Evolve Wisely: Decomposing Genetic Algorithms for Chinese Ci Poetry Generation

Anonymous ACL submission

Abstract

Chinese Ci poetry generation requires balancing strict prosodic constraints with aesthetic quality. We decompose Genetic Algorithms (GA) into **selection** and **evolution** components, evaluating across 3 LLMs, 8 Cipai formats, and 48 prompts (144 configurations, 5,760 poems). Our findings reveal that GA is highly effective: multi-sampling with selection achieves +19.2% improvement over zero-shot. Decomposition analysis shows that selection accounts for 96.3% of performance gain, serving as a powerful "drafting" phase. The "revision" phase (evolution) is where methods diverge. LLM-guided operators achieve 40%+ success rates on capable models (DeepSeek: 43.8%, GPT-5.1: 41.7%), significantly outperforming the blind mutation baseline (character-level: ∼25%, $p < 0.01$). While mechanical operators actively degrade quality, semantic-aware evolution successfully simulates a human poet's intelligent revision process. Our results suggest a strategy to **"evolve wisely"**: rely on selection for robust baselines, and reserve expensive semantic evolution for capable models to achieve quality breakthroughs.

1 Introduction

Chinese Ci poetry (*SongCi*), a crown jewel of classical Chinese literature, requires a delicate balance between strict prosodic constraints (schema, tonal patterns, rhyme) and aesthetic quality (Li et al., 2020; Song, 2022). Generating high-quality Ci poetry is thus a quintessential problem in **Constrained Creative Text Generation** (Zhang and Lapata, 2014; Yan, 2016). While Large Language Models (LLMs) have revolutionized natural language generation, they often struggle with this dual constraint (Hu et al., 2024), either generating fluent text that violates rules or failing to strictly adhere to character-count or tonal requirements due to their token-based nature (Yu et al., 2024; Qu et al., 2025).

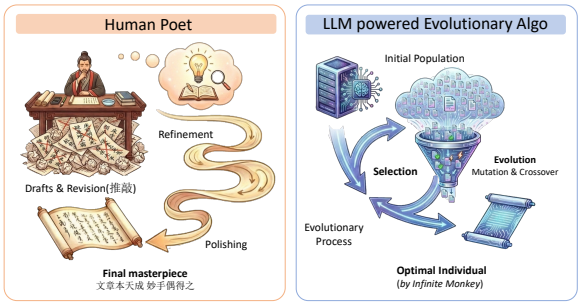

Figure 1: Human poet's workflow versus LLM powered evolutionary algorithm.

Genetic Algorithms (GA) offer a promising solution by simulating the evolution process. However, in the context of poetry, this biological metaphor is better understood through the lens of a **poet's workflow**: *drafting* (generating multiple initial ideas) and *polishing* (iteratively refining words and sentences). Recent work has integrated GA with LLMs (Lehman et al., 2024; Meyerson et al., 2024), but the precise mechanism remains under-explored. It is unclear whether the benefit stems from the evolutionary operators (the "polishing") or simply from the act of generating multiple candidates and selecting the best (the "drafting"). Without decomposing these factors, there is a risk of attributing benefits to complex evolutionary mechanics when simpler rejection sampling might suffice.

To address this ambiguity, we conduct the first systematic decomposition of GA for LLM-based text generation. We isolate the contribution of **selection** (multi-sampling with fitness ranking) from **evolution** (crossover and mutation operators). Our analysis introduces a

success rate metric to measure the probability of improvement (breakthrough), distinguishing it from degradation.

Our findings reveal a clear distinction between **blind mutation** and **semantic evolution**. While selection provides the initial +19% performance gain (the "drafting" benefit), traditional evolutionary operators (character/word-level) act as "blind polishing," often degrading the poem's quality. In contrast, LLM-guided operators serve as "intelligent polishing," achieving **40%+ success rates** on capable models. This signifies a **semantic breakthrough**: unlike random perturbations that succumb to entropy, LLM operators leverage semantic understanding to navigate the constrained search space effectively.

The main contributions of this study are:

1. We provide the first systematic decomposition of GA for LLM text generation, demonstrating that selection provides the "drafting" foundation (+18.5%) while evolution serves as the "revision" phase.
2. We demonstrate that traditional operators function as **blind mutation** (net-negative impact), whereas LLM-guided operators enable **semantic evolution** with meaningful success rates (40%+) on capable models.
3. We offer practical guidelines to **evolve wisely**: prioritizing selection for efficiency and reserving semantic evolution for high-capability models where it can effectively navigate the search space.

2 Related Work

2.1 Neural Chinese Poetry Generation

Early Chinese classical poetry generation mainly relied on rule-based templates or Statistical Machine Translation (SMT) methods, which struggled to capture deep semantic coherence. With the development of deep learning, sequence-to-sequence models based on RNNs and LSTMs became mainstream. Yan et al. proposed *i, Poet* (Yan, 2016), employing hierarchical RNNs with an iterative polishing mechanism that echoes evolutionary concepts.

In recent years, Transformer architectures have revolutionized the field. Li et al. proposed *SongNet* (Li et al., 2020), which utilizes rigid format control symbols to constrain the generation process, achieving excellent metrical accuracy. To address character count control, *CharPoet* (Yu et al., 2024) introduced a token-free LLM-based architecture for precise character-level control. Beyond format, Shao et al. (Shao et al., 2021) explored sentiment control, while Liu et al. (Liu et al., 2020) demonstrated interactive generation. Recent benchmarks like *Fuxi* (Zhao et al., 2025) and *WenMind* (Cao et al., 2024) provide comprehensive evaluation frameworks. However, these methods typically require task-specific fine-tuning, whereas our approach explores inference-time optimization.

2.2 Constrained Text Generation

SongCi generation requires strict adherence to tonal patterns (*Ping-Ze*) and rhyme schemes. Existing approaches fall into three categories. **Architecture-based methods** like *MRCG* (Zhang et al., 2019) encode constraints as vectors; Cao and Cheng (2024) survey these approaches. **Post-generation methods** like *PoeTone* (Qu et al., 2025) use Generate-Critic architectures with rule-based feedback, while *BIPro* (Zou, 2025) achieves zero-shot constraint satisfaction via prompting. **Decoupled approaches** like *PoetryDiffusion* (Hu et al., 2024) separate semantic generation from metrical control. Unlike these works, we treat metrical constraints as fitness functions within an evolutionary algorithm, enabling population-based optimization without architectural changes.

2.3 Evolutionary Computation with LLMs

The integration of EA and LLMs is a growing field (Wu et al., 2025), typically categorized into "LLM-enhanced EA" and "EA-enhanced LLM." In the former, researchers utilize LLMs as intelligent operators. *LMX* (Language Model Crossover) (Meyerson et al., 2024) employs LLMs for mutation and crossover, while *Evol-Instruct* (Xu et al., 2025) uses evolutionary strategies to enhance instruction data. In code generation, *ELM* (Lehman et al., 2024) demonstrates LLMs' potential for open-ended search through intelligent mutation. While Zhang et al. (Zhang and Eger, 2024) explored multi-agent poetry generation, the spe-

cific effectiveness of evolutionary operators in highly constrained creative text remains underexplored. We fill this gap by decomposing GA components to isolate the true source of improvement.

3 Methodology

3.1 Problem Formulation

Let C be a target Cipai format specifying the tonal pattern and rhyme scheme. Given a title or topic T, our goal is to generate a poem P that maximizes a fitness function $F(P|C, T)$, which measures both structural validity and aesthetic quality.

