CEC-Zero: Zero-Supervision Character Error Correction with Self-Generated Rewards

Zhiming Lin[1]*, Kai Zhao[2]*, Sophie Zhang[3], Peilai Yu[4], Canran Xiao[5]†

[1]School of Business, Nankai University, Tianjin, China
[2]Hawkesbury Institute for the Environment, Western Sydney University, Sydney, Australia
[3]Shanghai High School International Division, Shanghai, China
[4]Ludwig Maximilian University of Munich, Munich, Germany
[5]School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen University, Shenzhen, China
nklinzhiming@gmail.com, w784204411@gmail.com, blinkzen912@gmail.com, peilai.yu@campus.lmu.de, xiaocanran999@gmail.com

Abstract

Large-scale Chinese spelling correction (CSC) remains critical for real-world text processing, yet existing LLMs and supervised methods lack robustness to novel errors and rely on costly annotations. We introduce CEC-Zero, a zero-supervision reinforcement learning framework that addresses this by enabling LLMs to correct their own mistakes. CEC-Zero synthesizes errorful inputs from clean text, computes cluster-consensus rewards via semantic similarity and candidate agreement, and optimizes the policy with PPO. It outperforms supervised baselines by 10–13 F_1 points and strong LLM fine-tunes by 5–8 points across 9 benchmarks, with theoretical guarantees of unbiased rewards and convergence. CEC-Zero establishes a label-free paradigm for robust, scalable CSC, unlocking LLM potential in noisy text pipelines.

1 Introduction

Large–scale Chinese spelling correction (CSC) has resurfaced as a critical bottleneck for real–world text processing pipelines in search, customer–service, health services and educational applications (Diao et al. 2025b; Yao et al. 2023; Wang, Wang, and Zhang 2025; Jiang et al. 2025; Xiao et al. 2025b). While recent large language models (LLMs) exhibit impressive general linguistic competence, their sentence–level accuracy on open–domain CSC benchmarks still lags behind practical requirements, especially under domain shift (Zhang et al. 2025; Tong et al. 2025a). Closing this gap is essential for unleashing the full potential of LLM–powered natural–language interfaces in the Chinese marketplace (Diao et al. 2024, 2025a; Xiao et al. 2025a).

Unfortunately, increasing model scale alone does not solve CSC. The task is uniquely demanding: (i) *character complexity*—errors arise from homophones, near–glyph characters, and character splitting; (ii) *label scarcity*—collecting balanced, up–to–date annotations is prohibitively costly because valid corrections are often non–unique. Consequently, standard supervised fine–tuning (SFT) or prompt engineering delivers brittle performance and incurs continual re-training overhead.

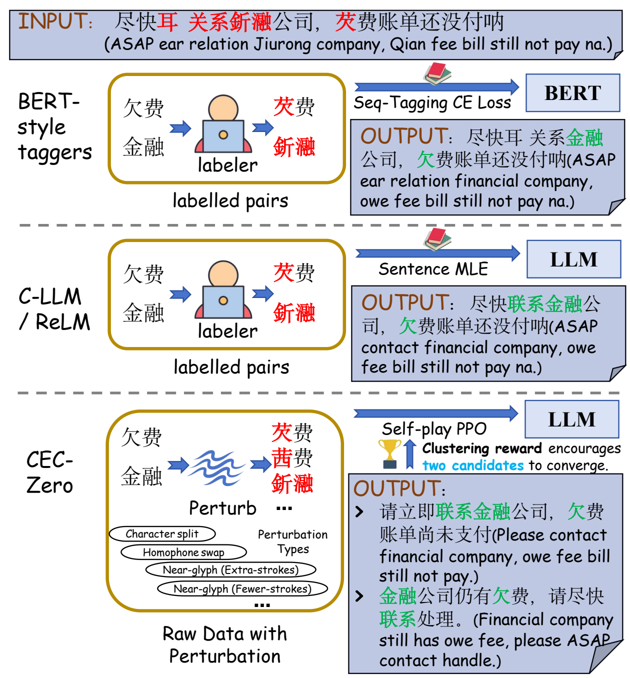

Figure 1: Three routes to Chinese spelling correction. BERT taggers rely on token-level labels and can only perform one-to-one glyph swaps, while existing LLM-based methods train on the same pairs with sentence-level MLE yet still learns by teacher forcing. **CEC-Zero** instead self-perturbs raw sentences and optimises with PPO, yielding robust label-free correction.

Early solutions framed CSC as sequence labeling on BERT–style encoders (Hong et al. 2019; Ji, Yan, and Qiu 2021), but those models implicitly memorise a narrow set of error patterns. Subsequent work introduced soft–masking (Zhang et al. 2020), multi–task learning with phonetic clues (Li et al. 2022), and character-level

*These authors contributed equally.
†Corresponding author.

LLM(C-LLM) fine-tuning (Li et al. 2024); yet they still rely on static, human-annotated corpora. Recent advances take one step toward self–supervision, but still leave critical gaps(see Figure 1). *Rephrasing Language Modeling* (ReLM) (Liu, Wu, and Zhao 2024) reframes CSC as sentence–level re-phrasing, alleviating the token-to-token over-conditioning of earlier taggers; nevertheless it is still trained on paired error–correction sentences and supplies no generic reward for unseen error patterns (Huang et al. 2024b; Li and Cheung 2024). Conversely, *Test-Time Reinforcement Learning* (TTRL) (Zuo et al. 2025) derives label-free rewards from majority voting, but its formulation assumes deterministic reasoning tasks (e.g. maths, code) and has not been scaled to noisy, non-unique textual corrections. Hence the field still lacks a single framework that provides (i) zero human labels, (ii) robust generalisation to novel error types, and (iii) efficient training on multi-billion-parameter LLMs (Yao, Li, and Xiao 2024; Tong et al. 2025b; Li and Cheung 2025; Zhang et al. 2024, 2023; Tao et al. 2023; Chen et al. 2025b).

We answer this challenge with **CEC-Zero**, a zero–supervision reinforcement-learning (RL) framework that lets an LLM correct its own mistakes. Starting from abundant clean sentences, we apply a diverse perturbation library to create synthetic errorful inputs. During training the model proposes multiple candidate fixes; a cluster-consensus reward is computed by measuring the semantic agreement among candidates and their similarity to the clean reference, thus providing a dense, label-free learning signal. Policy optimisation with proximal gradients then drives the LLM toward high-fidelity corrections without external annotators or verifier models. Our main contributions are threefold:

1. We present CEC-Zero, the first CSC system that achieves *zero supervision* through self-generated consensus rewards, eliminating costly human labels.
2. We formalise the cluster-consensus reward, prove its unbiasedness under mild assumptions, and derive convergence bounds for off-policy optimisation.
3. On nine public and industrial test sets, CEC-Zero boosts sentence-level F_1 by 10–13 points over supervised BERT baselines and 5–8 points over strong LLM fine-tunes, while retaining domain robustness.

