From Scaffolding to Assimilation: Progressive Structural Internalization for Format-Constrained Creative Text Generation

Wenhao Li[1,2,3], **Yuwei Yang** [2,3,4], **Xiaoqing Wu**[2,3,4], **Yufeng Han**[1,2,3], **Cunliang Kong**[1,2,3], **Yuzhuo Bai**[1,2,3], **Xin Cong**[5*], **Maosong Sun**[1,2,3,6*]

[1] Department of Computer Science and Technology, Tsinghua University, Beijing
[2] Beijing National Research Center for Information Science and Technology
[3] Institute for Artificial Intelligence, Tsinghua University, Beijing
[4] School of Linguistic Sciences and Arts, Jiangsu Normal University, Xuzhou
[5] Department of Statistics and Data Science, Tsinghua University, Beijing
[6] Jiangsu Collaborative Innovation Center for Language Ability, Jiangsu Normal University, Xuzhou
peterliwenhao@163.com, {congxin1995,sms}@mail.tsinghua.edu.cn

Abstract

While Large Language Models (LLMs) demonstrate remarkable capabilities in open-ended creative generation, they notably struggle with *Format-Constrained Generation* tasks—such as poetry and lyrics—where strict adherence to multidimensional structural constraints (i.e., format, phonetics, and rhyme) is prerequisite to aesthetic value. Existing paradigms predominantly rely on unreliable prompting or rigid constrained decoding strategies; the former often fails to ensure compliance, while the latter compromises inference latency and disrupts the natural probability distribution, degrading generation quality. To bridge this gap, we establish **CCP-Arena**, a rigorous testbed for Chinese Classical Poetry, and propose **Progressive Structural Internalization (PSI)**, a novel framework designed to embed external constraints into the model's intrinsic intuition. PSI initiates with *Structural Scaffolding via Explicit Cognitive Planning*, utilizing explicit template to provide a structural scaffold for subsequent generation. This is followed by a *Cascaded Reinforcement Learning* stage guided by a *Holistic Reward Model*, which optimizes for precise structural-semantic alignment. Extensive experiments demonstrate that PSI achieves state-of-the-art performance, surpassing baselines in both strict constraint adherence and literary aesthetics. Furthermore, mechanistic analysis confirms that our method effectively internalizes structural information into the model's latent representations, offering a robust and efficient solution for constrained creative generation.

1 Introduction

Creative text generation has long been a pivotal task in NLP, spanning forms such as poetry, stories, and scripts (Zhipeng et al., 2019; Fan et al., 2018; Han et al., 2024; Ma et al., 2021). It serves not only as a practical tool for human-machine co-creation but also as a critical benchmark for evaluating the "creativity" of Large Language Models (LLMs) (Peng, 2022). A crucial subset of this domain is Format-Constrained Generation, where texts must adhere to strict rules—such as melody in lyrics or metrical schemes in Sonnets—to achieve aesthetic unity (Zhang et al., 2020; Tian and Peng, 2022).

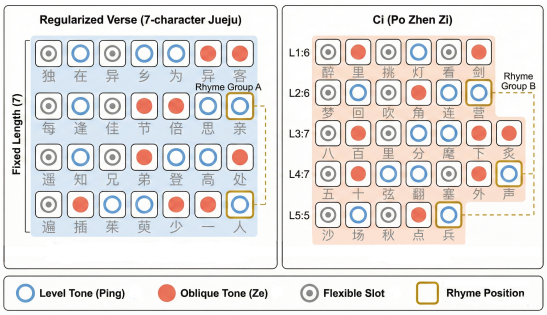

Figure 1: Structural visualization of a 7-character *Jueju* (Left) and a variable-length *Ci* (Right). Translations are provided in the Appendix F due to space limits.

However, satisfying these constraints is non-trivial. As illustrated in Figure 1, Chinese Classical Poetry imposes rigorous demands on format (character count), tonal patterns, and rhyming schemes, which are intrinsic to their beauty. Existing approaches primarily rely on Prompting (Shorten et al., 2024; Tam et al., 2024) or Constrained Decoding (Mündler et al., 2025; Wang et al., 2025a). However, the former frequently lacks reliability for complex constraints, while the latter, despite guaranteeing compliance, significantly increases inference latency and degrades semantic quality by disrupting the model's natural probability distribution. Furthermore, earlier architectural modifications (Li et al., 2020) are impractical for modern LLMs

* Corresponding author. Email: sms@mail.tsinghua.edu.cn, congxin1995@mail.tsinghua.edu.cn

Findings of the Association for Computational Linguistics: ACL 2026, pages 38372–38389

where often only fine-tuning is feasible. Thus, a central challenge remains: ***How can we enable LLMs to intrinsically master strict constraints without relying on external decoding hacks?***

To address this, we approach the problem through two synergistic dimensions: *infrastructure* and *methodology*. First, we focus on Chinese Classical Poetry, a domain with arguably the most codified and verifiable rules in literature. We systematically organize these prosodic rules to build a rigorous metric-verifier and construct a structure-aligned corpus. These resources transform raw text into **CCP-Arena**, a comprehensive benchmark designed for verifying Multi-objective Structure Constrained Creative Text Generation.

Methodologically, we propose the **Progressive Structural Internalization (PSI)** framework.[1] Unlike previous methods that treat constraints as external guardrails, PSI aims to embed these rules into the model's parameters. We begin with **Structural Scaffolding** (Sec. 3.1), using explicit template to guide reasoning and generation. This foundation supports a **Cascaded RL** mechanism guided by **Holistic Reward Modeling** (Sec. 3.2 & Sec 3.3). This process progressively internalizes external constraints, transforming them into the model's intrinsic intuition.

Extensive experiments demonstrate that our approach outperforms baselines in both structural adherence and semantic quality. Notably, our model matches the generation quality of SOTA closed-source models while achieving superior constraint compliance. Furthermore, via **Mechanistic Analysis**, we provide evidence that the model effectively internalizes structural information within its internal representations, validating our core hypothesis.

Our contributions can be summarized as follows:

- **Infrastructure (CCP-Arena):** We establish a comprehensive evaluation arena for constrained generation by formalizing the complex prosodic rules of Chinese Classical Poetry into a rigorous metric-verifier and constructing a structure-aligned corpus.
- **Methodology (PSI Framework):** We propose the Progressive Structural Internalization framework, which combines Templated-guided Reasoning with Cascaded RL and Holistic Reward Modeling. This paradigm internalizes external constraints as intrinsic intuition without architectural modifications.
- **Validation & Insight:** Through rigorous automated and human evaluations, we demonstrate that our method achieves State-of-the-Art performance in balancing strict structural compliance with literary aesthetics. Moreover, we provide mechanistic evidence confirming the successful internalization of structural constraints within the model's latent space.

[1] Source code and implementation details are available at https://github.com/THUNLP-AIPoet/PSI.

2 *CCP-Arena*

2.1 Preliminaries in CCP

Classical Chinese Poetry (CCP) primarily consists of **Regulated Verse** , characterized by fixed lengths (e.g., 5- or 7-character ***Jueju*** and ***Lushi***), and **Ci** , which follows variable-length Tune Patterns (*Cipai*). Despite structural differences, both genres represent a classic **multi-objective hard-constrained generation task**. Generating valid CCP requires satisfying three rigid constraints:

Format Constraints: The chosen template imposes a non-negotiable skeleton, pre-allocating a fixed sequence of "slots" (characters) that must be filled precisely to strictly control length and layout.

Tonal Constraints: Based on historical phonology, every slot is assigned a required **Level (*Ping*)** or **Oblique (*Ze*)** tone. This creates a **binary constraint mask** (similar to a 0/1 bitmap) that the generated tokens must strictly match.

Rhyme Constraints: Specific line endings serve as anchor positions where characters must belong to the same rhyme category (e.g., *Pingshui Rhyme*). This imposes strict lexical constraints on the candidate vocabulary at these positions.

Figure 1 visualizes these constraints using a standard 7-character *Jueju* (Left) and a variable-length *Ci* (Right) by representative poets. The Figure explicitly maps the specific **format structure** to the text, marking the optional and mandatory **rhyming positions**, and overlaying the rigid binary ***Ping-Ze*** mask on each character slot. This comparison shows the strict correspondence between the logical templates and the creative content, highlighting the zero-tolerance nature of the task.

