieve full correction. Our model, with its domain - adapted capabilities or advanced error-correction mechanisms, exhibits superior performance in accurately identifying and correcting various types of spelling mistakes, showcasing its effectiveness in complex Chinese spelling correction tasks.

5 Conclusions

This paper introduced a multi-granularity curriculum learning framework for Chinese spelling correction, focusing on the legal domain. By combining batch-level sample scheduling with instance-level uncertainty weighting, mgCL addresses the dual challenges of error complexity and domain-specific terminology. Experiments on the CNLAW dataset demonstrated state-of-the-art results, achieving a

98.02% F1 score and substantially reducing false positive corrections. The consistent improvements observed on the general-domain SIGHAN15 dataset further confirm the robustness and adaptability of the framework. Beyond empirical performance, this work contributes a domain-specific benchmark (CNLAW) that can support future evaluation and model development. The legal domain case study illustrates the importance of minimizing false positives in contexts where terminological precision is critical. Moreover, the proposed framework is not restricted to legal text: the multi-granularity curriculum design provides a general training strategy that can be adapted to other specialized domains such as healthcare or finance. Future research should investigate integration with LLMs, explore richer uncertainty estimation methods, and extend evaluation to additional professional domains. Taken together, these directions highlight the broader potential of mgCL to support reliable and domain-aware spelling correction across diverse applications in computational linguistics and informatics.

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Yu, J.; Li, Z. (2014). Chinese spelling error detection and correction based on language model, pronunciation, and shape, *Proceedings of The Third CIPS-SIGHAN Joint Conference on Chinese Language Processing*, 220–223, 2014.

[2] Wang, D.; Song, Y.; Li, J.; Han, J.; Zhang, H. (2018). A hybrid approach to automatic corpus generation for Chinese spelling check, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2517–2527, 2018.

[3] Shih-Hung Wu,Chao-Lin Liu, and Lung-Hao Lee (2013). Chinese Spelling Check Evaluation at SIGHAN Bake-off 2013, *Proceedings of the 7th SIGHAN Workshop on Chinese Language Processing*, 35–42, 2013.

[4] Yu, L.-C.; Lee, L.-H.; Tseng, Y.-H.; Chen, H.-H. (2014). Overview of SIGHAN 2014 bake-off for Chinese spelling check, *Proceedings of The Third CIPS-SIGHAN Joint Conference on Chinese Language Processing*, 126–132, 2014.

[5] Wang, X. et al. (2024). An empirical investigation of domain adaptation ability for chinese spelling check models, *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2024.

[6] Wang, X. et al. (2024). An Unsupervised Domain-Adaptive Framework for Chinese Spelling Checking, *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(11): 1–16, 2024.

[7] 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, *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, 160–169, 2019.

[8] Wang, B.; Che, W.; Wu, D.; Wang, S.; Hu, G.; Liu, T. (2021). Dynamic connected networks for chinese spelling check, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2437–2446, 2021.

[9] Bai, J.; Bai, S.; Chu, Y. et al. (2023). Qwen technical report, *arXiv preprint arXiv:2309.16609*, 2023.

[10] Achiam, J.; Adler, S.; Agarwal, S. et al. (2023). Gpt-4 technical report, *arXiv preprint arXiv:2303.08774*, 2023.

[11] Liu, S.; Yang, T.; Yue, T.; Zhang, F.; Wang, D. (2021). PLOME: Pre-training with misspelled knowledge for Chinese spelling correction, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2991–3000, 2021.

[12] Wu, H.; Zhang, H.; Xuan, R.; Song, D. (2024). Bi-DCSpell: A bidirectional detector-corrector interactive framework for Chinese spelling check, *Findings of the Association for Computational Linguistics: EMNLP 2024*, 3974–3984, 2024.

[13] Liu, L.; Wu, H.; Zhao, H. (2024). Chinese Spelling Correction as Rephrasing Language Model, *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17): 18662–18670, 2024.

[14] Bengio, Y.; Louradour, J.; Collobert, R. et al. (2009). Curriculum learning, *Proceedings of the 26th annual international conference on machine learning*, 41–48, 2009.

[15] Gan, Z.; Xu, H.; Zan, H. (2021). Self-supervised curriculum learning for spelling error correction, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 3487–3494, 2021.

[16] Zhou, Y.; Yang, B.; Wong, D. F.; Wan, Y.; Chao, L. S. (2020). Uncertainty-aware curriculum learning for neural machine translation, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6934–6944, 2020.

[17] Li, J.; Wang, Q.; Mao, Z.; Guo, J.; Yang, Y.; Zhang, Y. (2022). Improving Chinese spelling check by character pronunciation prediction: The effects of adaptivity and granularity, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 4275–4286, 2022.

[18] Shulin Liu, Shengkang Song, Tianchi Yue, Tao Yang, Huihui Cai, TingHao Yu, Shengli sun. CRASpell: A Contextual Typo Robust Approach to lmprove Chinese Spelling Correction, *Findings of the Association for Computational Linguistics*, 3008–3018, 2022.

[19] Liang, Z.; Quan, X.; Wang, Q. (2023). Disentangled phonetic representation for Chinese spelling correction, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 13509–13521, 2023.