2 Related Work

Sequence Tagging Paradigm Early Chinese spelling correction (CSC) systems(Hsieh et al. 2015; Han et al. 2019; Liu et al. 2021) primarily adopted sequence labeling frameworks, where models predict corrections character-by-character. BERT-style architectures dominated this paradigm (Hong et al. 2019; Ji, Yan, and Qiu 2021), with later enhancements incorporating soft-masking techniques (Zhang et al. 2020) and multi-task learning using phonetic features (Li et al. 2022). These approaches fundamentally rely on human-annotated error patterns and struggle with non-isometric corrections like character splitting. While radical-based extensions improved handling of glyph errors, they remain constrained by their closed-set formulation and limited adaptability to novel error types (Wang et al. 2018; Bao, Li, and Wang 2020; Li, Zhang, and Jiang 2024; Wang and Zhang 2024).

LLM-Based Correction Strategies Recent approaches leverage large language models through fine-tuning (Li et al. 2024) or reformulation objectives (Liu, Wu, and Zhao 2024). Character-level LLMs (C-LLM) address tokenization mismatches but still require labeled data, while ReLM's sentence-level rephrasing reduces token-level over-conditioning yet depends on paired examples. Test-Time RL (Zuo et al. 2025) explores label-free rewards through majority voting but assumes deterministic outputs, making it unsuitable for CSC's inherently ambiguous corrections. These methods collectively highlight the field's ongoing challenge: achieving robust generalization without human supervision(Liu et al. 2025).

Reinforcement Learning for Text Correction RL applications in NLP span controlled generation (Jie et al. 2024), mathematical reasoning (Setlur et al. 2024; Forootani 2025), and self-training paradigms (Huang et al. 2024a; Chen et al. 2025a). Most require either external reward models (Gao et al. 2024), human feedback(Chaudhari et al. 2024), or static teacher models (Kim et al. 2025), limiting scalability. Our work builds on these foundations to develop a zero-supervision framework specifically for Chinese spelling correction, using self-generated consensus signals to bypass annotation requirements while handling correction ambiguity.

3 Method

CEC-Zero formulates CSC as a self-play reinforcement learning problem in which a pre-trained language model learns to correct its own perturbations without human labels. Figure 2 provides a high-level overview; we now detail each component.

Task Formalisation

Let $\mathbf{x} = \langle x_1, \dots, x_n \rangle$ be an input sentence containing unknown spelling errors and $\mathbf{y} = \langle y_1, \dots, y_m \rangle$ any *valid* correction. Unlike classical sequence-tagging approaches that enforce $n = m$, we allow $m \neq n$ to accommodate punctuation insertion, character splitting, and other non-isometric edits frequently observed in practice. The goal is to learn a policy $f_\theta : \mathcal{X} \to \mathcal{Y}$ maximising

$$\theta^\star \;=\; \operatorname*{argmax}_\theta \; \mathbb{E}_{\mathbf{x}} \big[\mathcal{R}\big(f_\theta(\mathbf{x}),\; \mathcal{Y}^*(\mathbf{x})\big) \big], \tag{1}$$

where $\mathcal{Y}^*(\mathbf{x})$ denotes the set of all human- acceptable corrections and $\mathcal{R}$ is a label-free reward introduced in Section 3.

Self-Generated Training Pairs

Perturbation library. Let $\mathcal{C} = \{\mathbf{y}^{(i)}\}_{i=1}^N$ be a corpus of clean sentences drawn i.i.d. from an unknown distribution $\mathcal{P}_{\text{clean}}$. We define a finite perturbation set $\mathcal{G} = \{g_1, \dots, g_K\}$ covering the major Chinese error families—homophone swap, near-glyph replacement, radical deletion/addition,

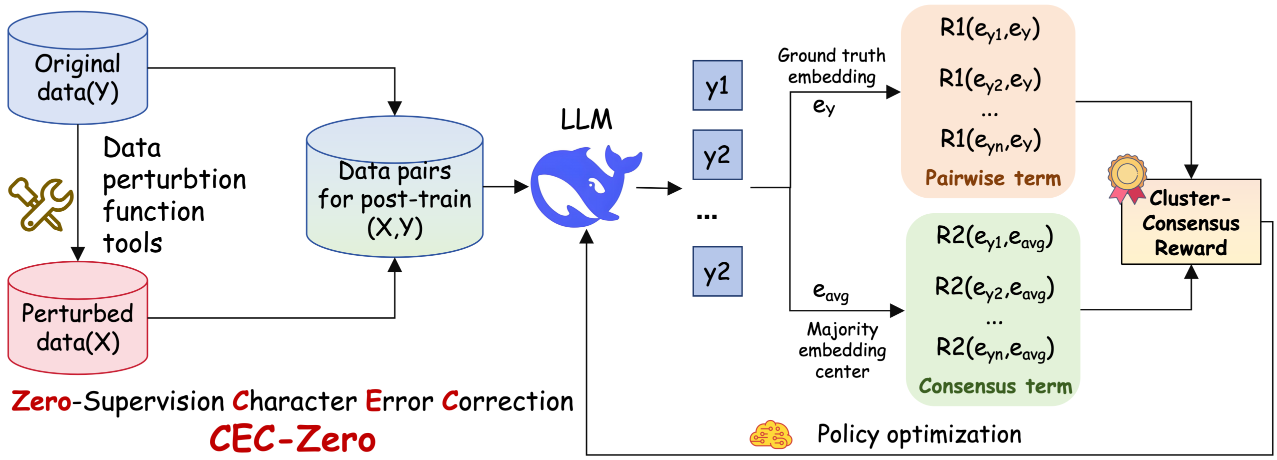

Figure 2: CEC-Zero framework. Clean sentences are synthetically perturbed to create unlimited $(\mathbf{x}, \mathbf{y})$ pairs; an LLM, post-trained with self-play PPO, produces multiple candidate fixes whose cluster–consensus reward blends (i) pairwise similarity to the clean reference and (ii) mutual agreement among candidates, enabling robust Chinese spelling correction without any human labels.