2.2 CCP-Arena Infrastructure Building

To transform static textual resources into a dynamic and rigorous *Arena*, we establish a robust infrastructure through the digitization of verification logic and the retroactive sourcing of high-quality data.

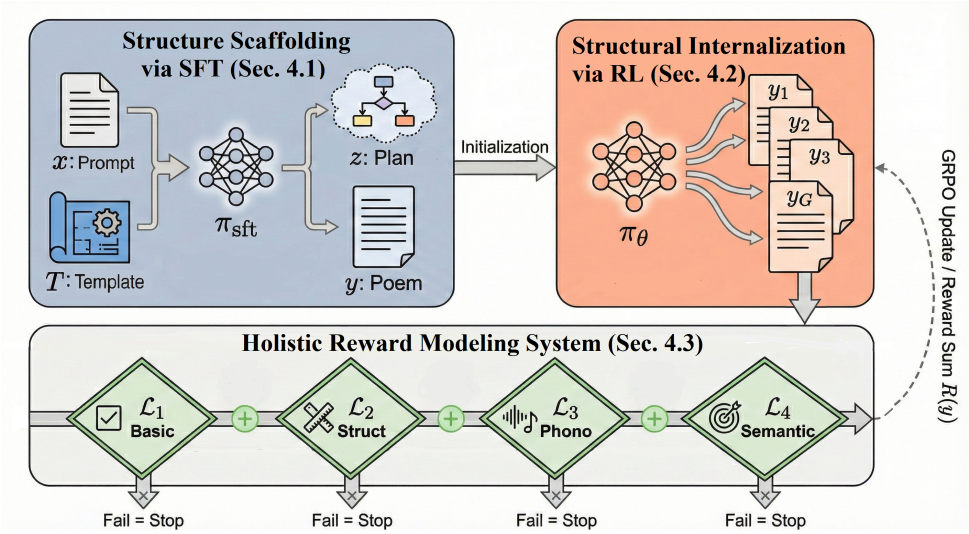

Figure 2: The **Progressive Structural Internalization (PSI)** framework. The training trajectory moves from Structural Scaffolding (SFT) to Structural Internalization (Cascaded RL). Crucially, this progression is governed by a **Holistic Reward** design that hierarchically enforces hard constraints before refining semantic alignment.

1. Rule Digitization. Traditional philological rules are descriptive; to make them actionable, we digitized two layers of logic. First, we encoded structural blueprints from authoritative sources—*Wang Li's Prosody* and the *Imperially Commissioned Prosody of Ci*—to define fixed templates for *Jueju*, *Lushi*, and various *Cipai*. Second, we digitized historical dictionaries (*Pingshui Rhyme* and *Cilin Zhengyun*) to create a queryable database mapping characters to their specific tonal categories and rhyme groups.

2. Constructing the Verifier. Integrating these digitized components, we developed the **Metric-Verifier**. This deterministic engine evaluates compliance across three critical dimensions: **Format Constraints**, **Tonal Prosody**, and **Phonological Rhyme**. The Verifier is designed to provide dual-modality feedback: it functions as a strict gatekeeper offering **binary (0/1) validation**, while simultaneously calculating a **quantified score** (ratio of compliant positions) as a dense reward signal.

3. Corpus Alignment. Utilizing the Verifier, we constructed the **Structurally-Aligned Corpus** from open-source repositories via a **Best-Match Alignment** strategy. For each poem, specifically matching specific genres or *Cipai* patterns, we identified the template yielding the highest verification score. To ensure data quality, we filtered out samples falling below a threshold (score $< 80/100$), partitioning the remaining high-quality data into Train, Validation, and Test sets (see Table 4).

The establishment of this infrastructure completes the CCP-Arena. By standardizing both the verification metric and the training data, we provide an effective testbed specifically tailored for **multi-objective constrained creative generation**. This standardized environment allows for fair and rigorous benchmarking of LLMs' capabilities in complying with complex structural rules. To foster further research in this challenging domain and promote reproducibility, we commit to **open-sourcing** both our Metric-Verifier engine and the Structurally-Aligned Corpus upon publication.

3 Methodology

As illustrated in Figure 2, our Progressive Structural Internalization (PSI) framework unfolds in three coherent logical steps. We initiate with ***Structural Scaffolding via Explicit Cognitive Planning*** (Sec. 3.1), utilizing explicit thought traces to cognitively decouple planning from generation. This prepares the model for ***Structural Internalization via Cascaded RL*** (Sec. 3.2), which serves as the core optimization mechanism to progressively enforce constraints. Crucially, to guide this optimization, ***Holistic Reward Modeling for Structural-Semantic Alignment*** (Sec. 3.3) constructs the specific signal formulations, detailing how structural precision and semantic quality are quantified and integrated into the RL objective.

3.1 Structural Scaffolding

To instill preliminary structure-aware competency, we construct a supervised scaffolding that trans-

forms the raw model into a rule-aware planner through two strategic components:

Template-Based Alignment. To facilitate the internalization of structural rules, we integrate the digitized formats (described in Sec. 2.2) directly into the SFT process. By utilizing these templates as a "structural skeleton" for the training data, we explicitly guide the model to align its generation with hard constraints.

Reasoning Synthesis. To enhance long-horizon planning, we synthesize Chain-of-Thought (Wei et al., 2022) trajectories styled after the *Appreciation Dictionary of Tang and Song Poetry* via a teacher model. This grants our model a dedicated planning phase, as validated in Section 4.4.

We instantiate this architecture via Thought-Augmented Supervised Fine-Tuning (SFT) on the constructed dataset $\mathcal{D} = \{(x, T, z, y)\}$.

3.2 Structural Internalization

To solidify the SFT scaffolding, we employ a reinforcement learning framework that formulates rigorous constraint satisfaction as a hierarchical optimization problem. While specific reward functions are mathematically defined in 3.3, we conceptually decompose the objective into a logical cascade of $K = 4$ levels: $\mathcal{L}_1$ **Basic Compliance**, $\mathcal{L}_2$ **Structural Precision**, $\mathcal{L}_3$ **Phonological Constraints**, and $\mathcal{L}_4$ **Semantic Constraints**. This layered formulation mirrors the dependency chain of creative generation, preventing the optimization landscape from becoming chaotic.

To rigorously enforce this hierarchy, we design a **Fine-Grained Cascaded Gating Mechanism**. Within each level k, we aggregate a set of sub-constraints $\mathcal{S}_k$, where the reward signal for level k is active only if *all* sub-constraints in all preceding levels $j < k$ are perfectly satisfied. We define the binary satisfaction status for level j as $\Phi_j(y) = \prod_{i \in \mathcal{S}_j} I(r_{j,i}(y) = 1)$. Consequently, the total reward $R(y)$ is formulated as:

$$R(y) = \sum_{k=1}^{K} \underbrace{\left(\prod_{j=1}^{k-1} \Phi_j(y) \right)}_{\text{Gate}} \cdot \underbrace{\left(\sum_{i \in \mathcal{S}_k} r_{k,i}(y) \right)}_{\text{Level Score}} \quad (1)$$

The rationale behind this hard gating is grounded in the inherent dependencies of the generation process. If the model fails at $\mathcal{L}_1$ (e.g., by violating the thought format), the subsequent generation is fundamentally invalid and meaningless. Similarly, strict adherence to $\mathcal{L}_2$ (e.g., character count and line length) is a prerequisite for $\mathcal{L}_3$; if the line length is incorrect, it becomes infeasible to align or verify the rigid positional rules of tones and rhymes. This mathematical structure ensures that the model distributes its capacity logically.

Functionally, this design induces an **Implicit Curriculum Learning Effect** during training. By gating rewards, the model naturally prioritizes the mastery of fundamental constraints before tackling complex semantic objectives, aligning with established multi-task learning strategies that progress from simple to difficult sub-tasks (Parashar et al., 2025; Zhou et al., 2020).