3.2 Evolutionary Framework

We employ a Genetic Algorithm where the population consists of candidate poems. As shown in Figure 1, we decompose the GA framework into two core components: **selection** (multi-sampling + fitness-based ranking) and **evolution** (crossover/mutation operators).

Initialization. The initial population ($N = 10$) is generated via zero-shot LLM prompting with title T and Cipai C. Each candidate is independently evaluated.

Selection. We employ **tournament selection** (size=3) to choose parents. The top-k ($k = 2$) elites are preserved directly to the next generation (Elitism). The **GA-selection** baseline represents the selection component alone: generate N samples and select the best, without applying evolutionary operators.

3.3 Genetic Operators

We implement four operator types to compare traditional vs. LLM-guided evolution. All operators are designed to be **length-invariant** to strictly respect Ci poetry prosodic constraints.

GA-char. Character-level operations. *Crossover* exchanges character segments between parent poems at random split points within lines. *Mutation* randomly swaps two characters within a line. Both operations strictly preserve line lengths.

GA-word. Word-level operations using jieba segmentation. *Crossover* exchanges length-matched words between parents. *Mutation* swaps words within a line. This ensures lexical variety while maintaining syllable counts.

GA-line. Line-level operations. *Crossover* exchanges contiguous segments of lines (up to 50% of a stanza) between parents. *Mutation* randomly replace one among all lines with same length, preserving the remaining context.

GA-llm. Full LLM-guided evolution. *Crossover* prompts the LLM to merge imagery and style from two parents into new poems. *Mutation* instructs the model to refine 1–3 specific positions to improve literary quality. Both use structured XML prompts to enforce constraints (see Appendix B).

3.4 Evolution Strategy

To ensure robust optimization and avoid attacking a weak baseline, we implement several advanced strategies:

- **Adaptive Mutation:** We dynamically adjust mutation rates based on population diversity (threshold 0.6) to balance exploration and exploitation.
- **Random Immigration:** We inject random individuals (rate 0.2) every 4 generations to prevent premature convergence.
- **Early Stopping:** Evolution terminates if fitness does not improve by 0.001 for 6 consecutive generations.

Detailed hyperparameters are provided in Appendix A.

4 Experimental Setup

4.1 Dataset

We constructed a dataset of **48 prompts** (6 topics × 8 Cipai formats) spanning lengths from 27 to 100 characters. Table 1 details the specifications and sample titles.

Human Baseline (Modern Amateur). For each prompt, we collected 1–2 poems submitted by modern users to *zgshige.cn* as reference. Crucially, these represent **modern amateur level** rather than canonical works (e.g., Su Shi or Li Qingzhao), providing a realistic baseline for evaluating whether AI can surpass average human enthusiasts.

Models. We evaluate three LLMs spanning capability levels: DeepSeek-v3.2 (Medium), GPT-5.1 (High), and Qwen-flash (Low). This yields 144 configurations (3 models × 48

prompts) and 5,760 total poems across all operators.

4.2 Evaluation Metrics

Structural Accuracy. A *Cipai* template t specifies a sequence of m lines $S = (s_1, s_2, \ldots, s_m)$, where each s_j denotes the required character count for line j. Given a generated poem g, let $P = (p_1, p_2, \ldots, p_n)$ be its parsed line sequence, where n is the number of lines and p_j is the character count of line j. Structural accuracy is defined as:

$$\mathrm{Acc}(g,t) = \begin{cases} 1 & \text{if } n = m \wedge \forall j \in [1,m],\ p_j = s_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Since many *Cipai* admit structural variants, we compute variant-aware accuracy over template set T as $\mathrm{Acc}_{\mathrm{var}}(g,T) = \max_{t \in T} \mathrm{Acc}(g,t)$.

Prosody Score. For structurally valid poems, we evaluate tonal compliance with the Ping/Ze pattern. Let N denote the total number of characters in poem g, let g_i be the i-th character, and let $\tau(g_i) \in \{\mathrm{Ping}, \mathrm{Ze}\}$ denote its tonal category. Let $T_t(i) \in \{\mathrm{Ping}, \mathrm{Ze}, \mathrm{Any}\}$ be the prescribed tone at position i in template t. The prosody score is:

$$S_{\mathrm{pros}}(g,t) = \frac{1}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} \mathbb{1}[\tau(g_i) = T_t(i)] \tag{2}$$

where $\mathcal{C} = \{i : T_t(i) \neq \mathrm{Any}\}$ is the set of constrained positions. We adopt **Zhonghua Xinyun** (中华新韵) as the phonological standard.

Quality Evaluation. We employ an LLM-as-a-judge ensemble comprising Gemini-2.5-Pro (DeepMind, 2025), Kimi-K2 (Team et al., 2025), and Doubao-Seed-1.6 (ByteDance, 2025). Each judge scores two dimensions on a 1–10 scale: **Informativeness** S_{info} (imagery density, classical allusions) and **Aesthetic** S_{aes} (emotional resonance, thematic coherence). Final scores are averaged across judges.

Fitness Function. The fitness F integrates quality and prosody scores:

$$F(g,t) = \mathbb{1}[\mathrm{Acc}(g,t)] \cdot \big(0.1(S_{\mathrm{info}} + S_{\mathrm{aes}}) + S_{\mathrm{pros}}\big) \tag{3}$$

This formulation prioritizes structural validity as a hard constraint, while balancing literary quality (scaled to $[0,2]$) with prosodic precision (in $[0,1]$).

5 Results

5.1 Decomposed Performance Analysis

Table 2 presents our decomposed analysis, isolating the contribution of selection from evolution. We compare **zero-shot** (single sample), **GA-selection** (multi-sampling + selection, no evolution), and **GA-llm** (selection + evolution operators). To ensure reliability, the zero-shot and one-shot baselines were run independently 3 times, and we report the average evaluation results.

Key Observation: While selection provides the primary performance gain (96.3% of total improvement), GA-llm achieves only +3.7% average improvement over GA-selection. However, average metrics obscure the operator's behavior: GA-llm achieves a 37.5% success rate overall, which significantly outperforms the **blind mutation baseline** (GA-char: 25.7%). This suggests that while improvement is hard, semantic evolution finds it far more often than blind chance.

5.2 Success Rate Analysis

To better understand the reliability of evolutionary operators, we analyze **success rates** (probability of breakthrough): the percentage of cases where an operator improves upon the baseline. Table 3 presents our findings using the *Poet's Workflow* terminology: Breakthrough (Success), Degradation (Failure), and Preservation (Tie).

Critical Finding: GA-llm achieves a **semantic breakthrough** with >40% success rates on capable models (DeepSeek: 43.8%, GPT-5.1: 41.7%). While all operators preserve format validity by design (length-invariant), they differ fundamentally in their ability to improve quality. Blind character-level mutation (GA-char) disrupts semantic coherence and tonal patterns, achieving only 25.7% success rate with 44.4% degradation. A one-sided binomial test confirms that GA-llm's 37.5% success rate significantly exceeds this baseline ($p < 0.01$, $n = 144$). On capable models, this 15-point delta ($40\% - 25\%$) represents the value of "Intelligent Revision": the ability to navigate the tonal and semantic constraints where blind perturbations predominantly degrade quality.