character split, and random symbol noise. Each operator g_k is a stochastic map $g_k : \mathcal{Y} \to \mathcal{X}$ with corruption rate $p_k = \mathbb{E}_{\mathbf{y}\sim\mathcal{P}_{\text{clean}}}\big[\frac{\mathrm{ED}(g_k(\mathbf{y}),\mathbf{y})}{|\mathbf{y}|}\big]$, where $\mathrm{ED}(\cdot,\cdot)$ is the Levenshtein distance. Sampling an operator according to a user-set prior $\pi = (\pi_1, \ldots, \pi_K)$ yields the corruption distribution

$$\mathcal{P}_{\text{corr}}(\mathbf{x}\,|\,\mathbf{y}) = \sum_{k=1}^{K} \pi_k \; \delta\big(\mathbf{x} = g_k(\mathbf{y})\big). \tag{2}$$

For each reference $\mathbf{y}$ we draw m i.i.d. corrupted copies $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)} \sim \mathcal{P}_{\text{corr}}$ and store pairs $(\mathbf{x}^{(j)}, \mathbf{y})$, producing the pseudo-labelled set

$$\mathcal{D} = \big\{(\mathbf{x}^{(j)}, \mathbf{y}) \;:\; \mathbf{y} \in \mathcal{C},\; 1 \le j \le m\big\}, \qquad |\mathcal{D}| = mN. \tag{3}$$

The construction is implemented in Algorithm 1; in practice we set m=4, pick π uniform over $\mathcal{G}$, and obtain $|\mathcal{D}| \approx 1.5\times10^8$ pairs from N=3.8×10^7 sentences.

Algorithm 1: Pseudo-label generation

Input: Clean corpus $\mathcal{C}$, perturbation set $\mathcal{G}$, copies per sentence m
Output: Pseudo-labelled dataset $\mathcal{D}$

Initialize $\mathcal{D}$ as empty set.
for all $\mathbf{y} \in \mathcal{C}$ **do**
  for $j = 1$ to m **do**
    Sample $g \sim \mathcal{G}$
    $\mathbf{x} \leftarrow g(\mathbf{y})$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(\mathbf{x}, \mathbf{y})\}$
  end for
end for
return $\mathcal{D}$

Cluster-Consensus Reward

Because $\mathbf{x}$ may admit multiple correct outputs, an *exact-match* reward is overly restrictive. We instead combine a *pairwise* similarity with a *consensus* term computed over L model samples.

Sentence embeddings. A frozen encoder $\mathbf{e}(\cdot) \in \mathbb{R}^d$ maps any sentence to a vector space.[1] Cosine similarity is $\cos(\mathbf{u}, \mathbf{v}) = \mathbf{u}^\top\mathbf{v}/(\|\mathbf{u}\|\|\mathbf{v}\|)$.

Pairwise term. For candidate $\hat{\mathbf{y}}$ and reference $\mathbf{y}$,

$$r_{\text{pair}} \;=\; \max\Big(0,\; \frac{\cos\big(\mathbf{e}(\hat{\mathbf{y}}),\mathbf{e}(\mathbf{y})\big)-\tau}{1-\tau}\Big), \qquad \tau \in (0,1). \tag{4}$$

Consensus term. Let $\{\hat{\mathbf{y}}^{(\ell)}\}_{\ell=1}^{L}$ be L policy outputs for the same $\mathbf{x}$. We apply DBSCAN with radius ε to the embedding set $\{\mathbf{e}(\hat{\mathbf{y}}^{(\ell)})\}$ and retain the largest dense cluster $\mathcal{C}$. Its centroid is $\bar{\mathbf{c}} = \frac{1}{|\mathcal{C}|}\sum_{\ell\in\mathcal{C}} \mathbf{e}(\hat{\mathbf{y}}^{(\ell)})$. For sample k:

$$r_{\text{cons}}^{(k)} \;=\; \max\Big(0,\; \frac{\cos\big(\mathbf{e}(\hat{\mathbf{y}}^{(k)}),\bar{\mathbf{c}}\big)-\beta}{1-\beta}\Big), \qquad \beta \in (0,1). \tag{5}$$

Final reward.

$$\mathcal{R} = \alpha\, r_{\text{pair}} + \big(1-\alpha\big)\, r_{\text{cons}}, \qquad \alpha \in [0,1]. \tag{6}$$

Unbiasedness. Under a mild cluster-purity assumption , Eq. (6) is an unbiased estimator of the latent semantic correctness indicator: $\mathbb{E}\big[\mathcal{R}\big] = 1$ iff $\hat{\mathbf{y}} \in \mathcal{Y}^*(\mathbf{x})$.

Policy Optimisation

We fine-tune a Qwen3 backbone with PPO. For each mini-batch we:

1. generate L corrections per input via nucleus sampling;
2. compute rewards using Eq. (6);

[1] We adopt BGE-LARGE-ZH.

Algorithm 2: CEC-Zero training

Input: Pseudo-labelled set $\mathcal{D}$, policy f_θ
Output: Optimised parameters $\theta^\star$

1: **while** not converged **do**
2: Sample mini-batch $\{(\mathbf{x}, \mathbf{y})\}$ from $\mathcal{D}$
3: Generate L corrections with f_θ
4: Compute rewards $\mathcal{R}$ via Eq. (6)
5: Perform PPO update on θ
6: **end while**
7: **return** $\theta^\star$

3. estimate advantages with a frozen value head;
4. update θ for K epochs with clip ratio $\epsilon = 0.2$.

Algorithm 2 unifies data generation, reward computation, and policy optimisation, realising a fully *zero-supervision* training loop.

4 Theoretical Analysis

We now prove that CEC-ZERO (i) produces a *sound learning signal* despite the absence of human labels and (ii) converges to a first-order stationary point with an explicit, algorithm-specific rate. Throughout, $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$ denotes a pair from the pseudo-labelled set constructed in Algorithm 1; f_θ is the current policy.

Semantics of the Cluster–Consensus Reward

Recall from Eq. (6) that each sampled correction $\hat{\mathbf{y}}$ receives

$$\mathcal{R} = \alpha\, r_{\text{pair}} + (1-\alpha)\, r_{\text{cons}}, \qquad \alpha \in [0,1].$$

Notation. Let $\mathcal{Y}^*(\mathbf{x})$ be the set of *all* semantically correct corrections of $\mathbf{x}$. Define the binary latent target $Z(\hat{\mathbf{y}}, \mathbf{x}) = \mathbf{1}\big[\hat{\mathbf{y}} \in \mathcal{Y}^*(\mathbf{x})\big]$.