We optimize the policy π_θ against this strict cascaded signal using **Group Relative Policy Optimization (GRPO)** (Shao et al., 2024). For each prompt q, we sample a group of outputs $\{y_1, \ldots, y_G\}$ from the old policy $\pi_{\theta_{old}}$. The advantage A_g for each sample is computed by normalizing the total reward $R(y_g)$ against the group statistics: $A_g = (R(y_g) - \text{mean}(R_{group}))/(\text{std}(R_{group}) + \epsilon)$. The objective function is defined as:

$$\mathcal{L}_{GRPO}(\theta) = E_{q \sim P(Q), y \sim \pi_{\theta_{old}}} \Bigg[\frac{1}{G} \sum_{g=1}^{G} \min \Bigg(\frac{\pi_\theta(y_g|q)}{\pi_{\theta_{old}}(y_g|q)} A_g, \text{clip}\Big(\frac{\pi_\theta(y_g|q)}{\pi_{\theta_{old}}(y_g|q)}, 1-\epsilon, 1+\epsilon \Big) A_g \Bigg) - \beta D_{KL}(\pi_\theta || \pi_{ref}) \Bigg] \quad (2)$$

3.3 Holistic Reward Modeling

To guide the cascaded optimization process, we construct a holistic reward function $R(y, z)$ that evaluates both the final response y and the reasoning trace z. We mathematically formulate the objectives into four hierarchical levels corresponding to the cascade stages defined in Sec. 3.2.

Level 1: Basic Compliance ($\mathcal{L}_1$). This level enforces fundamental validity. Since these are hard constraints, we model them as binary indicators to filter out invalid samples early.

1. **Language Purity (r_{lang}):** To prevent multilingual hallucination, we penalize tokens belonging to non-target languages.

2. **Thinking Format (r_{fmt}):** To ensure the structural integrity of the thought process, the output must contain exactly one pair of `<think>` and `</think>` tags.

The reward is formulated as:

$$r_{\mathcal{L}_1} = I(y \in \text{TargetLang}) + I(N_{\text{tags}} = 1) \quad (3)$$

where $I(\cdot)$ is the indicator function.

Level 2: Structural Precision ($\mathcal{L}_2$). This level quantifies the geometric correctness of the poem based on the specified template T. We adopt a *soft-penalty* formulation using an inverse distance function to provide dense gradient signals.

1. **Line Count (r_{line}):** Measures the deviation of the generated line count C_{gen} from the template requirement C_{tgt}.

2. **Character Count (r_{char}):** Measures the average deviation of character counts per line $len(l_i)$ against the template constraints len_i^*.

The combined reward for $\mathcal{L}_2$ is defined as:

$$r_{\mathcal{L}_2} = \underbrace{\frac{1}{1 + |C_{gen} - C_{tgt}|}}_{\text{Line Precision}} + \frac{1}{L}\sum_{i=1}^{L} \underbrace{\frac{1}{1 + |len(l_i) - len_i^*|}}_{\text{Char Precision}} \quad (4)$$

Level 3: Phonological Constraints ($\mathcal{L}_3$). This level evaluates the prosodic rules—the core of Chinese poetic aesthetics. We utilize a phonological dictionary Φ to map characters to their tonal and rhyming categories.

1. **Tonal Compliance (r_{tone}):** Calculates the ratio of characters matching the required Ping/Ze (Flat/Oblique) pattern based on the set of target positions M_{tone}.

2. **Rhyme Compliance (r_{rhyme}):** Calculates the ratio of line-ending characters in set M_{rhyme} that match the designated rhyme group G.

$$r_{\mathcal{L}_3} = \frac{\sum_{j \in M_{\text{tone}}} I(\Phi(y_j) = \tau_j)}{|M_{\text{tone}}|} + \frac{\sum_{k \in M_{\text{rhyme}}} I(\Phi(y_k) \in G)}{|M_{\text{rhyme}}|} \quad (5)$$

where τ_j is the target tone at position j.

Level 4: Semantic Constraints ($\mathcal{L}_4$). Since previous rewards ($\mathcal{L}_1$-$\mathcal{L}_3$) are input-agnostic, reliance solely on them risks **mode collapse**, where the model generates structurally perfect but semantically repetitive content. We introduce $\mathcal{L}_4$ to enforce semantic grounding and cognitive consistency.

1. Semantic Anchoring via ROUGE-L (r_{sem}): To prevent the model from drifting into nonsense while satisfying constraints, we use the ground truth reference y^* as a semantic anchor. We prioritize **Rule-based ROUGE-L**(Lin, 2004) over LLM-based judges to avoid *reward hacking*:

$$r_{\text{sem}} = \text{ROUGE-L}(y, y^*) = \frac{R_{lcs} P_{lcs}}{R_{lcs} + P_{lcs}} \quad (6)$$

Here, $LCS(y, y^*)$ denotes the length of the longest common subsequence between the generation and reference, $R_{lcs} = \frac{LCS(y,y^*)}{|y^*|}$ represents recall, and $P_{lcs} = \frac{LCS(y,y^*)}{|y|}$ represents precision.

2. Plan-Realization Alignment (r_{align}): To ensure the "thought" process z genuinely guides generation, we enforce consistency between the thought trace and the final poem. We calculate the proportion of generated poem lines $l_i \in y$ that logically originate from the thought trace z:

$$r_{\text{align}} = \frac{1}{|y|}\sum_{l_i \in y} I(l_i \subseteq z) \quad (7)$$

Here, $\subseteq$ denotes a substring match.

Finally, the total reward $R(y, z)$ for each training step is derived by aggregating these four level-specific signals using the dynamic weighting strategy formulated in Eq. 1.

4 Experiments

4.1 Experimental Settings

We initialize our training with DeepSeek-R1-0528-Qwen3-8B[2], a model further post-trained on Qwen3-8B (Yang et al., 2025) (a Transformer model comprising 36 layers and a hidden dimension of 4096) using distillation trajectories from DeepSeek-R1 (Guo et al., 2025). We selected this base model as it enhances the state-of-the-art foundation of Qwen3-8B by integrating superior reasoning capabilities derived from the DeepSeek-R1 distillation process. Our training pipeline employs `LLaMA-Factory` (Zheng et al., 2024) for Supervised Fine-Tuning (SFT) and `VeRL` (Sheng et al.,

[2]https://huggingface.co/deepseek-ai/DeepSeek-R1-0528-Qwen3-8B

Model	Regularized Verse				Ci				Speed↑
	Format↑	Tonal↑	Rhyme↑	Diversity↓	Format↑	Tonal↑	Rhyme↑	Diversity↓	(Token/s)
Base Model	75.62	10.88	33.00	34.73	12.21	0.23	2.77	47.35	46.65
Prompted	78.80	6.00	27.05	32.67	9.72	0.23	2.09	39.07	45.32
Constrained	*100.0*	*100.0*	*100.0*	30.00	*100.0*	*100.0*	*100.0*	47.91	**14.94**
NeuroLogic A*	*100.0*	*100.0*	*100.0*	37.82	*100.0*	*100.0*	*100.0*	48.21	**0.34**
Naive SFT	67.88	49.43	55.70	**22.18**	20.80	6.78	10.57	28.85	47.41
Naive RL	99.85	93.05	96.88	23.37	89.54	56.92	68.85	32.54	48.38
Ours (SFT)	84.10	52.70	62.28	22.45	46.81	18.71	20.69	29.46	46.19
Ours (RL)	**99.98**	**97.80**	**97.30**	22.44	**95.76**	**66.25**	**76.99**	**28.56**	45.05
DeepSeek-V3.2	77.70	36.08	64.38	27.42	68.51	24.14	48.73	35.94	–
+ Prompted	91.57	49.08	79.88	28.26	79.03	29.96	60.03	34.30	–
GPT-5.2	99.25	10.03	50.78	26.37	46.92	4.92	24.19	40.02	–
+ Prompted	97.45	54.15	96.38	23.90	70.60	30.19	64.50	32.70	–

Table 1: Automatic Evaluation Results. ↑ indicates higher is better, and ↓ indicates lower is better. The best results are highlighted in **bold**, and the second-best results are underlined. All speed measures are on the same GPUs

2024) for the subsequent Reinforcement Learning (RL) stage. Additional implementation details are provided in Appendix C.