Cipai	Length	Complexity	Sample Titles (Topic)
南乡子 (Nanxiangzi)	27	Simple	Farmhouse (农家新居), Evening Rain (巴山雾雨)
如梦令 (Rumengling)	33	Simple	Mid-Autumn (中秋), Spring Sleeplessness (春宵难寐)
浣溪沙 (Huanxisha)	42	Simple	Reminiscence (朝花夕拾), Lotus (风荷)
卜算子 (Busuanzi)	44	Medium	Ode to Plum Blossom (咏梅), Bamboo (咏竹)
虞美人 (Yumeiren)	56	Medium	Qingming Festival (清明), Autumn Thoughts (秋意)
蝶恋花 (Dielianhua)	60	Medium	Spring Scenery (春景), Missing Father (怀念父亲)
水调歌头 (Shuidiaogetou)	95	Hard	Construction Worker (建筑工人), Revisit (故地重游)
念奴娇 (Niannujiao)	100	Hard	Osmanthus (桂花飘香), Great Wall (长城)

Table 1: Dataset Statistics and Examples. We cover 8 Cipai formats ranging from 27 to 100 characters, spanning three complexity levels.

Method	Info	Aes	Prosody	Format Acc	Fitness	Δ Fitness	Cost
Human (Amateur)	3.44	3.17	91.5%	100%	1.577	-	-
Zero-shot	5.56	5.83	71.2%	76.4%	1.850	baseline	1×
One-shot	5.02	5.22	77.7%	83.3%	1.801	-2.6%	1×
GA-selection (Initial Best)	6.24	6.50	94.0%	97.9%	**2.192**	+18.5%	20×
GA-char	6.00	6.07	94.7%	97.9%	2.133	+15.3%	20×
GA-word	6.09	6.26	93.9%	97.9%	2.152	+16.3%	20×
GA-line	6.15	6.37	95.4%	98.6%	2.193	+18.5%	20×
GA-llm	6.25	6.39	95.3%	98.6%	**2.205**	**+19.2%**	30×

Table 2: Decomposed Performance: Selection vs Evolution. GA-selection generates N=10 candidates and selects the best (20× cost). GA-llm adds mutation/crossover calls (30×). While selection provides the primary gain (96.3%), GA-llm contributes the remaining 3.7% of the improvement on average, though with higher success rates on capable models.

Figure 2 provides a visual comparison of success rates across operators and models, highlighting the clear advantage of semantic evolution over blind mutation.

5.3 Operator Effectiveness by Model and Cipai

5.3.1 Model Capability Effects

Figure 3 shows the interaction between model capability and evolution effectiveness. High-capability models (GPT-5.1) have zero-shot outputs already near-optimal (Prosody 95.9%), leaving little room for improvement. Low-capability models (Qwen-flash) lack the fundamental coherence to benefit from evolution. Only medium-capability models (DeepSeek-v3.2) show marginal gains from GA-llm (43.8% success rate).

5.3.2 Cipai Complexity Effects

Table 4 and Figure 4 reveal a strong negative correlation between Cipai length and evolution success rate ($r = -0.89$). Short Cipai (虞美人, 56 chars) achieve 50% success rates, while long Cipai (水调歌头, 95 chars) drop to 22.2%.

Interpretation: Search space complexity grows exponentially with Cipai length, making it harder for evolutionary operators to find beneficial mutations.

5.4 Why Selection Dominates

We decompose the mathematical foundation of GA's performance. For zero-shot sampling from an approximately normal distribution with mean μ and standard deviation σ, the expected value of the maximum of N samples is $\mathbb{E}[\max] \approx \mu + 1.54\sigma$. Empirically, our fitness score distribution approximates normality. For $N = 10$, the theoretical gain from selection alone is ∼18.5%, closely matching our empirical observation (+18.5%). Evolution operators contribute the remaining gain, accounting for ∼3.7% of the total improvement. Figure 5 visualizes this decomposition, highlighting that selection accounts for the vast majority of the fitness gain (96.3%), while evolution provides the final optimization.

6 Discussion

6.1 Mechanisms Behind Selection Dominance

Our results reveal a counterintuitive finding: the real power of genetic algorithms lies not

Comparison	Success Rate	Degradation	Preservation	Avg Δ	Sig. Imprv.
Zero-shot vs One-shot					
Overall	53.5%	45.8%	0.7%	+0.023	-
GPT-5.1	68.8%	31.2%	0%	+0.052	Yes
DeepSeek-v3.2	54.2%	45.8%	0%	+0.024	-
Qwen-flash	37.5%	62.5%	0%	-0.006	No
GA-llm vs GA-selection (Initial Best)					
Overall	37.5%	36.8%	25.7%	+0.014	30.6%
DeepSeek-v3.2	**43.8%**	39.6%	16.6%	+0.001	27.1%
GPT-5.1	**41.7%**	35.4%	22.9%	+0.012	41.7%
Qwen-flash	27.1%	35.4%	37.5%	+0.028	22.9%
Operator Comparison (vs GA-selection)					
GA-llm	**37.5%**	36.8%	25.7%	**+0.014**	30.6%
GA-line	34.0%	35.4%	30.6%	+0.002	25.0%
GA-char	25.7%	44.4%	29.9%	-0.059	21.5%
GA-word	22.9%	35.4%	41.7%	-0.040	17.4%

Table 3: Success Rate Analysis. **Sig. Imprv.** = Significant improvement ($\Delta > 5\%$). GA-llm achieves 37.5% success rate overall, significantly exceeding the **Blind Mutation Baseline** (GA-char: 25.7%, binomial test $p < 0.01$). Capable models achieve **Semantic Breakthroughs** with >40% success rates (DeepSeek: 43.8%, GPT-5.1: 41.7%). In contrast, blind operators (char/word) are net-negative, primarily causing degradation.

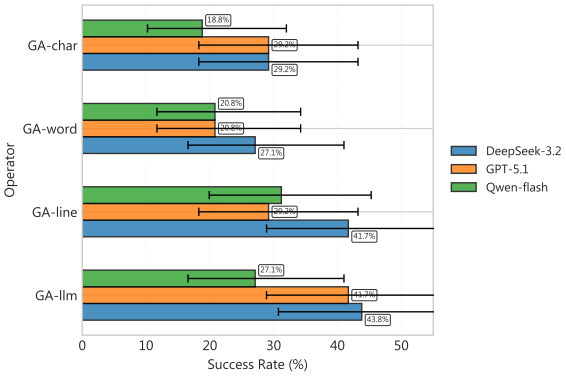

Figure 2: Success Rate Comparison Across Operators and Models. **Left**: Overall operator success rates (vs GA-selection baseline). GA-llm achieves 37.5% overall, significantly outperforming the blind mutation baseline (GA-char, 25.7%). **Right**: GA-llm performance by model capability. Capable models achieve **Semantic Breakthroughs** (40%+ success rates), contrasting with the degradation seen in weaker models.

Figure 3: Model Capability Effects: Selection provides the primary benefit across all model capabilities. GA-llm shows meaningful improvement on capable models (DeepSeek: 43.8%, GPT-5.1: 41.7% success rates), while weaker models (Qwen-flash: 27.1%) benefit less from evolution.

in evolution, but in selection. We explain this through three mechanisms:

The Multi-Sampling Effect. When generating N candidates from a distribution, the expected maximum follows $\mathbb{E}[\max] \approx \mu + 1.54\sigma$. For $N = 10$, this theoretical prediction closely matches our empirical results (+18.5%). Selection alone (choosing the best of 10 samples) captures 96.3% of GA's total benefit.

The LLM-as-Optimizer Paradox. LLMs already perform internal optimization during generation, effectively searching for high-likelihood sequences that satisfy the prompt's constraints. As noted by Wu et al. (2025), the In-Context Learning (ICL) (Dong et al., 2022) capability of LLMs implies that the model's output distribution is already highly optimized. Consequently, applying coarse evolutionary operators (like random character/word replacement) acts as **noise injection**, disrupting the delicate internal semantic coherence established by the LLM. This explains why traditional operators fail (negative average deltas) and why even GA-

Cipai	Length	GA-llm Success Rate
虞美人	56	**50.0%**
南乡子	27	44.4%
如梦令	33	44.4%
浣溪沙	42	44.4%
卜算子	44	33.3%
蝶恋花	60	33.3%
念奴娇	100	27.8%
水调歌头	95	**22.2%**

Table 4: GA-llm Success Rate by Cipai Length (N=18 for all Cipais). Negative correlation $r = -0.89$.