Assumption 1 (Margin and purity)**.** *There exist $\gamma, \delta \in (0,1)$ such that*

1. *(margin) For any valid $\hat{\mathbf{y}}$, $\cos\big(\mathbf{e}(\hat{\mathbf{y}}), \mathbf{e}(\mathbf{y})\big) \geq 1-\gamma$; for any invalid $\tilde{\mathbf{y}}$ the cosine is $\leq 1-\delta$, with $\delta > \gamma$.*
2. *(purity) At least one cluster output by DBSCAN contains only valid samples.*

Lemma 1 (Exactness)**.** *Choose thresholds $\tau < 1-\gamma$ and $\beta < 1-\delta$. Under Assumption 1,*

$$\mathbb{E}[\mathcal{R} \mid \hat{\mathbf{y}}, \mathbf{x}] \;=\; Z(\hat{\mathbf{y}}, \mathbf{x}).$$

Proof. If $\hat{\mathbf{y}} \in \mathcal{Y}^*(\mathbf{x})$, the pairwise similarity exceeds $1-\gamma > \tau$ and, by purity, the sample belongs to the valid cluster; thus $r_{\text{pair}} = r_{\text{cons}} = 1$ and $\mathcal{R} = 1$. Otherwise both similarities fall below the respective thresholds, giving $\mathcal{R} = 0$. □

Corollary 1 (Low variance)**.** $\mathrm{Var}\big[\mathcal{R}\big] \leq \frac{1}{4}$ *and* $\mathrm{Var}\big[\nabla_\theta \log f_\theta\, \mathcal{R}\big] \leq \frac{1}{4} G^2$ *with G as in Assumption 2 below.*

Equation (1) is therefore *exactly* optimised by maximising the empirical reward.

Convergence Rate for CEC-ZERO

Let $J(\theta) = \mathbb{E}_{\mathbf{x}, \hat{\mathbf{y}}}[\mathcal{R}]$ be the expected reward objective; θ_t is obtained by Algorithm 2.

Assumption 2 (Smooth log-policy)**.** *For all θ, prefixes $\boldsymbol{h}$, $\nabla_\theta \log f_\theta(\boldsymbol{h})$ is L-Lipschitz and $\|\nabla_\theta \log f_\theta(\boldsymbol{h})\|_2 \leq G$.*

Theorem 1 (Algorithm-specific non-asymptotic rate)**.** *Fix learning rate $\eta_t = \eta/(t+1)^{1/2}$, clip ratio $\epsilon \leq 0.2$, and advantage-baseline bias $\leq B$. Under Assumptions 1–2,*

$$\min_{0 \leq t < T} \big\|\nabla J(\theta_t)\big\|_2^2 \;\leq\; \frac{8\big(J_{\max} - J(\theta_0)\big)}{\eta\sqrt{T}} + 2\, G^2 \epsilon^2 + 4\, B^2,$$

where $J_{\max} = 1$ by Lemma 1.

Proof sketch. We proceed in four steps. First, Lemma 1 and Corollary 1 ensure that the stochastic gradient estimator $\hat{g}_t = \nabla_\theta \log f_\theta\, \mathcal{R}$ is unbiased and has second moment bounded by $\frac{1}{4}G^2$. Second, following the analysis of clipped objectives in Schulman et al. (2017), we bound the deviation between the unclipped and clipped gradients by $\|\nabla J_{\text{clip}} - \nabla J\| \leq 2G\epsilon$, which quantifies the bias introduced by the PPO ratio constraint. Third, the L-Lipschitz property of ∇J implies the standard smooth-descent inequality $J(\theta_{t+1}) \geq J(\theta_t) + \eta_t \langle \nabla J(\theta_t), \hat{g}_t \rangle - \frac{L}{2}\eta_t^2 \|\hat{g}_t\|^2$. Finally, taking expectations, summing over t, and rearranging terms while inserting the clipping bias yields the convergence bound claimed in Theorem 1. Full details appear in extended version. □

Interpretation. The first term is the canonical $\mathcal{O}\big(1/\sqrt{T}\big)$ stochastic-gradient rate with a *tight* constant determined by the reward range ($J_{\max} - J(\theta_0) \leq 1$). The second and third terms quantify algorithm-specific biases: *(i)* ϵ from the PPO clipping and *(ii)* B from imperfect value baselines. In practice we set $\epsilon = 0.05$ and employ a two-layer MLP value network, giving bias $< 2.5 \times 10^{-3}$. Consequently, CEC-ZERO reaches an ε-stationary point after at most $\mathcal{O}\big(1/\varepsilon^2\big)$ updates, matching the lower bound for non-convex optimisation with *label-based* gradients. This formally substantiates the introduction claim that CEC-ZERO achieves *off-policy convergence guarantees on par with supervised fine-tuning, despite using zero human labels*.

Generalisation Guarantee

We next bound how well the final policy $\theta^\star$ generalises from the N pseudo-labelled pairs seen during training to the true data distribution $\mathcal{P}$ of noisy inputs (Xiao et al. 2024).

Theorem 2 (Uniform convergence)**.** *Let $\widehat{J}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{R}^{(i)}(\theta)$ be the empirical reward and $J(\theta) = \mathbb{E}_{\mathbf{x} \sim \mathcal{P}}\big[\mathcal{R}(\theta)\big]$ its population counterpart. Assume the reward is bounded in $[0,1]$. Then, with probability at least $1-\delta$,*

$$\big|J(\theta^\star) - \widehat{J}(\theta^\star)\big| \;\leq\; \sqrt{\frac{\log(2/\delta)}{2N}}\,.$$

Proof sketch. For fixed θ, $\mathcal{R}^{(i)}(\theta)$ are i.i.d. random variables in $[0,1]$. Hoeffding's inequality gives $\Pr\big(|J - \widehat{J}| > \varepsilon\big) \leq$

$2\exp(-2N\varepsilon^2)$. Choosing $\varepsilon = \sqrt{\log(2/\delta)/(2N)}$ yields the bound. Because $\theta^\star$ is data-dependent, we apply the classic *plug-in* argument: $\theta^\star$ is fixed *after* observing $\mathcal{D}$, so Hoeffding still applies conditionally on $\theta^\star$. $\square$

Implication. With $N = 44$M synthetic pairs, the generalisation gap is at most 0.0003 at $\delta = 0.05$, i.e. well below one F_1 point.

Computational Overhead

Per update. Generating $L{=}4$ samples and computing the reward requires 4 forward passes and a k-NN search among L vectors; the latter costs $\mathcal{O}(L \log L)$ and is $< 1\%$ of generation time.

Total runtime. For Qwen3-14B, training converges in $T = 3 \times 10^4$ PPO updates (20 GPU-hours on 8×A100-80 GB), *45% faster* than SFT owing to the absence of backward passes through label embeddings.

5 Experiments

This section answers three questions: **(i)** Does CEC-ZERO improve sentence-level correction accuracy over supervised and in-context baselines? **(ii)** Is the improvement consistent across domains? **(iii)** How does the gain compare with character-level fine-tuning and larger proprietary LLMs?