We constructed five representative baselines adapting common format control methods to demonstrate the effectiveness of our approach: (1) **Base Model**, the vanilla DeepSeek-R1-0528-Qwen3-8B; (2) **Prompt Engineering** (Jie et al., 2024), which imposes constraints via natural language instructions; (3) **Constrained Decoding** (Deutsch et al., 2019), which forcibly masks the logits of illegal tokens during inference; (4) **NeuroLogic A*esque Decoding** (Lu et al., 2022), an advanced constrained decoding method with lookahead heuristics; (5) **Standard SFT**, trained on standard data; and (6) **Standard SFT+RL**, a traditional alignment pipeline. We also include **Ours (SFT)**, the intermediate checkpoint of our method, to perform a study on the impact of our RL stage. Additionally, we define the experimental upper bound by evaluating significantly larger SOTA General LLMs: **DeepSeek-V3.2** (DeepSeek-AI et al., 2025) and **GPT-5.2** (Singh et al., 2025). For a comprehensive comparison, we measured their performance in two distinct settings: providing the explicit structure template in the prompt versus excluding it.

4.2 Automatic Evaluation Results

Evaluation Metrics. Preliminary tests indicated that *LLM-as-a-judge* shows low correlation with human preference in poetry; thus, we rely exclusively on automatic metrics for structure and diversity. For structural correctness, we employ our custom verifier in a binary **"Strict Mode"**—awarding a score of 1 only for perfect constraint adherence (unlike the soft scoring used in training). For diversity, we report the geometric mean of **Self-BLEU-2/4** (Zhu et al., 2018), where lower scores indicate higher diversity. Moreover, the evaluation of content quality is presented in Sec. 4.3.

Performance Analysis. As shown in Table 1, our model significantly outperforms all baselines and larger closed-source SOTA models (excluding *Constrained Decoding*), empirically validating our PSI framework. Specifically, while prompted versions of *DeepSeek* and *GPT-5.2* show improvements over their base counterparts, they still fall short of our results, indicating that models often struggle with complex template understanding without specialized tuning—a gap our method effectively bridges. Notably, our lead widens on the more variable *Ci* task compared to *Regularized Verse*, underscoring our model's robustness in handling complex structural constraints.

Crucially, we achieve state-of-the-art diversity, confirming that strict compliance does not come at the cost of generation variety. Results further show that *Ours (SFT/RL)* consistently surpasses *Standard SFT/RL*, proving the efficacy of our *Structure Scaffolding* and *Initialization*. While *Constrained Decoding* forces 100% accuracy, it induces a 3× latency spike; meanwhile, *NeuroLogic A*esque Decoding* suffers from an even more extreme overhead—reaching up to 100× baseline latency—due to its lookahead heuristics. Furthermore, as verified in the subsequent Human Evaluation, these decoding-constrained methods often suffer from degraded semantic quality.

Impact of Data Frequency. To evaluate general-

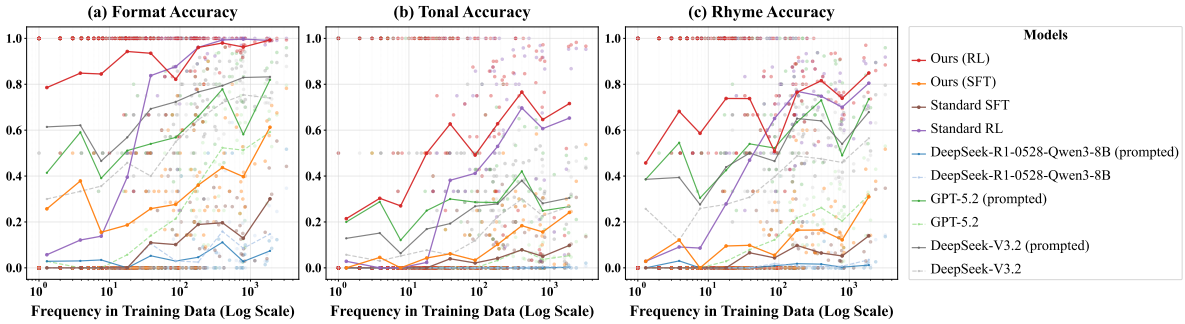

Figure 3: **Accuracy vs. Template Frequency.** Comparison of (a) Format, (b) Tonal, and (c) Rhyme accuracy across *Cipai* templates of varying training frequencies. we aggregated the results into 10 representative data points.

Model	Flu.	Coh.	Mean.	Aes.	Overall
Base Model	3.69	3.51	3.71	3.62	3.54
Prompted	3.47	3.35	3.47	3.33	3.28
Constrained	3.31	3.23	3.32	3.31	3.23
Standard RL	3.98	3.87	3.99	3.94	3.94
Ours	4.10	3.91	**4.06**	3.99	**4.04**
GPT-5.2	3.93	3.73	3.91	3.83	3.83
DeepSeek-V3.2	**4.17**	**3.93**	3.97	**4.04**	3.95

Table 2: Human Evaluation on Fluency, Coherence, Meaningfulness, Aesthetics, and Overall Score. Best scores are **bolded**, and second-best scores underlined. The Kendall's W is 0.42, indicating moderate inter-annotator agreement.

Model	Format↑	Tonal↑	Rhyme↑	Diversity↓
Dataset: Regularized Verse				
Our Method	**99.98**	**97.80**	**97.30**	**22.31**
w/o Cascaded	99.88	95.62	93.38	22.59
w/o Template	99.80	93.55	95.83	23.21
w/o Thought	99.90	96.73	96.70	23.19
Dataset: Ci				
Our Method	95.76	**66.25**	**76.99**	**28.56**
w/o Cascaded	90.73	60.99	65.18	29.70
w/o Template	82.76	52.12	66.42	30.10
w/o Thought	**98.98**	63.82	73.37	31.75

Table 3: Ablation study on different components.

ization, we plotted accuracy against *Cipai* training frequency in Figure 3. Our method dominates all baselines across the spectrum. While *Standard RL* approaches our performance on high-frequency (head) templates, its accuracy drops precipitously on rare (tail) ones. In contrast, our approach maintains high accuracy on long-tail data, demonstrating superior generalization to unseen or rare structures.

4.3 Human Evaluation Results

Given the limitations of current LLM-as-a-Judge paradigms in capturing poetic nuance, we conducted an expert human evaluation using five metrics adopted from (Yi et al., 2018): *Fluency*, *Coherence*, *Meaningfulness*, *Aesthetics*, and *Overall Score*, rated on a 0–5 scale. To maintain a balance between evaluation cost and depth, we focused on comparing our final model with key baselines, omitting intermediate SFT checkpoints. The original constrained decoding method was selected as a representative for constrained decoding approaches, including NeuroLogic A*. Our evaluation was conducted by six domain experts from the Department of Chinese Language and Literature, including Ph.D. candidates and senior undergraduates specializing in classical poetry. The process strictly adhered to ethical standards, ensuring informed consent and fair compensation for all participants. Detailed information regarding annotator demographics, ethical protocols, and inter-annotator agreement score calculations is available in the Appendix E. Notably, SOTA larger models (e.g., DeepSeek) were included as a "quality upper bound" and evaluated in a *template-free setting*, allowing them to demonstrate maximum creative potential without rigid formatting constraints.

The results are in Table 2. Despite its smaller scale, our model demonstrates performance parity with the much larger DeepSeek and significantly outperforms other baselines. Crucially, our model surpasses DeepSeek in *Meaningfulness* and *Overall Score*, suggesting that valid structural constraints serve as a cognitive scaffold that enhances, rather than hinders, semantic depth. Furthermore, the results highlight a distinct advantage of our approach: while Prompt Engineering and Constrained Decoding baselines incur a significant *alignment tax*—sacrificing literary quality to satisfy strict constraints—our **Structural Internalization** strategy eliminates this trade-off, achieving structural precision without compromising content aesthetics.

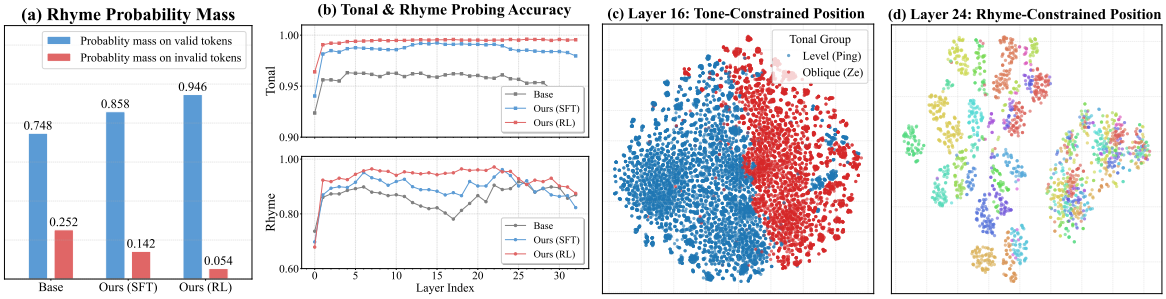

Figure 4: **Mechanistic Analysis of Structural Internalization.** (a): The evolution of probability mass assigned to valid vs. invalid tokens at constrained positions during training. (b): Layer-wise probing accuracy for structural features, showing deeper encoding of constraints from SFT to RL. (c) & (d): t-SNE visualization of hidden states where tonal categories show linear separability and rhyme groups form distinct clusters.