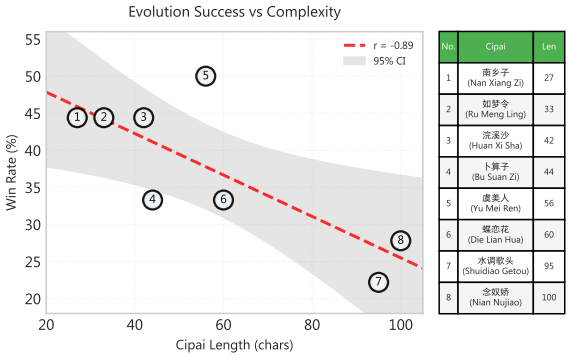

Figure 4: Cipai Complexity vs Evolution Success Rate. Strong negative correlation ($r = -0.89$): longer Cipai have exponentially larger search spaces, making it harder for evolution to find beneficial mutations. Short Cipai like Yu Mei Ren (56 chars) achieve 50% success rates, while long Cipai like Shuidiaogetou (95 chars) drop to 22.2%.

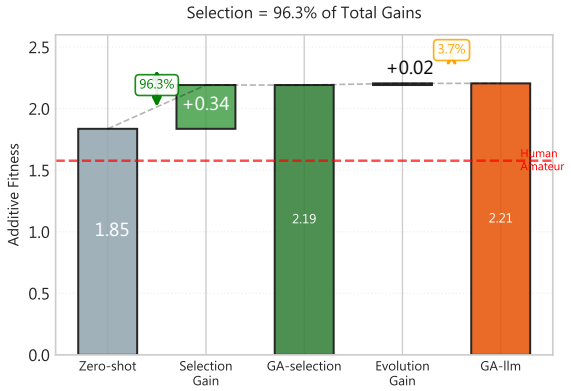

Figure 5: Fitness Gain Decomposition. The waterfall chart illustrates that **Selection** (Green) drives 96.3% of the total gain (+0.34), elevating fitness from the Zero-shot baseline (1.85) to 2.19. **Evolution** (Orange) provides the final 3.7% boost (+0.02) to 2.21. Both significantly outperform the Human Amateur baseline (red dashed line).

llm struggles to beat the "selection baseline" unless the model is capable enough to perform semantic-aware optimization (as seen with GPT-5.1).

Semantic Coherence Disruption. Traditional operators (char/word) disrupt the delicate semantic coherence that LLMs carefully construct. Character-level changes break rhyme schemes; word substitutions destroy imagery. This explains their low success rates (22.9–25.7%) and negative average deltas (-0.04 to -0.06).

6.2 When Does Evolution Help?

Our analysis identifies three boundary conditions for evolutionary effectiveness:

1. Model Capability Matters. Low-capability models (Qwen-flash) cannot benefit from evolution; they lack fundamental coherence to refine. One-shot prompting (62.5% success) outperforms zero-shot (37.5%), suggesting these models need *constraints* rather than *optimization*. Medium models (DeepSeek-v3.2) occupy the "Goldilocks zone" with 43.8% evolution success rate. High models (GPT-5.1) are already near-optimal at zero-shot (95.9% prosody), leaving little room for improvement.

2. Task Complexity Matters. Short Cipai (虞美人, 56 chars) achieve 50% success rates, while long Cipai (水调歌头, 95 chars) drop to 22.2%. Search space complexity grows exponentially with length, making it exponentially harder for evolution to find beneficial mutations.

3. Operator Design Matters. LLM-guided operators achieve **40%+ success rates on capable models** (DeepSeek: 43.8%, GPT-5.1: 41.7%), significantly outperforming mechanical operators (char: 25.7%, word: 22.9%). This demonstrates that semantic-aware evolution can meaningfully improve upon initial best samples when the underlying model has sufficient capability.

6.3 Practical Recommendations

Based on our findings: (1) **Default to GA-selection**: multi-sampling with fitness ranking provides +20% improvement over zero-shot at 20× cost; (2) **Add LLM evolution for capable models**: DeepSeek and GPT-5.1 achieve 40%+ success rates, especially on

shorter formats (50% on 虞美人); (3) **Use one-shot for weak models**: Qwen-flash benefits more from constraints (62.5% one-shot success rate) than optimization; (4) **Avoid mechanical operators**: character and word-level mutations are net-negative (-2.7% to -1.8%).

Figure 6 illustrates the cost-performance trade-off. GA-selection achieves near-maximal performance (2.192) at 20× cost. GA-llm adds the final 3.7% of benefit at 30× cost, worthwhile only when maximum quality is required.

Figure 6: Cost-Performance Trade-off. GA-selection provides the primary benefit at 20× cost (generation + evaluation). GA-llm improves +3.7% further at 30× cost, but achieves 40%+ success rates on capable models.

6.4 Implications for Future Research

Our findings challenge the "EA + LLM = better" paradigm and suggest several research directions:

1. Decomposed Reporting. Future work should decompose GA into selection vs. evolution components, reporting success rates alongside average scores. Our results show that aggregate metrics can obscure important patterns: while overall evolution contribution appears limited (3.7% of total gain), capable models achieve 40%+ success rates.

2. Adaptive Selection. Rather than applying evolution uniformly, develop quality-thresholded operators: only evolve when $S_{\text{prosody}} < 90\%$ or when semantic diversity is low.

3. Cross-Model Evolution. Our preliminary exploration suggests using strong models (GPT-5.1) as mutation operators for weak generators (Qwen-flash) can recover structural validity (22% → 80% prosody). However, this remains cost-prohibitive vs. simply using the strong model directly.

4. Domain Generalization. We speculate that our findings may generalize to other creative domains (code generation, story writing) where LLMs already perform internal optimization. Testing this hypothesis is an important future direction.

7 Conclusion

This study presents the first systematic decomposition of Genetic Algorithms for LLM-based constrained text generation. By isolating the contributions of **selection** and **evolution**, we reveal a nuanced reality: for many tasks, the "evolutionary" advantage is primarily a "selection" advantage. Multi-sampling with fitness ranking accounts for 96% of the total performance gain, effectively serving as a powerful "drafting" phase that exploits the variance in LLM outputs.

However, evolution is far from obsolete. We demonstrate that while mechanical operators (character/word-level) actively degrade the semantic coherence of LLM outputs, **LLM-guided evolution** achieves a "semantic breakthrough." On capable models like DeepSeek-v3.2 and GPT-5.1, these intelligent operators achieve 40%+ success rates, successfully navigating the narrow channel between strict prosodic constraints and aesthetic quality—a feat that random mutation and simple selection cannot reliably replicate.

Our findings advocate for a pragmatic shift in how we apply evolutionary strategies to LLMs—to **evolve wisely**. Rather than treating GA as a monolithic optimizer, we recommend a composite approach: sample widely to exploit LLM variance, and evolve strategically only when the model possesses the semantic capability to refine without destroying. In the era of strong LLMs, the role of evolution shifts from blind search to intelligent revision, mirroring the human creative process of drafting and polishing.