[20] Huang, L.; Li, J.; Jiang, W.; Zhang, Z.; Chen, M.; Wang, S.; Xiao, J. (2021). Phmospell: Phonological and morphological knowledge guided chinese spelling check, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 5958–5967, 2021.

[21] 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, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 871–881, 2020.

[22] Liu, C.; Zhang, K.; Jiang, J.; Liu, Z.; Tao, H.; Gao, M.; Chen, E. (2024). ARM: An alignment-and-replacement module for Chinese spelling check based on LLMs, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 10156–10168, 2024.

[23] 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, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 716–728, 2021.

[24] Zhang, S.; Huang, H.; Liu, J.; Li, H. (2020). Spelling error correction with soft-masked BERT, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 882–890, 2020.

[25] Zhu, C.; Ying, Z.; Zhang, B.; Mao, F. (2022). Mdcspell: A multi-task detector-corrector framework for chinese spelling correction, *Findings of the Association for Computational Linguistics: ACL 2022*, 1244–1253, 2022.

[26] Huang, H.; Ye, J.; Zhou, Q.; Li, Y. (2023). A Frustratingly Easy Plug-and-Play Detection-and-Reasoning Module for Chinese Spelling Check, *Findings of the Association for Computational Linguistics: EMNLP 2023*, 11514–11525, 2023.

[27] Wu, H.; Zhang, H.; Xuan, R.; Song, D. (2024). Bi-DCSpell: A Bi-directional Detector-Corrector Interactive Framework for Chinese Spelling Check, *findings of the Association for Computational Linguistics: EMNLP 2024*, 3974–3984, 2024.

[28] Sarafianos, N.; Giannakopoulos, T.; Nikou, C.; Kakadiaris, I. A. (2018). Curriculum learning of visual attribute clusters for multi-task classification, *Pattern Recognition*, 94–108, 2018.

[29] Wang, Y.; Gan, W.; Yang, J.; Wu, W.; Yan, J. (2019). Dynamic Curriculum Learning for Imbalanced Data Classification, *Proceedings of IEEE/CVF International Conference on Computer Vision*, 5016–5025, 2019.

[30] Guo, S.; Huang, W.; Zhang, H.; Zhuang, C.; Dong, D.; Scott, M. R.; Huang, D. (2018). CurriculumNet: Weakly Supervised Learning from Large-Scale Web Images, *Proceedings of Computer Vision – ECCV 2018: 15th European Conference*, 139–154, 2018.

[31] Cirik, V.; Hovy, E.; Morency, L. P. (2016). Visualizing and understanding curriculum learning for long short-term memory networks, *arXiv preprint arXiv:1611.06204*, 2016.

[32] Liu, C.; He, S.; Liu, K.; Zhao, J. (2018). Curriculum learning for natural answer generation, *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 4223–4229, 2018.

[33] Kocmi, T.; Bojar. (2017). Curriculum Learning and Minibatch Bucketing in Neural Machine Translation, *Proceedings of the International Conference Recent Advances in Natural Language Processing*, 379–386, 2017.

[34] Thompson, B.; Khayrallah, H.; Anastasopoulos, A.; McCarthy, A. D.; Duh, K.; Marvin, R.; McNamee, P.; Gwinnup, J.; Anderson, T.; Koehn, P. (2018). Freezing Subnetworks to Analyze Domain Adaptation in Neural Machine Translation, *Proceedings of the Third Conference on Machine Translation: Research Papers*, 124–132, 2018.

[35] Zhang, X.; Shapiro, P.; Kumar, G.; McNamee, P.; Carpuat, M.; Duh, K. (2019). Curriculum Learning for Domain Adaptation in Neural Machine Translation, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1903–1915, 2019.

[36] Wang, W.; Caswell, I.; Chelba, C. (2019). Dynamically Composing Domain-Data Selection with Clean-Data Selection by "Co-Curricular Learning" for Neural Machine Translation, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1282–1292, 2019.

[37] Kumar, G.; Foster, G.; Cherry, C.; Krikun, M. (2019). Reinforcement Learning Based Curriculum Optimization for Neural Machine Translation, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2054–2061, 2019.

[38] Zhang, C.; Li, W. (2026). Enhancing Graph Neural Network Vulnerability Detection via Dynamic Edge Removal and Natural Language Processing Integration, *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL*, 21(1), 2026. doi:10.15837/ijccc.2026.1.6905.

[39] Zhang, X.; Kumar, G.; Khayrallah, H.; Murray, K.; Gwinnup, J.; Martindale, M. J.; McNamee, P.; Duh, K.; Carpuat, M. (2018). An Empirical Exploration of Curriculum Learning for Neural Machine Translation, *CoRR*, abs/1811.00739, 2018.

[40] Zhang, D.; Li, Y.; Bai, L.; Zhang, H.; Li, Y.; Lin, H.; Zheng, H.-T.; Su, X.; Shan, Z. (2025). Loss-Aware Curriculum Learning for Chinese Grammatical Error Correction, *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5, 2025.

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:

Zhou, S.; Liu, Y. (2026). Multi-granularity Curriculum Learning for Chinese Spelling Correction in Legal Texts, *International Journal of Computers Communications & Control*, 21(3), 7213, 2026.
https://doi.org/10.15837/ijccc.2026.3.7213