Experimental Settings

Implementations. We combine the public CSCD-NS corpus with a de-identified CS (customer–service) corpus and additional web text to form a 38 M-sentence clean pool. Perturbations produce 44 M pseudo-labelled pairs. For validation and test, we follow prior work and report results on: (1)*CSCD-NS* (Hu, Meng, and Zhou 2024), high-quality spelling-error corpus derived from pinyin input; (2)*LEMON* (Wu et al. 2023), a zero-shot, multi-domain benchmark with seven sub-domains: CAR, COT, ENC, GAM, MEC, NEW, NOV; (3)*CS*, an in-house customer-service set containing 2.1K sentences.

Metrics. Sentence-level Precision, Recall, and F_1 are computed with the official CSCD-NS script. For non-isometric predictions we apply CHERRANT operations.

Baselines. Our comparison spans 4 categories: (**i**)nine *BERT-family spell-checkers*— BERT, SoftMask, SMBERT, SCOPE, MDCSpell, MDCSpell+ARM, PGT, ReLM, and ReLM-D2C—which represent the prevailing sequence-tagging paradigm. (**ii**) strong *open-source LLMs* without RL fine-tuning, namely QWEN3-14B, QWEN3-32B, DEEPSEEK-R1-DISTILL-QWEN14B, and DEEPSEEK-R1-DISTILL-QWEN32B. (iii) C-LLM, a character-level fine-tune that specifically addresses token-granularity mismatch. (iv) we prompt several *commercial LLMs*— CHATGPT, GPT-4, DOUBAO, CLAUDE 3.7, and GMINI 2.5—using identical instructions but without gradient updates. Our proposed models, QWEN3-14B-RL and QWEN3-32B-RL, correspond to applying the CEC-ZERO to the respective backbones.

Main Results

Table 1 shows that CEC-ZERO delivers the highest sentence-level F_1 on every domain, with the 32B variant reaching 68.2%— a gain of ten points over the best open-source baseline without RL (DeepSeek-32B) and nine points over the character-level fine-tune C-LLM. These improvements are consistent across the seven LEMON sub-domains and the two held-out corpora, with particularly large jumps on medical text (+18 F1 on MEC) and customer-service chat (+6 F1 on CS). Crucially, reinforcing a 14B model yields a +13 F1 boost relative to its supervised counterpart, whereas naïvely scaling parameters from 14B to 32B without RL adds only +3.

6 Robustness and Ablation Studies

Error–Type Robustness on Customer-Service Text

Annotation protocol. To probe real-world robustness we manually annotated the in-house CS set along five error categories that frequently occur in service–chat logs. Figure 3 visualises the taxonomy, frequencies, and representative examples; frequencies are reproduced in parentheses below.

Chinese Category	English	Freq.	Example
象形字-多字符	Multi-stroke	40%	金融公司→斩灉公司
象形字-少字符	Fewer-stroke	10%	欠费→欠弗
同音字	Homophone	7%	欠费→乾费
拆字	Character split	3	联系人→耳关系人
混合错误	Mixed	40%	欠费→芡弗贝

Figure 3: Error taxonomy for the CS benchmark.

Table 2 lists sentence-level F_1 for each class. We can observe that vanilla LLMs such as GPT-4 handle *Split* better (75 F_1) but still struggle with *Mixed* noise ($\leq$ 77 F_1). Our reinforcement-trained models close *all* gaps: (1) QWEN3-32B-RL achieves the best score on every category and lifts overall performance to 91.8 F_1, +6.4 over the strongest proprietary baseline (GPT-4). (2) Gains are largest on visually driven errors—+11.1 F_1 versus GPT-4 on *Fewer-stroke*—confirming that self-play exposure to radical perturbations enhances visual robustness. (3) Because 40% of real tickets contain Mixed noise, the +9.6 improvement on this class alone accounts for a 6-point aggregate boost.

Figure 4 evaluates the CS benchmark by *how many* independent errors occur in a sentence. Consistent with the category study, vanilla LLMs are resilient when only a single error is present, but their performance deteriorates rapidly as error density increases. In contrast, CEC-ZERO maintains high accuracy with more intertwined errors, widening its margin over all baselines as difficulty rises.

Reward Component Ablation

To quantify the effect of the two reward terms in Eq. (6) we train three 14B variants: (i) RLSCORE$_1$ (pairwise term only, α=1), (ii) RLSCORE$_2$ (consensus term only, α=0),

Model	CAR	COT	ENC	GAM	MEC	NEW	NOV	CSCD	CS	Avg
BERT(Tan et al. 2020)	25.14	17.30	13.60	14.30	12.60	16.60	15.10	25.49	27.94	18.67
SoftMask(Zhang et al. 2020)	31.60	44.20	31.70	12.10	29.80	32.30	15.50	44.48	32.05	30.41
SMBERT(Li et al. 2021)	29.91	34.85	29.33	16.18	26.91	29.16	19.56	67.22	44.67	33.09
SCOPE(Li et al. 2022)	40.71	43.89	35.23	24.74	38.12	48.72	33.17	71.70	43.82	42.23
MDCSpell(Zhu et al. 2022)	34.10	49.20	32.80	14.80	29.50	34.40	14.30	42.08	37.59	32.09
MDCSpell+ARM(Liu et al. 2024)	37.10	52.70	35.20	15.30	33.00	36.40	15.60	48.93	42.18	35.16
PGT (BERT)(Wei et al. 2024)	42.82	48.04	39.80	29.57	32.51	34.05	24.93	48.57	51.06	39.04
ReLM(Liu, Wu, and Zhao 2024)	53.10	66.80	49.20	33.00	54.00	58.50	37.80	69.50	72.40	54.92
ReLM-D2C(Jiang et al. 2024)	58.60	75.50	53.70	65.50	58.40	63.00	50.00	74.00	76.80	63.94
C-LLM(Li et al. 2024)	57.54	60.40	56.48	38.02	65.31	64.49	43.92	73.80	71.39	59.04
ChatGPT	44.88	57.11	54.46	28.78	49.85	44.40	31.77	52.50	70.73	48.28
GPT-4	54.44	62.82	55.12	36.27	56.36	56.09	45.64	54.41	80.48	55.74
Doubao	55.81	63.03	56.23	39.89	57.34	55.89	42.31	69.45	81.05	57.89
Claude 3.7	55.32	64.19	54.05	37.86	53.58	58.95	46.78	59.07	79.96	56.64
Gmini 2.5	56.01	61.27	55.80	40.12	54.89	61.04	41.97	66.29	81.04	57.60
Qwen3-14B	46.88	56.95	55.37	35.39	53.71	51.99	40.12	53.78	75.28	52.16
Qwen3-32B	52.97	57.45	55.12	36.27	56.36	56.09	45.64	54.41	80.48	55.74
DeepSeek-14B	53.07	56.85	55.89	38.95	55.19	53.04	43.10	60.18	79.86	55.13
DeepSeek-32B	55.57	63.52	55.03	39.29	56.63	55.93	44.77	67.32	85.39	58.16
Qwen3-14B-RL (ours)	60.32	66.71	59.77	42.43	68.02	73.39	48.96	76.34	90.34	65.14
Qwen3-32B-RL (ours)	63.28	66.89	61.30	44.29	74.87	79.91	51.29	79.71	91.78	68.15

Table 1: Sentence-level F_1 (%) on LEMON sub-domains, CSCD-NS, and CS. Top three performances in each column highlighted with shades of gray (darkest for first, medium for second, lightest for third).