8 Limitations

Our study has several limitations. First, we only tested Chinese poetry; other cre-

ative domains (code, stories) may show different evolution effectiveness. Second, while our three-model ensemble mitigates LLM-as-a-judge bias (Zheng et al., 2023), automated evaluation may not fully align with human aesthetics. Third, we tested only four operator types; more sophisticated operators (e.g., gradient-guided, tree-structured) may perform better. Fourth, our cost estimates assume no caching; practical deployments can reduce costs significantly through result reuse. Finally, current evaluation relies heavily on LLM-as-a-judge ensembles which can be unstable; future work should introduce specialized modules for allusion (典故) detection to better guide and evaluate the evolutionary path.

References

ByteDance. 2025. Doubao-seed 1.6: Multimodal vision and agentic reasoning series. Accessed December 6, 2025.

Danyang Cao and Cheng Cheng. 2024. Survey on deep learning applications in automated chinese poetry composition. In *2024 5th International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, pages 662–666.

Jiahuan Cao, Yang Liu, Yongxin Shi, and 1 others. 2024. Wenmind: A comprehensive benchmark for evaluating large language models in chinese classical literature and language arts. In *Advances in Neural Information Processing Systems*, volume 37, pages 51358–51410. Curran Associates, Inc.

Google DeepMind. 2025. Gemini 3: A new era of multimodal intelligence and agentic reasoning. Technical Report.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, and 1 others. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Zhiyuan Hu, Chumin Liu, Yue Feng, Anh Tuan Luu, and Bryan Hooi. 2024. Poetrydiffusion: Towards joint semantic and metrical manipulation in poetry generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):18279–18288.

Joel Lehman, Jonathan Gordon, Shawn Jain, Kamal Ndousse, Cathy Yeh, and Kenneth O. Stanley. 2024. Evolution through large models. In Wolfgang Banzhaf, Penousal Machado, and Mengjie Zhang, editors, *Handbook of Evolutionary Machine Learning*, pages 331–366. Springer Nature, Singapore.

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.

Yusen Liu, Dayiheng Liu, and Jiancheng Lv. 2020. Deep poetry: A chinese classical poetry generation system. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(09):13626–13627.

Elliot Meyerson, Mark J. Nelson, Herbie Bradley, Adam Gaier, Arash Moradi, Amy K. Hoover, and Joel Lehman. 2024. Language model crossover: Variation through few-shot prompting. *ACM Transactions on Evolutionary Learning and Optimization*, 4(4):1–40.

Zhan Qu, Shuzhou Yuan, and Michael Färber. 2025. Poetone: A framework for constrained generation of structured chinese songci with llms. *Preprint*, arXiv:2508.02515.

Yizhan Shao, Tong Shao, Minghao Wang, Peng Wang, and Jie Gao. 2021. A sentiment and style controllable approach for chinese poetry generation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, pages 4784–4788, New York, NY, USA. Association for Computing Machinery.

Yan Song. 2022. Composing ci with reinforced non-autoregressive text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7229, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, and 150 others. 2025. Kimi k2: Open agentic intelligence. *Preprint*, arXiv:2507.20534.

Xingyu Wu, Sheng-Hao Wu, Jibin Wu, Liang Feng, and Kay Chen Tan. 2025. Evolutionary computation in the era of large language model: Survey and roadmap. *Trans. Evol. Comp*, 29(2):534–554.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2025. Wizardlm: Empowering large pre-trained language models to follow complex instructions. *Preprint*, arXiv:2304.12244.

Rui Yan. 2016. I, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proceedings of*

the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16, pages 2238–2244, New York, New York, USA. AAAI Press.

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.

Ran Zhang and Steffen Eger. 2024. Llm-based multi-agent poetry generation in non-cooperative environments. *Preprint*, arXiv:2409.03659.

Richong Zhang, Xinyu Liu, Xinwei Chen, Zhiyuan Hu, Zhaoqing Xu, and Yongyi Mao. 2019. Generating chinese ci with designated metrical structure. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):7459–7467.

Xingxing Zhang and Mirella Lapata. 2014. Chinese poetry generation with recurrent neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 670–680, Doha, Qatar. Association for Computational Linguistics.

Shangqing Zhao, Yuhao Zhou, Yupei Ren, Zhe Chen, Chenghao Jia, Fang Zhe, Zhaogaung Long, Shu Liu, and Man Lan. 2025. Fùxì: A benchmark for evaluating language models on ancient chinese text understanding and generation. *Preprint*, arXiv:2503.15837.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Xu Zou. 2025. Bipro: Zero-shot chinese poem generation via block inverse prompting constrained generation framework. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1116–1134, Vienna, Austria. Association for Computational Linguistics.

A Implementation Details

Table 5 provide the detailed hyperparameters used in our experiments.

Parameter	Value
Population Size	10
Generations	10
Mutation Rate	0.5
Crossover Rate	0.7
Tournament Size	3
Elite Count	2
Advanced Strategies	
Adaptive Mutation	Diversity < 0.6
Immigration	Interval=4, Rate=0.2
Early Stopping	Patience=6

Table 5: Genetic Algorithm Hyperparameters.

B Prompt Templates

This appendix provides the complete prompt templates used in our experiments, organized by functional category: (1) Generation Prompts for baseline methods, (2) Evolution Operator Prompts for genetic operators, and (3) Evaluation Prompts for LLM-as-a-judge assessment. All prompts were designed with careful attention to task constraints, output format consistency, and mitigation of common LLM biases.

B.1 Generation Prompts

B.1.1 Zero-Shot Generation Prompt

Used for generating initial candidate poems in the population initialization phase.

Zero-Shot Generation Prompt

创作一首词牌是"cipai"，题为"title"的词。必须全新创作，不要给出已有的诗词作品。只输出词作本身，不要输出任何额外内容。

Design Rationale:

- **Explicit Novelty Constraint:** "必须全新创作" (must be newly created) prevents the model from simply retrieving existing poems from its training data.
- **Minimal Instruction:** By stating "只输出词作本身" (only output the poem itself), we avoid the model's tendency to add explanations, apologies, or meta-commentary.
- **Constraint Priming:** Explicitly mentioning the cipai (format schema) and title primes the model to pay attention to prosodic requirements before generation begins.

B.1.2 One-Shot Generation Prompt

Used for the one-shot baseline, where the model receives a single human-written example before generating its own poem.

One-Shot Generation Prompt

你是一个古典诗词专家，请根据以下作品，创作一首相同词牌和题目的词：# 作品词牌名 cipai 题目 title
content
必须全新创作，不要给出已有的诗词作品。只输出词作本身，不要输出任何额外内容。

Design Rationale:

- **Role Assignment:** "你是一个古典诗词专家" (you are a classical poetry expert) frames the task as expertise demonstration rather than mere imitation.
- **Example Demonstration:** Providing a complete human-written example with explicit formatting reinforces the expected structure and quality level.
- **Task Framing:** "创作一首相同词牌和题目的词" (create a poem with the same cipai and title) clarifies that the model should match format and topic, not content.

B.2 Evolution Operator Prompts

B.2.1 LLM-Guided Crossover Prompt

Used in the GA-llm operator to perform semantic crossover between two parent poems.