4.4 Ablation Studies

We performed an ablation study to verify the effectiveness of the Cascaded Gating Mechanism, the Template Guidance, and the Thought Process, with results shown in Table 3. Generally, removing any module impairs performance, particularly on the complex Ci task. Interestingly, removing the *Thought* process slightly improves Ci formatting but causes significant drops in Tonal and Rhyme scores. This suggests that while Templates sufficiently handle basic structure context, the Thought mechanism is vital for directing the model's attention toward harder phonetic constraints. Thus, despite a negligible formatting trade-off, the Thought module proves indispensable for ensuring both semantic depth and strict rhythmic compliance.

4.5 Mechanistic Analysis

To empirically verify that our framework essentially internalizes structural constraints into the model parameters rather than merely memorizing surface patterns, we conducted a comprehensive "outside-in" investigation.

Probability Mass Distribution. First, we analyzed the model's output behavior at positions governed by specific strict constraints (i.e., tonal and rhyme positions). We aggregated the probability mass allocated to valid tokens (those satisfying the constraint) versus invalid tokens. As illustrated in Figure 4(a) and Figure 9, we observe a clear trend: throughout the transition from SFT to RL, the probability mass assigned to valid tokens steadily increases, while the mass for invalid tokens diminishes. This shift indicates that the model becomes increasingly confident and precise in adhering to structural rules during training.

Linear Probing of Layer-wise Representations. Moving deeper into the model, we trained linear probes on the hidden states of each layer to quantify the amount of structural information encoded at different depths. The results in Figure 4(b) show that as training progresses from SFT to RL, the decodability of structural features (tonal and rhyme categories) improves consistently across all layers. This serves as compelling evidence for the success of our **Progressive Structural Internalization** framework, demonstrating that the structural constraints are not just enforced at the final output layer but are deeply embedded throughout the model's internal representations.

Visualization of Hidden States. Finally, to provide a qualitative perspective, we employed t-SNE to visualize representations from selected intermediate layers. As shown in Figure 4 (c) & (d), the visualization reveals striking geometric patterns: (1) The representations of characters with flat (*Ping*) vs. oblique (*Ze*) tones exhibit clear **linear separability**; (2) Characters belonging to the same rhyme group spontaneously form distinct, cohesive **clusters**. These geometric structures confirm that the model has successfully learned to map symbolic structural constraints into its latent semantic space.

4.6 Case Study

In the qualitative cases in Figure 5, we observe that general-purpose baselines (Base, Prompted, and DeepSeek) struggle with **structural adherence**, exhibiting frequent tonal/rhyme violations and even length hallucinations (e.g., redundant lines in the Base model) despite explicit prompts. Conversely, while the Constrained baseline guarantees structural correctness, it suffers from **semantic degradation**, evident in the degenerate repetition of the title rather than initiating a coherent narrative. This is also evident in **"forced rhyming"** artifacts—such as the unnatural insertion of "Qin Shang" (zither

Figure 5: **Qualitative comparison of genereted pieces.** Red characters denote tonal, rhyme, or grammatical errors.

and cup)—which exposes the intrinsic limitation of rigid decoding mechanisms: sacrificing contextual coherence to strictly satisfy rhyme schemes. In contrast, our model achieves a **harmonious unification** of form and content: it perfectly fits the rigid structural constraints while maintaining semantic fluency and vivid imagery, notably capturing the authentic linguistic rhythm and stylistic ethos of the classic genre in the final verses.

5 Conclusion

In this paper, we address the intrinsic conflict between strict structural constraints and creative freedom in LLMs through the lens of Chinese Classical Poetry. We establish **CCP-Arena**, a robust infrastructure for constrained generation, and propose the **Progressive Structural Internalization (PSI)** framework. By integrating Structural Scaffolding with Cascaded Reinforcement Learning, PSI transforms external rules into the model's intrinsic intuition. Extensive experiments demonstrate that our approach achieves state-of-the-art performance, paving the way for future research on complex, multi-objective creative generation tasks.

Limitations

Despite its effectiveness, our framework faces two limitations. First, the application is limited by the lack of robust evaluation infrastructure for other languages. Since tools for quantifying meter and rhyme in forms like English Sonnets are underdeveloped, we focused on Chinese forms but plan to expand to multilingual contexts as better scansion algorithms become available. Second, we have yet to incorporate LLM-as-a-Judge for semantic guidance. We found that current LLM judges suffer from low accuracy in creative evaluation and are prone to reward hacking. We plan to incorporate such methods to enhance semantic constraints once these evaluation systems perform better.

Ethical Considerations

Our research is motivated by a commitment to the revitalization of classical literary forms through modern artificial intelligence. Our framework automates complex constraints to lower technical barriers, serving as a "digital muse" rather than replacing human artistry. By managing rigid structural logic, the AI empowers users to prioritize semantic depth and emotion in Human-AI co-creation. Responsibly applied, this technology democratizes access to cultural heritage, ensuring it enriches rather than dilutes traditional aesthetics.

Potential risks regarding educational utility must be acknowledged. While PSI serves as a powerful assistive tool, over-reliance on such automated generation systems in educational settings might diminish the learning curve for students seeking to master the complex prosodic rules of Classical Chinese Poetry manually. To mitigate this risk, we advocate for a 'human-in-the-loop' paradigm where the model is positioned as a creative co-pilot rather than a substitute. By using the model primarily for brainstorming imagery or verifying rigid tonal patterns, users can maintain their creative agency while leveraging AI to overcome technical bottlenecks.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No.T2341003), National Natural Science Foundation of China (No. 62236011) and a grant from the Guoqiang Institute, Tsinghua University.

References

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. 2024. Guiding llms the right way: fast, non-invasive constrained generation. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Bradley Butcher, Michael O'Keefe, and James Titchener. 2025. Precise length control for large language models. *Natural Language Processing Journal*, 11:100143.

Huimin Chen, Xiaoyuan Yi, Maosong Sun, Wenhao Li, Cheng Yang, and Zhipeng Guo. 2019. Sentiment-controllable chinese poetry generation. In *IJCAI*, pages 4925–4931.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenhao Xu, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, and 245 others. 2025. Deepseek-v3.2: Pushing the frontier of open large language models. *Preprint*, arXiv:2512.02556.

Daniel Deutsch, Shyam Upadhyay, and Dan Roth. 2019. A general-purpose algorithm for constrained sequential inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 482–492, Hong Kong, China. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 175 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638.

Senyu Han, Lu Chen, Li-Min Lin, Zhengshan Xu, and Kai Yu. 2024. Ibsen: Director-actor agent collaboration for controllable and interactive drama script generation. *Preprint*, arXiv:2407.01093.

Renlong Jie, Xiaojun Meng, Lifeng Shang, Xin Jiang, and Qun Liu. 2024. Prompt-based length controlled generation with multiple control types. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1067–1085, Bangkok, Thailand. Association for Computational Linguistics.

Jiaming Li, Lei Zhang, Yunshui Li, Ziqiang Liu, Yuelin Bai, Run Luo, Longze Chen, and Min Yang. 2024. Ruler: A model-agnostic method to control generated length for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3042–3059, Miami, Florida, USA. Association for Computational Linguistics.

Piji Li, Haisong Zhang, Xiaojiang Liu, and Shuming Shi. 2020. Rigid formats controlled text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 742–751, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Ximing Lu, Sean Welleck, Peter West, Liwei Jiang, Jungo Kasai, Daniel Khashabi, Ronan Le Bras, Lianhui Qin, Youngjae Yu, Rowan Zellers, Noah A. Smith, and Yejin Choi. 2022. NeuroLogic a*esque decoding: Constrained text generation with lookahead heuristics. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 780–799, Seattle, United States. Association for Computational Linguistics.