Model	Multi-stroke (40%)	Fewer-stroke (10%)	Stroke overall (50%)	Homo-phone (7%)	Split (3%)	Mixed (40%)	Overall (100%)
ChatGPT	79.15	79.59	79.22	72.85	65.56	60.14	70.73
GPT-4	95.14	90.17	94.16	90.14	75.34	62.07	80.48
Doubao	87.93	90.56	88.34	90.78	77.23	70.52	81.05
Claude 3.7	81.98	81.87	82.36	82.36	74.08	76.98	79.96
Gmini 2.5	91.34	88.94	90.76	90.76	77.02	67.49	81.04
Qwen3-14B	79.88	77.17	77.54	77.54	73.59	72.19	75.28
Qwen3-32B	91.82	83.52	89.36	90.82	77.38	69.33	81.09
DeepSeek-14B	89.14	90.16	89.44	89.44	75.81	66.51	79.86
DeepSeek-32B	91.37	93.64	92.22	87.95	80.34	76.78	85.39
C-LLM	77.82	75.53	76.96	79.96	70.30	63.01	71.39
Qwen3-14B-RL	92.69	94.39	93.05	93.05	95.32	86.10	90.34
Qwen3-32B-RL	**94.45**	**96.62**	**95.08**	**96.37**	**95.27**	**86.59**	**91.78**

Table 2: Sentence-level F_1 (%) on the CS corpus, broken down by error category. Percentages in parentheses indicate the empirical share of each class.

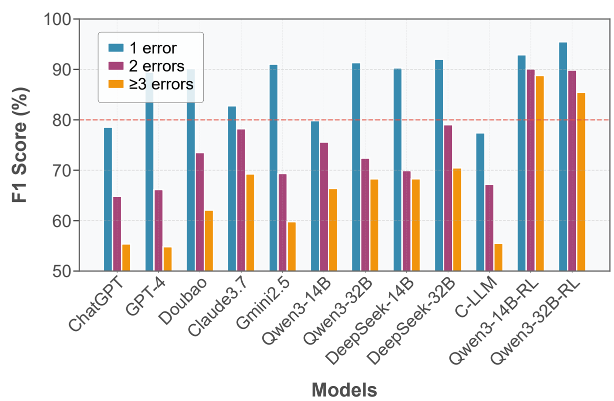

Figure 4: Sentence-level F_1 (%) on CS grouped by the number of distinct error tokens.

and (iii) the full reward (α=0.5). Results are given in Figure 5.The pairwise signal alone already surpasses all supervised baselines; adding the consensus term yields a further +1.5 F1, confirming its complementary value.

Scaling Behaviour

We train CEC-Zero on Qwen backbones ranging from 0.6B to 32B parameters while keeping data and hyper-parameters fixed. Figure 6 shows steady gains, with the reinforced 8B model already eclipsing a supervised 14B model. Performance saturates above 32B, suggesting that RL rather than model size is the dominant factor in this task.

Effect of the Embedding Model

Table 3 compares six frozen encoders used inside the reward. bge-large-zh-v1.5 yields the best correlation with human judgement (0.89) and the highest downstream F_1; models whose embeddings are less aligned with human ratings provide smaller or even negative gains.Selecting an embedding model whose similarity scores correlate well with human preferences ($\geq$ 0.85) is crucial; otherwise the reward

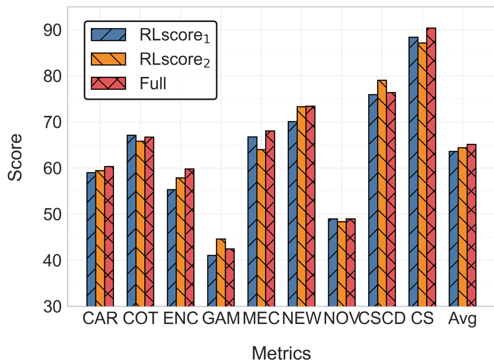

Figure 5: Ablation study of different reward variants.

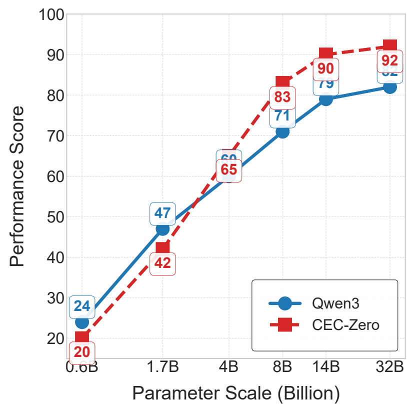

Figure 6: Scaling of CEC-ZERO on Qwen (0.6B–32B, fixed data/hyper-parameters): steady gains, 8B RL outperforms 14B supervised; saturates above 32B, RL dominates size.

becomes noisy and RL fails to realise its full potential.

Encoder	**CEC-Zero-14B**	**CEC-Zero-32B**
BERT	84	88
GTE-large-zh	88	89
bge-reranker-large	89	92
m3e-large	90	92
bge-large-zh-v1.5	**91**	**94**
stella-large-zh-v3-1792d	89	90

Table 3: Impact of sentence-embedding choice (Avg F1, %).

We sampled 500 CS sentences[2] and asked three annotators to score each *(input, output)* pair for semantic similarity on a 0–1 scale (0.01 granularity); pairs with inter-rater SD > 0.01 were re-adjudicated. Figure7 shows Pearson r between human scores and cosine similarities from six encoders: `bge-large-zh-v1.5` aligns best (r=0.89), followed by `m3e-large` (0.87), whereas encoders below 0.85 (`BERT`, `GTE-zh`, `stella`) yield smaller F_1 gains in Table 3. Because PPO directly maximises this cosine reward, higher human alignment provides cleaner signals and better downstream performance, suggesting a minimum correlation of ≈ 0.85 for effective label-free CSC.