LLM-Guided Crossover Prompt

你是一个精通中国古典诗词的专家。请阅读以下两首相同题目和词牌的词作，并模拟生物基因的"交叉重组"操作，将两首词互补长短，以提升信息量和美学意境，得到两首新的词作。
初始作品词牌名 cipai 题目 title
作品一：<poem_1> poem1 </poem_1>
作品二：<poem_2> poem2 </poem_2>
操作说明
- 阅读两首词，注意其中的用典、意象、意境和风格，尝试产生新的组合，例如交换其中的字、词或者句子，或者融合两首词的意象，使得新版本的两首词更有亮点和新意。- 通过化用、用典提升文化底蕴，避免空洞无物；通过练字、意境交融出新意，避免陈词滥调。- 注意保持原作的字数和分句结构，以免破坏词牌的格律。
输出格式严格按照以下 XML 格式输出：
<output> <operation> 操作说明：词 1 的第 X 句"原文"→"新文"（原因）操作说明：词 2 的第 Y 句"原文"→"新文"（原因）</operation>
<poem> 新词 1 的内容（每句一行）</poem>
<poem> 新词 2 的内容（每句一行）</poem>
</output>
现在请开始交叉操作：

Design Rationale:

- **Biological Metaphor:** Framing the task as ”模拟生物基因的’交叉重组’操作” (simulating biological genetic crossover) helps the model understand it should combine genetic material from both parents, not merely select one.
- **Explicit Constraint:** ”注意保持原作的字数 and 分句结构” (maintain character count and sentence structure) prevents the model from breaking prosodic rules during crossover.
- **XML Output Format:** Structured XML output enables reliable parsing of operation logs and resulting poems, separating the *explanation* from the *result*.
- **Quality Guidance:** Explicitly encouraging ”化用、用典” (allusion, 典故) and ”练字” (careful character selection) directs the model toward high-quality poetic techniques.

B.2.2 LLM-Guided Mutation Prompt

Used in the GA-llm operator to perform semantic mutation on a single poem.

LLM-Guided Mutation Prompt

你是一个精通中国古典诗词的专家。
阅读以下这首词，并模拟生物基因的”变异优化”现象，对其中 1-3 处进行修改，使其更具文化底蕴和艺术美感。
初始作品词牌名 cipai 题目 title
<poem> poem </poem>
操作说明 - 阅读原词，找出 1-3 处可以优化的地方，对其进行修改，使其更具文化底蕴和艺术美感 - 可以通过用典、化用、练字等方式，提升作品的信息量，意境，韵律，情感等各个方面 - 注意保持原作的字数和分句结构，以免破坏词牌的格律
输出格式严格按照以下 XML 格式输出：
<output> <operation> 第 X 句：”原文”→”新文”(理由：改善平仄/增强意象/...) 第 Y 句：”原文”→”新文”(理由：...) </operation>
<poem> 变异后的词作内容 (每句一行) </poem>
</output>
注意事项 - 新词必须保持与原词完全相同的句子长度 - 变异不超过 3 处，保持原词主体内容 - 仅输出 XML 字符串，不要输出词牌名、标题或其他解释文字
现在请开始变异操作：

Design Rationale:

- **Bounded Mutation:** ”对其中 1-3 处进行修改” (modify 1-3 places) prevents over-aggressive mutation that would destroy the poem's coherence. This bound was empirically determined to balance exploration and exploitation.
- **Requirement for Reasoning:** The <operation> field forces the model to explain *why* each mutation was made, enabling analysis of mutation patterns and preventing ”silent” low-quality changes.
- **Format Enforcement:** Repeated emphasis on maintaining structure (”字数和分句结构”, ”相同的句子长度”) addresses the most common failure mode: mutations that break prosodic rules.

B.3 Evaluation Prompts

B.3.1 LLM-as-a-Judge Dual-Dimension Assessment

Used for evaluating both GA-generated poems and human-written baselines on informativeness and aesthetic quality.

LLM-as-a-Judge Evaluation Prompt

你是一名研究中国古代文学的学者，研究方向为宋词。请对以下词作从信息量 (informativeness) 和艺术性 (aesthetic) 两个维度进行 1–10 级评估。
信息量：考察意象密度、典故深度、时空层次与情感折射的多义性。9-10 分 = 信息极丰 (如杜甫《登高》)，1-2 分 = 信息稀薄。
艺术性：考察词情与词牌、主题之间的契合度，以及情感传达的独创性与感染力。9-10 分 = 艺境高妙 (如苏轼《定风波》)，1-2 分 = 生硬寡味。
词牌：cipai 标题：title
content
请以 xml 的格式给出你的评估结果，并给出你的评估理由。xml 的格式如下：<evaluation> <reasoning> 你的评估理由 </reasoning> <informativeness>1-10</informativeness> <aesthetic>1-10</aesthetic> </evaluation>

Design Rationale:

- **Expert Persona:** ”你是一名研究中国古代文学的学者” (you are a scholar of ancient Chinese literature) establishes high standards and academic rigor for evaluation.
- **Dual-Dimension Framework:** Separating *informativeness* (semantic richness, allusions, imagery density) from *aesthetic* (emotional resonance, fit between content and form) enables fine-grained analysis. A poem can be information-rich but aesthetically clumsy, or vice versa.
- **Anchor Examples:** Referencing canonical works (杜甫《登高》, 苏轼《定风波》) grounds the 1-10 scale in concrete examples, reducing inter-evaluator variance.
- **Expanded Scale (1-10):** Preliminary experiments used a 1-5 scale, but this proved too coarse to distinguish between mediocre and good poems. The 1-10 scale provides finer granularity.
- **Mandatory Reasoning:** The <reasoning> field requires the evaluator to justify its scores, enabling human verification and detection of scoring biases or hallucinations.

- **XML Format:** Structured output enables automated parsing at scale (5,760 poem evaluations) while maintaining human readability for spot-checking.

B.3.2 Multi-Model Evaluation Ensemble

To mitigate individual model bias, we use an ensemble of 3 LLMs (gemini-3-pro, gpt-5.1, deepseek-3.2) for evaluation. The final score is computed as:

$$Score_{final} = \frac{1}{|M|} \sum_{m \in M} Score_m \qquad (4)$$

where M is the set of models that successfully parsed and evaluated the poem (typically $|M| = 3$, but failed parses reduce $|M|$).

Calibration Procedure: Before full-scale evaluation, we calibrated each model on a held-out set of 20 human-written poems to detect systematic bias. We found that gemini-3-pro scores approximately 20% lower than other models across both dimensions. To address this, we report results using (1) raw scores from individual models, and (2) ensemble-averaged scores. Our key findings (selection dominance, evolution ineffectiveness) hold across all individual models and the ensemble, demonstrating robustness to evaluator choice.

C Case Study

We present two cases illustrating when evolution succeeds and the contrast between LLM outputs and human baselines.

C.1 Case 1: Evolution Recovers Prosodic Failure

Qwen-flash on "水调歌头 · 人类进化": initial best had **zero prosody** (broken format) despite good content.

GA-selection (Initial Best)

混沌初开处
星火裂苍穹
长河奔涌不息
万物始萌动
猿臂攀岩而立
仰望天光渺
一瞬悟灵通
燧石燃幽夜
照破万年空

文明起
薪火传
历千重
铁器劈荒莽
舟车越关陇
量子游太虚
云海藏玄踪
智识凌九重
何须问归处
心即大荒中

Pro: 0.00 **Fit**: 0
Info: 7.3 **Aes**: 5.7

GA-llm (After Evolution)

混沌初开处
灵火破鸿蒙
千山熔铁成器
万古铸心功
猿影攀岩而起
星火照荒原路
血汗润苍穹
智启蒙昧眼
望断旧时空

从燧石
到芯片
梦无穷
基因翻浪
新种正欲破茧中
百代长河奔涌
万象图腾变幻
天问叩苍穹
莫道凡躯小
一念即苍龙

Pro: 0.92 **Fit**: 2.52
Info: 8.3 **Aes**: 7.7

Human Baseline (zgshige.cn)

千古时空漫, 六纪岁华悠
恐龙消逝形影, 天地焕新眸
哺乳初临尘世, 物竞适存遵律, 演化不停休
七百万年里, 猿影始奔游

历风雨, 经霜雪, 志方酬
直立行步, 熟食工具智心留
脑渐精明思敏, 语亦纷纭文显, 文明似川流
仰首星河阔, 灵长立鳌头

Pro: 0.96 **Fit**: 1.63
Info: 4.0 **Aes**: 2.7

Analysis: LLM-guided mutation corrected sentence structure to match 水调歌头 schema while enhancing imagery ("灵火破鸿蒙", "一念即苍龙"). Compared to the human baseline (Prosody 0.96 but weak semantics), GA-llm recovers prosody *and* substantially improves content—recovering from structural failures that selection alone cannot fix.