Yingfeng Luo, Changliang Li, Canan Huang, Chen Xu, Xin Zeng, Binghao Wei, Tong Xiao, and Jingbo Zhu. 2021. Chinese poetry generation with metrical constraints. In *Natural Language Processing and Chinese Computing: 10th CCF International Conference, NLPCC 2021, Qingdao, China, October 13–17, 2021, Proceedings, Part I*, page 377–388, Berlin, Heidelberg. Springer-Verlag.

Xichu Ma, Ye Wang, Min-Yen Kan, and Wee Sun Lee. 2021. Ai-lyricist: Generating music and vocabulary constrained lyrics. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, page 1002–1011, New York, NY, USA. Association for Computing Machinery.

Niels Mündler, Jingxuan He, Hao Wang, Koushik Sen, Dawn Song, and Martin Vechev. 2025. Type-constrained code generation with language models. *Proceedings of the ACM on Programming Languages*, 9(PLDI):601–626.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Shubham Parashar, Shurui Gui, Xiner Li, Hongyi Ling, Sushil Vemuri, Blake Olson, Eric Li, Yu Zhang, James Caverlee, Dileep Kalathil, and Shuiwang Ji. 2025. Curriculum reinforcement learning from easy to hard tasks improves llm reasoning. *Preprint*, arXiv:2506.06632.

Nanyun Peng. 2022. Controllable text generation for open-domain creativity and fairness. *Preprint*, arXiv:2209.12099.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Connor Shorten, Charles Pierse, Thomas Benjamin Smith, Erika Cardenas, Akanksha Sharma, John Trengrove, and Bob van Luijt. 2024. Structuredrag: Json response formatting with large language models. *Preprint*, arXiv:2408.11061.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 465 others. 2025. Openai gpt-5 system card. *Preprint*, arXiv:2601.03267.

Seoha Song, Junhyun Lee, and Hyeonmok Ko. 2025. Hansel: output length controlling framework for large language models. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press.

Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. 2024. Let me speak freely? a study on the impact of format restrictions on large language model performance. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1218–1236, Miami, Florida, US. Association for Computational Linguistics.

Yufei Tian and Nanyun Peng. 2022. Zero-shot sonnet generation with discourse-level planning and aesthetics features. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3587–3597, Seattle, United States. Association for Computational Linguistics.

Guanghui Wang, Jinze Yu, Xing Zhang, Dayuan Jiang, Yin Song, Tomal Deb, Xuefeng Liu, and Peiyang He. 2025a. Sted and consistency scoring: A framework for evaluating llm structured output reliability. *Preprint*, arXiv:2512.23712.

Zhaoyang Wang, Jinqi Jiang, Huichi Zhou, Wenhao Zheng, Xuchao Zhang, Chetan Bansal, and Huaxiu Yao. 2025b. Verifiable format control for large language model generations. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3499–3513, Albuquerque, New Mexico. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Jin, and Fei Huang. 2025. Writingbench: A comprehensive benchmark for generative writing. *Preprint*, arXiv:2503.05244.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Cheng Yang, Maosong Sun, Xiaoyuan Yi, and Wenhao Li. 2018. Stylistic Chinese poetry generation via unsupervised style disentanglement. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3960–3969, Brussels, Belgium. Association for Computational Linguistics.

Xiaoyuan Yi, Maosong Sun, Ruoyu Li, and Wenhao Li. 2018. Automatic poetry generation with mutual reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3143–3153, Brussels, Belgium. Association for Computational Linguistics.

Chengyue Yu, Lei Zang, Jiaotuan Wang, Chenyi Zhuang, and Jinjie Gu. 2024. CharPoet: A Chinese classical poetry generation system based on token-free LLM. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 315–325, Bangkok, Thailand. Association for Computational Linguistics.

Rongsheng Zhang, Xiaoxi Mao, Le Li, Lin Jiang, Lin Chen, Zhiwei Hu, Yadong Xi, Changjie Fan, and Minlie Huang. 2020. Youling: an AI-assisted lyrics creation system. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 85–91, Online. Association for Computational Linguistics.

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

Guo Zhipeng, Xiaoyuan Yi, Maosong Sun, Wenhao Li, Cheng Yang, Jiannan Liang, Huimin Chen, Yuhui Zhang, and Ruoyu Li. 2019. Jiuge: A human-machine collaborative Chinese classical poetry generation system. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 25–30, Florence, Italy. Association for Computational Linguistics.

Tianyi Zhou, Shengjie Wang, and Jeffrey Bilmes. 2020. Curriculum learning by dynamic instance hardness. In *Advances in Neural Information Processing Systems*, volume 33, pages 8602–8613. Curran Associates, Inc.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. *SIGIR*.

Yutao Zhu, Ruihua Song, Zhicheng Dou, Jian-Yun Nie, and Jin Zhou. 2020. ScriptWriter: Narrative-guided script generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8647–8657, Online. Association for Computational Linguistics.

A Related Work

A.1 Format-Constrained Generation

Prior research on format-constrained generation has primarily focused on enforcing output schemas, such as generating valid JSON objects or code snippets. This line of work generally follows two main paradigms. The first is *Prompt Engineering*, which aims to boost adherence rates through carefully designed instructions and in-context examples (Shorten et al., 2024; Tam et al., 2024). The second involves *Constrained Decoding*, which guarantees syntactic validity by pruning the vocabulary space during inference (Mündler et al., 2025; Wang et al., 2025a). Strategies to enhance the efficiency of these decoding algorithms have also been explored (Beurer-Kellner et al., 2024). Furthermore, data-driven approaches have been proposed, utilizing large-scale benchmarks to fine-tune models specifically for structural compliance (Wang et al., 2025b). However, imposing strict constraints is not without drawbacks. (Tam et al., 2024) observed that constrained decoding can significantly degrade the model's reasoning capabilities and generation quality—a trade-off between strict adherence and semantic performance that aligns with the empirical findings in our own study.

Parallel to schema constraints, another stream of research targets length control as a single-dimensional constraint. Techniques in this area vary from modifying model architectures (Butcher et al., 2025; Song et al., 2025) and continual pre-training strategies (Li et al., 2024) to manipulating sampling mechanisms (Jie et al., 2024). However, the format-constrained creative text generation targeted in this paper presents a far more intricate challenge rather than simple length manipulation. Creative tasks involve multi-dimensional constraints, such as rhyme schemes, tonal patterns, and rigid sentence structures. Naively applying existing decoding restrictions to these complex scenarios often leads to a collapse in creative quality. Consequently, there is a need for advanced methodologies that can enforce rigorous constraints without compromising the aesthetic and semantic integrity of the generated text.

In the specific domain of formatted creative generation, SongNet (Li et al., 2020) represents a pioneering effort. While it achieved commendable length control, it relied on introducing rigid, task-specific modules into the neural architecture. Such an approach, however, faces significant limitations in the current era of Large Language Models (LLMs). Even when model weights are accessible (e.g., Qwen, Llama), modern deployment and fine-tuning paradigms rely on standardized, often frozen architectures to preserve pre-trained knowledge. Invasive structural modifications, such as inserting new layers or altering internal mechanisms as required by SongNet, are impractical and often detrimental to the general capabilities of foundational models. The field, therefore, calls for a more robust and compatible framework tailored to modern LLMs—one that can achieve high-precision format control based on existing weights without necessitating complex architectural redesigns.

A.2 Creative Text Generation

Creative text generation stands as a cornerstone in natural language processing (NLP), serving as a critical benchmark for evaluating the creativity and generalization capabilities of Large Language Models (LLMs) (Peng, 2022; Wu et al., 2025). Early research in this domain primarily focused on *short-form interactions*, such as generating poetry, lyrics, and short stories (Yi et al., 2018; Fan et al., 2018). These tasks required models to balance linguistic fluency with imaginative divergence.

With the emergence of scaling laws and the advanced capabilities of modern LLMs, the field has progressively expanded toward *long-form narratives*, including scriptwriting and novel generation (Zhu et al., 2020; Han et al., 2024). Beyond measuring generation quality and raw creativity, researchers have traditionally focused on controlling specific attributes such as sentiment (Chen et al., 2019), stylistic consistency (Yang et al., 2018), and structural formatting (Li et al., 2020).