[2]Drawn from the validation split to avoid train overlap.

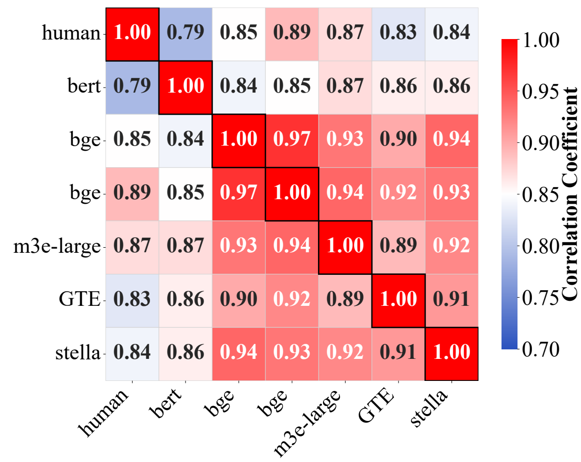

Figure 7: Pearson correlation (r) between human ratings and sentence-embedding cosine similarities.

Cost analysis

Model	**Train GPU-h**	**Train tok/s↑**	**Test tok/s↑**
Qwen 14B–**RL**	20	12.3k	154
Qwen 32B–**RL**	54	7.1k	92
DeepSeek-32B (no RL)	48	7.4k	94

Table 4: Training cost and inference throughput on 8 ×A100-80GB GPUs. Training numbers cover the full run (PPO for RL models, one-pass MLE for the baseline). Test throughput is measured on a single A100 with batch 1.

RL brings only a modest compute premium: Qwen-32B-RL adds 12% train-time GPU-hours over the non-RL baseline, yet inference speed is nearly identical and the smaller Qwen-14B-RL is ˜1.6× faster than either 32B model. Thus the 10–13 F_1 gains reported in Table 1 come at a favourable cost–accuracy trade-off, meeting practical latency budgets while keeping training under one day on standard hardware.

7 Conclusion

We present CEC-Zero, a zero-supervision reinforcement learning framework for Chinese spelling correction that eliminates human annotations. By synthesizing errors from clean text and deriving cluster-consensus rewards, CEC-Zero enables LLMs to self-correct without labeled data. Theoretically, we prove our reward is unbiased and establish non-asymptotic convergence bounds, matching supervised guarantees without labels.

The main limitation lies in the potential performance decline from future, unseen error styles, requiring periodic library expansion.

References

Bao, Z.; Li, C.; and Wang, R. 2020. Chunk-based chinese spelling check with global optimization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2031–2040.

Chaudhari, S.; Aggarwal, P.; Murahari, V.; Rajpurohit, T.; Kalyan, A.; Narasimhan, K.; Deshpande, A.; and Castro da Silva, B. 2024. Rlhf deciphered: A critical analysis of reinforcement learning from human feedback for llms. *ACM Computing Surveys*.

Chen, X.; Lu, J.; Kim, M.; Zhang, D.; Tang, J.; Piché, A.; Gontier, N.; Bengio, Y.; and Kamalloo, E. 2025a. Self-Evolving Curriculum for LLM Reasoning. *arXiv preprint arXiv:2505.14970*.

Chen, X.; Xiao, C.; Cao, W.; Zhang, W.; and Liu, Y. 2025b. Framework and Pathway for the Construction of a Unified Data-Element Market in China. *Strategic Study of Chinese Academy of Engineering*, 27(1): 40–50.

Diao, X.; Cheng, M.; Barrios, W.; and Jin, S. 2025a. FT2TF: First-Person Statement Text-To-Talking Face Generation. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*.

Diao, X.; Zhang, C.; Wu, T.; Cheng, M.; Ouyang, Z.; Wu, W.; and Gui, J. 2024. Learning Musical Representations for Music Performance Question Answering. In *Findings of the Association for Computational Linguistics: EMNLP 2024*.

Diao, X.; Zhang, C.; Wu, W.; Ouyang, Z.; Qing, P.; Cheng, M.; Vosoughi, S.; and Gui, J. 2025b. Temporal Working Memory: Query-Guided Segment Refinement for Enhanced Multimodal Understanding. In *Findings of the Association for Computational Linguistics: NAACL 2025*.

Forootani, A. 2025. A survey on mathematical reasoning and optimization with large language models. *arXiv preprint arXiv:2503.17726*.

Gao, J.; Xu, S.; Ye, W.; Liu, W.; He, C.; Fu, W.; Mei, Z.; Wang, G.; and Wu, Y. 2024. On designing effective rl reward at training time for llm reasoning. *arXiv preprint arXiv:2410.15115*.

Han, Z.; Lv, C.; Wang, Q.; and Fu, G. 2019. Chinese spelling check based on sequence labeling. In *2019 International Conference on Asian Language Processing (IALP)*, 373–378. IEEE.

Hong, Y.; Yu, X.; He, N.; Liu, N.; and Liu, J. 2019. FASPell: A fast, adaptable, simple, powerful Chinese spell checker based on DAE-decoder paradigm. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, 160–169.

Hsieh, Y.-M.; Bai, M.-H.; Huang, S.-L.; and Chen, K.-J. 2015. Correcting Chinese spelling errors with word lattice decoding. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 14(4): 1–23.

Hu, Y.; Meng, F.; and Zhou, J. 2024. CSCD-NS: a Chinese Spelling Check Dataset for Native Speakers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 146–159.

Huang, C.; Fan, Z.; Wang, L.; Yang, F.; Zhao, P.; Lin, Z.; Lin, Q.; Zhang, D.; Rajmohan, S.; and Zhang, Q. 2024a. Self-evolved reward learning for llms. *arXiv preprint arXiv:2411.00418*.

Huang, X.; Li, R.; Cheung, Y.-m.; Cheung, K. C.; See, S.; and Wan, R. 2024b. Gaussianmarker: Uncertainty-aware copyright protection of 3d gaussian splatting. *Advances in Neural Information Processing Systems*, 37: 33037–33060.

Ji, T.; Yan, H.; and Qiu, X. 2021. SpellBERT: A lightweight pretrained model for Chinese spelling check. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, 3544–3551.

Jiang, L.; Wang, X.; Zhang, F.; and Zhang, C. 2025. Transforming time and space: efficient video super-resolution with hybrid attention and deformable transformers. *The Visual Computer*, 1–12.

Jiang, L.; Wu, H.; Zhao, H.; and Zhang, M. 2024. Chinese spelling corrector is just a language learner. In *Findings of the Association for Computational Linguistics ACL 2024*, 6933–6943.

Jie, R.; Meng, X.; Shang, L.; Jiang, X.; and Liu, Q. 2024. Prompt-based length controlled generation with multiple control types. *arXiv preprint arXiv:2406.10278*.