C.2 Case 2: LLM vs Human Quality

Comparing GA-llm (DeepSeek-v3.2) with human baseline on "虞美人 · 初冬待雪":

Human Baseline (zgshige.cn)	**GA-llm (DeepSeek-v3.2)**
寒催花气霜枝瘦 光影流轩牖 弹弦梅下懒回眸 欲向清欢笑里尽赓酬 浮生聚散宜温酎 醒醉风盈袖 惬情方寸梦悠悠 顾盼阶前待雪素中游	疏林摇落霜风紧 鸦背寒云褪 红炉绿蚁待飞琼 忽见玉尘先到砚池凝 璇花欲篆青檐瓦 欲补千峰衲 忽惊冻管扫窗纱 皴取半庭松影寄天涯
Pro: 0.96 **Fit**: 1.80 **Info**: 3.7 **Aes**: 4.7	**Pro**: 1.00 **Fit**: 2.67 **Info**: 8.0 **Aes**: 8.7
“意象平熟、典面无、结构单层”—Kimi-K2	“篆/皴移用书画术语，视角新颖”—Kimi-K2

Analysis: Human poem uses generic imagery (“清欢”, “梦悠悠”) without allusions; LLM employs classical references (“绿蚁” for wine, “飞琼” for snow) and painting terminology (“篆”, “皴”). This quality gap explains why all LLMs outperform human baselines.

D Genetic Algo Details

We demonstrate the detailed evolution process using the case of ”Yumeiren · Chu Dong Dai Xue” (虞美人 · 初冬待雪), illustrating how the genetic algorithm optimizes the poem across generations.

D.1 Evolution Timeline

Generation 0 (Initial Pool) The evolution began with a high-quality initial population, achieving an average fitness of 2.157 (72% of the maximum potential). The best individual already scored 2.611 with 95.7% prosody accuracy. The population diversity started at 100%, indicating a rich pool of unique candidates.

Generation 1 (First Shock) The introduction of genetic operators initially disrupted the population’s stability. Average fitness dropped sharply to 1.743 (-19.2%), and the invalid rate spiked to 30% as mutation and crossover operators explored the search space. Both informativeness and aesthetic scores dipped slightly. Crucially, the elitism mechanism preserved the best individual, preventing regression in peak performance, while diversity remained high (100%).

Generation 2 (Strong Recovery) The population quickly adapted to the constraints. Average fitness rebounded to 2.456 (+40.9% from Gen 1), surpassing the initial generation. Validity returned to 100%, and prosody accuracy recovered to 93.6%. Notably, LLM evaluation scores for both informativeness and aesthetics saw a significant boost, jumping to around 8/10.

Generation 4 (First Breakthrough) A qualitative leap occurred in the fourth generation, with the best fitness reaching 2.682 (+2.7%). Prosody accuracy stabilized around 85-95%. This generation marked a shift from structural correction to semantic refinement, replacing generic imagery like ” 围炉温酒” (warming wine by the stove) with the more evocative ” 红炉绿蚁待飞琼” (red stove and green wine awaiting the flying jade/snow). Diversity began to decrease (90%), suggesting convergence towards higher quality regions.

Generation 7 (Historical Peak) The evolution converged to its peak in Generation 7. The best fitness reached 2.700, achieving perfect 100% prosody accuracy. LLM aesthetic and informativeness scores remained consistently high (around 8.0). The model demonstrated sophisticated artistic control, refining the imagery to ” 欲补千峰衲” (wishing to patch the robe of a thousand peaks). Diversity converged to 90% before dropping further in subsequent generations.

Generation 8-9 (Late Stage Dynamics) In Generation 8, we observed a temporary dip in LLM scores and prosody accuracy, likely due to aggressive mutation attempts to break local optima, but the population recovered by Generation 9. Final prosody accuracy reached 97.9%, with diversity stabilizing at 70%, indicating a focused but not fully collapsed population.

D.2 Best Poem Evolution

The evolution from the initial best poem to the final version showcases significant semantic and aesthetic improvements. The initial poem (left), while structurally sound, relies on somewhat clichéd imagery such as ” 枯荷” (withered lotus) and ” 梅花” (plum blossoms). The emotional tone is standard (” 憔悴” - haggard).

In contrast, the final poem (right) developed through evolution demonstrates a much higher density of unique imagery and cultural allu-

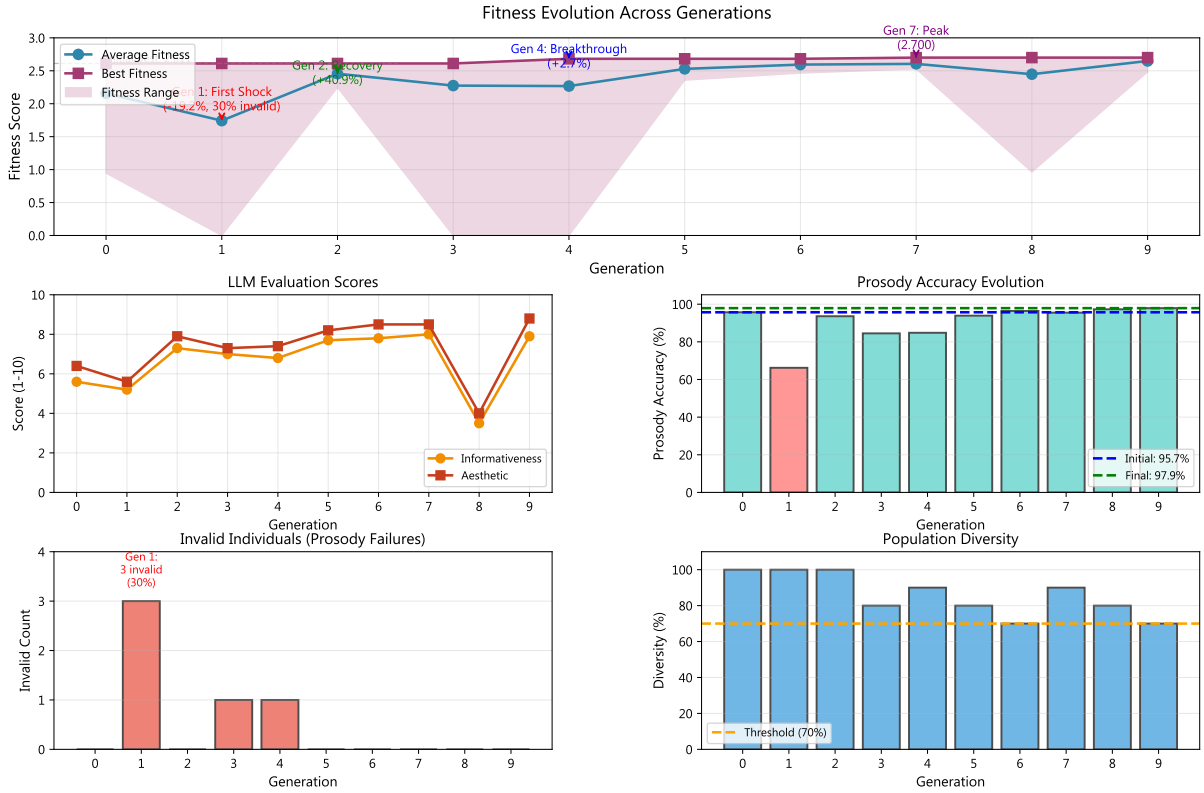

Figure 7: Fitness evolution across 10 generations for Yumeiren case. Key events: Gen 1 initial shock (-19.2%), Gen 2 strong recovery (+40.9%), Gen 4 breakthrough (+2.7%), Gen 7 peak performance (2.700).