A significant paradigm shift occurred with the advent of instruction tuning. Modern LLMs can now effectively handle **high-level semantic constraints**—such as stylistic imitation or sentiment transfer—simply through natural language prompting (Ouyang et al., 2022). However, precise control over **rigid structural constraints** remains a persistent challenge. Attributes like exact length, specific formatting, and rhyme schemes are frequently compromised. This limitation is largely inherent to the architecture of LLMs: the widespread use of subword tokenization algorithms (e.g., BPE) obscures character-level and phonetic information, making it difficult for models to "perceive" and adhere to strict metrical or syllabic rules during autoregressive generation (Yu et al., 2024).

While prior works attempted to enforce such constraints, they often relied on external plug-in modules or specialized decoding strategies (e.g., constrained beam search) that operate outside the standard pretraining-finetuning paradigm (Li et al., 2020; Luo et al., 2021). These approaches are often computationally expensive or difficult to integrate seamlessly with massive, general-purpose LLMs. This creates a notable gap: the need for a method that internalizes structural constraints directly into the LLM without relying on external scaffolding. Our work aims to bridge this gap, leveraging a unified training framework to harmonize creative freedom with rigorous structural precision.

B Data Statistics

Table 4 summarizes the statistics of our structurally-aligned dataset. The dataset consists of two primary sources: *CI* and *Regularized Verse*. The data is partitioned into training, validation, and testing sets, with the training set containing a total of 187,806 samples. Specifically, the *Regularized Verse* subset constitutes the majority of the corpus across all splits. Our collected data is from open-source communities that do not contains any information that names or uniquely identifies individual people. The raw poetry data was sourced from the GitHub repository [https://github.com/Werneror/Poetry]. This repository is licensed under the MIT License, permitting its use for academic research. The poems are all Chinese.

C Additional Implementation Details

C.1 Hardware and Computational Budget

All experiments were conducted on NVIDIA A100 (40G) GPUs. In terms of computational cost, a single Supervised Fine-Tuning (SFT) session requires approximately 64 GPU-hours, while a single Reinforcement Learning (RL) training session takes about 80 GPU-hours. The total computational budget for this project is estimated to be around 5,000 GPU-hours. Following standard evaluation protocols, we report the performance of a single experimental run, averaged across the test set.

C.2 Hyperparameter Settings

We performed a grid search for hyperparameter tuning, specifically focusing on the learning rate within $\{1\text{e-}5, 2\text{e-}5, 5\text{e-}5, 1\text{e-}4, 2\text{e-}4\}$ and the per-device batch size within $\{4, 8, 16\}$.

SFT Configuration. Based on the search results, the optimal hyperparameters for SFT training were set as follows: `learning_rate` = 1.0e-4, `per_device_train_batch_size` = 4, and `gradient_accumulation_steps` = 4. For generation tasks during inference, we utilized nucleus sampling with a `temperature` of 0.7 and `top_p` of 0.95.

RL Configuration. For the Reinforcement Learning stage, we employed the PPO algorithm. The detailed hyperparameters, including rollout settings and KL penalty coefficients, are listed in Table 5.

D Additional Cost-Benefit Analysis

In this section, we provide a detailed analysis of the computational costs associated with the training pipeline of PSI and discuss the economic trade-offs compared to inference-time constrained decoding methods.

D.1 Training Efficiency

Despite the multi-stage nature of the PSI pipeline, the total computational overhead remains well within practical limits for standard research and production environments. The training process consists of two primary phases:

- **Supervised Fine-Tuning (SFT) Phase:** This stage requires approximately 8 hours of computation.
- **Reinforcement Learning (RL) Phase:** The alignment stage takes approximately 10 hours.

The entire end-to-end pipeline can be completed in less than 24 hours using a single server equipped with NVIDIA A800 GPUs. This efficiency demonstrates that our method does not require prohibitively expensive hardware clusters.

D.2 The Training-Inference Trade-off

The core design philosophy of PSI is the strategic shift of computational complexity from *inference time* to *training time*.

One-time Training Investment: The complexity of the pipeline—incorporating structural constraints and RL optimization—is a one-time investment. Once the model is converged, these constraints are "baked" into the model's parametric knowledge.

Split	Ci	Regularized Verse	Total
Train	33,890	153,916	187,806
Validation	1,846	6,408	8,254
Test	1,769	4,000	5,769

Table 4: Statistics of the structurally-aligned dataset across training, validation, and testing splits.

Table 5: Hyperparameter settings for Reinforcement Learning training.

Hyperparameter	Value
General Training	
Global Train Batch Size	64
Max Prompt Length	512
Max Response Length	2048
Actor Optimization (PPO)	
Actor Learning Rate	2e-5
PPO Mini Batch Size	64
PPO Micro Batch Size (per GPU)	1
Use Dynamic Batch Size	True
Max Token Length (per GPU)	8000
Rollout Configuration	
Number of Rollouts (N)	4
Rollout Temperature	0.8
Rollout Hardware Engine	vLLM (bfloat16)
GPU Memory Utilization	0.7
KL Penalty & Loss	
Use KL Loss	True
KL Loss Coefficient	0.001
KL Loss Type	Low Variance KL
KL Controller Coefficient	0.001
System & Other	
Gradient Accumulation / Offload	True (Param Offload)
Critic Warmup Steps	0

Recurring Inference Savings: In contrast to search-based or constrained decoding methods, which incur significant overhead for every generated token, our model performs standard autoregressive generation. By eliminating the need for real-time constraint-checking or complex beam search during deployment, PSI significantly reduces inference latency and serving costs.

D.3 Real-world Deployment Advantage

For large-scale applications where the model is queried millions of times, the cumulative savings in computational resources are substantial. While constrained decoding methods often suffer from throughput bottlenecks, PSI maintains the same inference speed as standard LLMs, providing a superior user experience. We argue that spending moderate offline resources during the training phase to achieve massive gains in online efficiency is a highly favorable trade-off for real-world LLM deployment.

E Detailed Information on Human Evaluation

To assess the generation quality, we conducted a rigorous human evaluation using a subset of 64 distinct inputs randomly sampled from our test set, consisting of **32 Regularized Verses** and **32 Ci** poems. For each input, annotators evaluated the outputs of seven different models, resulting in a total of **448 poems** being assessed. To ensure a fair comparison, all samples were anonymized, and the outputs from different models were shuffled to prevent potential bias.

E.1 Annotator Expertise and Methodology

We recruited six domain experts from the Department of Chinese Language and Literature at our university, prioritizing specialized expertise over a larger pool of non-expert contributors. The group consists of **four Ph.D. candidates and two senior undergraduates (incoming Ph.D. students)**, all of whom possess extensive backgrounds in classical Chinese literature and prior research experience in classical poetry. Their academic background ensures a critical and professional assessment of prosody and aesthetics. The evaluation set was divided into two batches, and each batch was assigned to three independent annotators. After collecting the responses, we restored the order to compute the final scores.

E.2 Ethics, Consent, and Compensation

We strictly adhered to ethical guidelines throughout the study. Regarding the ethical review process, we clarify that our university/country does not currently mandate a formal Institutional Review Board (IRB) process for non-medical linguistic annotation tasks; however, we have strictly followed standard ethical principles regarding privacy, consent, and fair treatment.

- **Compensation:** All annotators were financially compensated at a rate significantly higher than the standard part-time hourly wage at our institution.
- **Informed Consent:** We obtained written informed consent from all participants, explicitly disclosing that their annotations would be used for AI model evaluation and academic publication.

E.3 Reliability and Validation

We believe this compact expert group provides highly consistent and statistically meaningful judgments for such a specialized task. To further validate our findings and address the inherent subjectivity of poetry evaluation, we emphasize that our human scores are **strongly corroborated by the Automatic Evaluation (Table 1)**. The objective metrics show consistent performance gains that align with the human assessments, thereby cross-validating the reliability of our results and confirming the superiority of our method.

The detailed instructions provided to the annotators are listed in Table 6, and the results categorized by genre are presented in Table 6.

E.4 Inter-Annotator Agreement (IAA)

Since poetry scoring is a highly subjective task, different annotators may naturally possess varying baselines of "strictness" (i.e., systematic bias). Consequently, relying on absolute score agreement metrics, such as Krippendorff's α, may understate the true consistency among evaluators. Instead, we utilized Kendall's Coefficient of Concordance (Kendall's W), which is a widely accepted and standard non-parametric statistic for assessing inter-rater agreement among ordinal raters (ranking).