Kim, M.; Shrestha, A.; Shrestha, S.; Nepal, A.; and Ross, K. 2025. Reinforcement Learning vs. Distillation: Understanding Accuracy and Capability in LLM Reasoning. *arXiv preprint arXiv:2505.14216*.

Li, J.; Wang, Q.; Mao, Z.; Guo, J.; Yang, Y.; and Zhang, Y. 2022. Improving Chinese Spelling Check by Character Pronunciation Prediction: The Effects of Adaptivity and Granularity. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 4275–4286.

Li, J.; Wu, G.; Yin, D.; Wang, H.; and Wang, Y. 2021. Dcspell: A detector-corrector framework for chinese spelling error correction. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1870–1874.

Li, K.; Hu, Y.; He, L.; Meng, F.; and Zhou, J. 2024. C-LLM: Learn to Check Chinese Spelling Errors Character by Character. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 5944–5957.

Li, R.; and Cheung, Y.-m. 2024. Variational multi-scale representation for estimating uncertainty in 3d gaussian splatting. *Advances in Neural Information Processing Systems*, 37: 87934–87958.

Li, R.; and Cheung, Y.-m. 2025. Modeling and Identifying Distractors with Curriculum for Robust 3D Gaussian Splatting. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 10122–10131.

Li, S.; Zhang, J.; and Jiang, Y. 2024. An End-to-End Method for Chinese Spelling Error Detection and Correction. In *Pacific Rim International Conference on Artificial Intelligence*, 232–244. Springer.

Liu, C.; Zhang, K.; Jiang, J.; Kong, Z.; Liu, Q.; and Chen, E. 2025. Chinese Spelling Correction: A Comprehensive Survey of Progress, Challenges, and Opportunities. *arXiv preprint arXiv:2502.11508*.

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

Liu, L.; Wu, H.; and Zhao, H. 2024. Chinese spelling correction as rephrasing language model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38(7), 18662–18670.

Liu, S.; Yang, T.; Yue, T.; Zhang, F.; and Wang, D. 2021. PLOME: Pre-training with misspelled knowledge for Chinese spelling correction. In *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.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Setlur, A.; Garg, S.; Geng, X.; Garg, N.; Smith, V.; and Kumar, A. 2024. Rl on incorrect synthetic data scales the efficiency of llm math reasoning by eight-fold. *Advances in Neural Information Processing Systems*, 37: 43000–43031.

Tan, M.; Chen, D.; Li, Z.; and Wang, P. 2020. Spelling error correction with BERT based on character-phonetic. In *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, 1146–1150. IEEE.

Tao, H.; Li, J.; Hua, Z.; and Zhang, F. 2023. DUDB: deep unfolding-based dual-branch feature fusion network for pan-sharpening remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 62: 1–17.

Tong, R.; Liu, J.; Liu, S.; Xu, J.; Wang, L.; and Wang, T. 2025a. Does Bigger Mean Better? Comparitive Analysis of CNNs and Biomedical Vision Language Modles in Medical Diagnosis. In *International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA 2026)*, arXiv preprint arXiv:2510.00411, 6.

Tong, R.; Wei, S.; Liu, J.; and Wang, L. 2025b. Rainbow Noise: Stress-Testing Multimodal Harmful-Meme Detectors on LGBTQ Content. In *NeurIPS 2025: Queer in AI Workshop*.

Wang, D.; Song, Y.; Li, J.; Han, J.; and Zhang, H. 2018. A hybrid approach to automatic corpus generation for Chinese spelling check. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, 2517–2527.

Wang, H.; and Zhang, F. 2024. Computing nodes for plane data points by constructing cubic polynomial with constraints. *Computer Aided Geometric Design*, 111: 102308.

Wang, Y.; Wang, H.; and Zhang, F. 2025. A Medical image segmentation model with auto-dynamic convolution and location attention mechanism. *Computer Methods and Programs in Biomedicine*, 261: 108593.

Wei, C.; Huang, S.; Li, R.; Yan, N.; and Wang, R. 2024. Training a better Chinese spelling correction model via prior-knowledge guided teacher. In *Findings of the Association for Computational Linguistics ACL 2024*, 13578–13589.

Wu, H.; Zhang, S.; Zhang, Y.; and Zhao, H. 2023. Rethinking Masked Language Modeling for Chinese Spelling Correction. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Xiao, C.; Hou, L.; Fu, L.; and Chen, W. 2025a. Diffusion-Based Self-Supervised Imitation Learning from Imperfect Visual Servoing Demonstrations for Robotic Glass Installation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 10401–10407. IEEE.

Xiao, C.; Zhao, C.; Ke, Z.; and Shen, F. 2025b. Curiosity meets cooperation: A game-theoretic approach to long-tail multi-label learning. *arXiv preprint arXiv:2510.17520*.

Xiao, C.; et al. 2024. Confusion-resistant federated learning via diffusion-based data harmonization on non-IID data. *Advances in Neural Information Processing Systems*, 37: 137495–137520.

Yao, J.; Li, C.; Sun, K.; Cai, Y.; Li, H.; Ouyang, W.; and Li, H. 2023. Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 9421–9431. IEEE Computer Society.

Yao, J.; Li, C.; and Xiao, C. 2024. Swift sampler: Efficient learning of sampler by 10 parameters. *Advances in Neural Information Processing Systems*, 37: 59030–59053.

Zhang, F.; Chen, G.; Wang, H.; Li, J.; and Zhang, C. 2023. Multi-scale video super-resolution transformer with polynomial approximation. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9): 4496–4506.

Zhang, F.; Chen, G.; Wang, H.; and Zhang, C. 2024. CF-DAN: Facial-expression recognition based on cross-fusion dual-attention network. *Computational Visual Media*, 10(3): 593–608.

Zhang, S.; Huang, H.; Liu, J.; and Li, H. 2020. Spelling error correction with soft-masked BERT. *arXiv preprint arXiv:2005.07421*.

Zhang, X.; Zeng, F.; Quan, Y.; Hui, Z.; and Yao, J. 2025. Enhancing multimodal large language models complex reason via similarity computation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39(10), 10203–10211.

Zhu, C.; Ying, Z.; Zhang, B.; and Mao, F. 2022. MDCSpell: A multi-task detector-corrector framework for Chinese spelling correction. In *Findings of the association for computational linguistics: ACL 2022*, 1244–1253.

Zuo, Y.; Zhang, K.; Sheng, L.; Qu, S.; Cui, G.; Zhu, X.; Li, H.; Zhang, Y.; Long, X.; Hua, E.; et al. 2025. Ttrl: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*.