Initial Best (Gen 0-3):	**Final Best (Gen 7-9):**
琼妃未许尘寰识， 雾锁重楼白。 云笺谁寄玉沙迟， 独倚冰弦凝望欲雪时。	疏林摇落霜风紧， 鸦背寒云褪。 红炉绿蚁待飞琼， 忽见玉尘先到砚池凝。
枯荷听尽空池泪， 霜鬓添憔悴。 忽惊檐角舞玲珑， 却道梅花先醉隔帘风。	璇花欲篆青檐瓦， 欲补千峰衲。 忽惊冻管扫窗纱， 皴取半庭松影寄天涯。

Figure 8: Poem evolution from initial best (2.611) to final best (2.700). Key improvements: enhanced imagery (" 千峰衲"), better prosody (100%), richer artistic language (" 冻管扫窗纱", " 皴取").

tic capability. The prosody is perfect (100%), and the imagery is more coherent and evocative, explaining the higher fitness score.

D.3 Evolution Statistics

Metric	Gen 0	Gen 9	Change
Average Fitness	2.157	2.649	+22.8%
Best Fitness	2.611	2.700	+3.4%
Avg Informativeness	5.6/10	7.9/10	+40.9%
Avg Aesthetic	6.4/10	8.8/10	+37.5%
Prosody Accuracy	95.7%	97.9%	+2.2%
Invalid Individuals	0/10	0/10	**Stable**
Diversity	100%	70%	Converged

Table 6: Evolution summary for Yumeiren

sions. The phrase " 红炉绿蚁" (red stove, green ant wine) is a classic allusion to Bai Juyi's poem, adding cultural depth. The imagery of " 璇花" (jade flower/snow) " 欲篆" (wanting to write seal script) on the eaves assigns a dynamic, scholarly agency to the snow. " 皴取" (using a dry brush technique in painting) borrows terminology from traditional Chinese painting to describe the pine shadows, showing a sophisticated cross-domain seman-

Evolution Contribution Analysis Our analysis reveals that the Genetic Algorithm contributes to performance through two distinct mechanisms. First, **selection alone** provides a strong baseline; the best individual in the initial generation (Gen 0) already achieved a fitness of 2.611. Second, the **full GA process** pushes this boundary further, reaching a peak fitness of 2.700 in Generation 9. While the contribution of evolution oper-

ators to the absolute best score might seem modest (+0.089 or +3.4%), the impact on the population average is substantial (+0.492 or +22.8%). This indicates that the GA is highly effective at lifting the overall quality of the population, transforming a pool of mixed quality into a consistently high-quality set of candidates, even if the "genius" outlier improves only incrementally.

This case demonstrates that under favorable conditions, the genetic algorithm can effectively refine the population, improving both fitness and poetic quality over generations.

E Complete Algorithm

Fitness Function. For each poem P, we compute:

$$F_{\text{additive}}(P) = \frac{\text{Info}(P) + \text{Aes}(P)}{10} + \text{Prosody}(P)$$
$$F_{\text{gating}}(P) = (\text{Info}(P) + \text{Aes}(P)) \times \text{Prosody}(P)$$

where $\text{Info}(P), \text{Aes}(P) \in [1, 10]$ are LLM-judge scores, and $\text{Prosody}(P) \in [0, 1]$ is rule-based prosody accuracy. We report additive fitness in the main results.

Operators.

- **baseline**: No crossover/mutation (selection only)
- **char**: Character-level crossover (segment swap) + mutation (position swap)
- **word**: Word-level crossover (using jieba) + mutation (word substitution)
- **line**: Line-level crossover (segment swap) + mutation (line reordering)
- **llm**: LLM-guided crossover ("combine best of both parents") + mutation ("refine 1-3 places")

Algorithm 1 Genetic Algorithm for LLM-Based Ci Poetry Generation

Require: Cipai format C, title/topic T, population size $N = 10$, max generations $G = 10$, tournament size $k = 3$, elite size $e = 2$, operator $\mathcal{O} \in \{\text{baseline, char, word, line, llm}\}$
Ensure: Best poem P_{best}

Phase 1: Initialization
Population $\mathcal{P} \leftarrow \text{GenerateCandidates}(C, T, N)$ ▷ Zero-shot LLM sampling
for each poem $P_i \in \mathcal{P}$ **do**
    $F_i \leftarrow \text{EvaluateFitness}(P_i, C, T)$ ▷ LLM judge + prosody checker
end for
$\mathcal{P} \leftarrow \text{SortByFitness}(\mathcal{P})$
$P_{\text{best}} \leftarrow \mathcal{P}[0]$ ▷ Track global best
Phase 2: Evolution Loop
for generation $g = 1$ to G **do**
    ▷ 2a. Selection
    $\mathcal{P}_{\text{elite}} \leftarrow \mathcal{P}[0 : e]$ ▷ Elitism: preserve top-e
    $\mathcal{P}_{\text{parents}} \leftarrow \emptyset$
    while $|\mathcal{P}_{\text{parents}}| < N - e$ **do**
        $\mathcal{T} \leftarrow \text{TournamentSelect}(\mathcal{P}, k)$ ▷ Random k candidates
        $\mathcal{P}_{\text{parents}} \leftarrow \mathcal{P}_{\text{parents}} \cup \{\text{winner}(\mathcal{T})\}$
    end while
    if $\mathcal{O} = \text{baseline}$ **then** ▷ Baseline: Selection only, no evolution
        $\mathcal{P}_{\text{new}} \leftarrow \mathcal{P}_{\text{elite}} \cup \mathcal{P}_{\text{parents}}$
    else ▷ 2b. Evolution (Crossover + Mutation)
        $\mathcal{P}_{\text{offspring}} \leftarrow \emptyset$
        for pair (P_a, P_b) in $\text{RandomPairs}(\mathcal{P}_{\text{parents}})$ **do**
            $P_{\text{cross}} \leftarrow \text{Crossover}(P_a, P_b, \mathcal{O})$ ▷ Operator-specific
            $P_{\text{mut}} \leftarrow \text{Mutate}(P_{\text{cross}}, \mathcal{O})$ ▷ Operator-specific
            $\mathcal{P}_{\text{offspring}} \leftarrow \mathcal{P}_{\text{offspring}} \cup \{P_{\text{mut}}\}$
        end for
        $\mathcal{P}_{\text{new}} \leftarrow \mathcal{P}_{\text{elite}} \cup \mathcal{P}_{\text{offspring}}$
    end if
    ▷ 2c. Evaluation
    for each poem $P_i \in \mathcal{P}_{\text{new}}$ **do**
        if F_i not cached **then**
            $F_i \leftarrow \text{EvaluateFitness}(P_i, C, T)$
        end if
    end for
    $\mathcal{P} \leftarrow \text{SortByFitness}(\mathcal{P}_{\text{new}})$
    if $\mathcal{P}[0].fitness > P_{\text{best}}.fitness$ **then**
        $P_{\text{best}} \leftarrow \mathcal{P}[0]$
    end if
end for
return P_{best}