Calculation Method Specifically, for each input, we converted the absolute scores given by the 6 annotators into relative rankings across the 7 evaluated models. We then calculated Kendall's W based on these rank lists to measure the extent to which the annotators agreed on the relative ordering of model quality.

Result The calculated Kendall's W for our human evaluation is 0.42, indicating a moderate level of agreement. Given the extremely high subjectivity inherent in creative poetry evaluation, this result confirms that despite individual differences in aesthetic taste, the expert annotators generally agreed on the relative quality of the models.

F Poem Translations

For the English translations of the poems presented in Figure 1, please refer to Figure 7 and Figure 8. These translations are provided to facilitate a better understanding of the thematic and stylistic nuances captured by our model.

G Additional Mechanistic Analysis

To complement the discussion in the main text, we provide further mechanistic analysis in Figure 9.

H Additional Case Studies

We provide additional examples of generated poems in Figure 10. This case showcases the model's performance across various genres and themes, further illustrating its creative capabilities and linguistic precision.

I Declaration of AI Use

We strictly limited the use of Artificial Intelligence (AI) tools in the preparation of this manuscript to the following auxiliary tasks:

1. **Linguistic Polishing**: AI tools were used to improve English phrasing and grammar.
2. **Visual Refinement**: AI assisted in refining the visual presentation of conceptual, non-experimental illustrations (specifically Figures 1, 2, 7, and 8).
3. **Coding Assistance**: AI was utilized to assist in writing auxiliary code scripts.

We confirm that no AI tools were used to generate the scientific ideas, experimental data, or the results presented in this work. All textual content and scientific claims were verified by the authors.

Instructions for Human Evaluators

Thank you for your contribution. This survey aims to evaluate the capability of models in generating classical Chinese poetry. Please adhere to the following guidelines:

1. **Scoring Strategy (Relative Comparison):** Rate each poem on an integer scale of **0 to 5**. The core principle is "horizontal comparison": ensure that score differences within the same row directly reflect the quality gap between candidates.
2. **Input Format:** Enter the score directly after the colon for each metric.
3. **Model Anonymity:** The order of models is **randomly shuffled** for each row. The same column index across different rows *does not* correspond to the same model.
4. **Content Focus:** Please ignore minor generation artifacts (e.g., residual tags) and focus solely on the poetic content.
5. **Reference:** The first column displays the original poem (Ground Truth) corresponding to the instruction. Use this as your gold standard for reference.
6. **View Settings:** If the text exceeds the window or cell height (in the spreadsheet), please adjust the zoom level or row height to ensure all content is visible before rating.
7. **Metric Definitions:**
   - **Fluency:** Is the language natural and smooth? Assess for grammatical errors, made-up words, or awkward phrasing.
   - **Coherence:** Is the logical flow and artistic conception consistent? Check for disjointed imagery or contextual gaps.
   - **Meaningfulness:** Does the poem convey rich substance and emotion? Avoid hollow rhetoric or meaningless repetition.
   - **Aesthetics:** Does the poem possess literary beauty? Evaluate the elegance of diction and the immersive quality of the imagery.
   - **Overall Score:** A holistic impression quality score based on the above dimensions.

Figure 6: The detailed instructions provided to human annotators for evaluating the generated poems. The evaluation interface is based on a spreadsheet format to facilitate side-by-side comparison.

Figure 7: Poem Translations of the Regularized Verse in Figure 1

Model	Regularized Verse					Ci					Global Average				
	Flu.	Coh.	Mean.	Aes.	All	Flu.	Coh.	Mean.	Aes.	All	Flu.	Coh.	Mean.	Aes.	All
Base Model	3.88	3.66	3.89	3.77	3.74	3.51	3.35	3.53	3.48	3.35	3.69	3.51	3.71	3.62	3.54
Prompt	3.61	3.42	3.59	3.40	3.41	3.32	3.28	3.34	3.26	3.15	3.47	3.35	3.47	3.33	3.28
Constrained	3.92	3.71	3.73	3.81	3.71	2.72	2.76	2.93	2.80	2.75	3.31	3.23	3.32	3.31	3.23
Naive RL	4.03	3.84	3.94	3.84	3.91	3.93	3.90	4.04	4.03	3.97	3.98	3.87	3.99	3.94	3.94
Ours	4.03	3.84	**4.00**	3.88	**3.96**	4.17	**3.97**	**4.12**	**4.09**	**4.12**	4.10	3.91	**4.06**	3.99	**4.04**
GPT-5.2	3.89	3.66	3.81	3.71	3.77	3.96	3.80	4.01	3.95	3.89	3.93	3.73	3.91	3.83	3.83
Deepseek	**4.16**	**3.93**	3.95	**4.02**	3.85	**4.18**	3.94	3.99	4.06	4.04	**4.17**	**3.93**	3.97	**4.04**	3.95

Table 6: Performance evaluation across different generation tasks. Scores encompass Fluency (Flu.), Coherence (Coh.), Meaningfulness (Mean.), Aesthetics (Aes.), and Overall Score (All). The best scores are **bolded**, and the second-best scores are underlined.

Figure 8: Poem Translations of the Ci in Figure 1

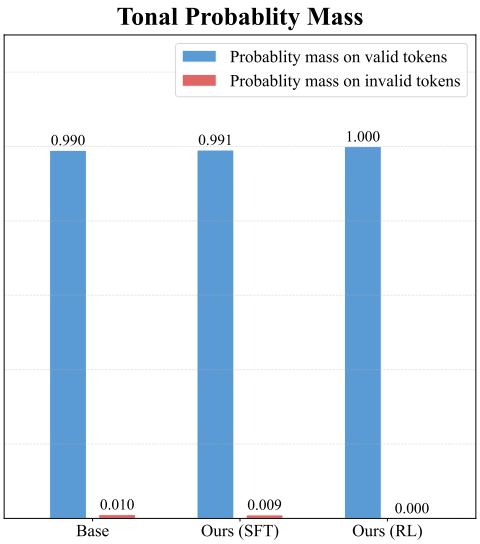

Figure 9: The evolution of probability mass assigned to valid vs. invalid tokens at constrained positions during training.

Base Model

虚檐梧影落秋光*，
◎●○○●●○
Parasol shadows fall in autumn light,
络纬声中夜气凉*。
⊙○◎●●○○
Night air cools amidst cricket songs.
风过空山惊客梦，
⊙○◎●○○●
Mountain winds startle a traveler's dream,
月移松籁入诗肠*。
◎●○○●●○
Moonlight and pine sighs inspire the soul.

Prompted

虚檐梧影月摇摇*，
◎●○○●●○
Shadows sway under eaves in moonlight,
络纬轻声入夜宵*。
⊙○◎●●○○
Soft cricket chirps drift through the night.
寒光透壁秋光老，
⊙○◎●○○●
Cold light pierces walls as autumn fades,
独对苍山月色遥*。
◎●○○●●○
Alone facing peaks and distant moon.

Constrained

虚槛梧阴月影斜*，
◎●○○●●○
Slanted moonbeams through parasol shade,
络丝声里夜凉赊*。
⊙○◎●●○○
The night's chill grows with cricket cries.
欲将秋思随云去，
⊙○◎●○○●
Wishing autumn thoughts away with clouds,
寒色凝成万壑霞*。
◎●○○●●○
Cold mist gathers in the deep valleys.

Ours

夜色秋光滟不欢*，
◎●○○●●○
Dull night and joyless autumn light,
一声络纬倚栏干*。
⊙○◎●●○○
A lone cricket chirps by the rail.
虚檐弄碎梧桐影，
⊙○◎●○○●
Eaves break the parasol tree shadows,
满地清寒月露漙*。
◎●○○●●○
Heavy dew and cold frost mask the ground.

Deepseek

虚檐人静夜光幽*，
◎●○○●●○
Quiet eaves under a dim night glow,
络纬啼残一叶秋*。
⊙○◎●●○○
Crickets cry for the falling leaves.
十二阑干梧影瘦，
⊙○◎●○○●
Thin shadows lean against the rails,
月明何处不关愁*。
◎●○○●●○
Nowhere is free of sorrow under this moon.

Figure 10: **Extra case of qualitative comparison of genereted pieces.** Red characters denote tonal, rhyme, or grammatical errors.