MuCPT: Music-related Natural Language Model Continued Pretraining

Kai Tian[1]* Yirong Mao[2] Wendong Bi[2] Hanjie Wang[2] Que Wenhui[2]†

[1] Tsinghua University
[2] WeChat, Tencent Inc., Beijing, China
tk23@mails.tsinghua.edu.cn
{erongmao,wendongbi,hankinwang,victorque}@tencent.com

Abstract

Large language models perform strongly on general tasks but remain constrained in specialized settings such as music,particularly in the music–entertainment domain, where corpus scale, purity, and the match between data and training objectives are critical. We address this by constructing a large, music-related natural language corpus (40B tokens) that combines open source and in-house data, and by implementing a domain-first data pipeline: a lightweight classifier filters and weights in-domain text, followed by multi-stage cleaning, de-duplication, and privacy-preserving masking. We further integrate multi-source music text with associated metadata to form a broader, better-structured foundation of domain knowledge. On the training side, we introduce reference-model (RM)–based token-level soft scoring for quality control: a unified loss-ratio criterion is used both for data selection and for dynamic down-weighting during optimization, reducing noise gradients and amplifying task-aligned signals, thereby enabling more effective music-domain continued pretraining and alignment. To assess factuality, we design the MusicSimpleQA benchmark, which adopts short, single-answer prompts with automated agreement scoring. Beyond the benchmark design, we conduct systematic comparisons along the axes of data composition. Overall, this work advances both the right corpus and the right objective, offering a scalable data–training framework and a reusable evaluation tool for building domain LLMs in the music field.

1 Introduction

Large language models excel on broad text generation and understanding tasks, yet their effectiveness in specialized domains remains constrained by domain coverage and data quality[5, 9, 20]. Music is a salient example: existing music-related natural language models demonstrate promise but are trained on relatively small or mixed-domain corpora, limiting factual coverage of artists, songs, and descriptors that matter in real applications[26]. To advance music-domain modeling, we *build a large music-related natural language dataset* and *train on it with an objective aligned to the domain* so that specialization improves factual music QA without sacrificing general ability.

Our approach starts from data. We curate a **40B tokens** music-domain corpus including two parts: (1) **Matrix-music dataset** (20B tokens) and the **WeChat-music dataset** (20B tokens). The former is mined from Matrix[28] which is a public *diverse and bilingual* pretraining dataset with roughly *4.5T*

*This paper was completed by Kai Tian during his internship at Tencent.
†Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI for Music.

tokens. The latter WeChat Dataset is curated from billions of WeChat articles and comment threads with entity-level links among singers, songs, and fine-grained passages.

To effectively collect music-related data from the above huge text sources, we implement a scalable processing pipeline: a lite domain-first classifier is trained to filter and weight generic web data, a multi-stage cleaning stack removes duplicates and low-salience spans, and a high-relevance WeChat mining route with entity-level linking. The result is an end-to-end data foundation—the **Matrix-music dataset** plus the **WeChat-music Dataset**—that supports continued pretraining and instruction tuning in a unified and reproducible manner.

On the training side, we introduce a *token-level soft scoring* method for fine-grained quality control, where a *reference model* (RM) is first trained in a high-quality dataset and its per-token likelihoods are used to score the full corpus. In this way, we can (i) *filter* low-quality tokens via a loss-ratio criterion and (ii) *downweight* unreliable positions during optimization with an RM-normalized objective.

To measure progress on factuality, we adopt MusicSimpleQA, a short-form, single-answer benchmark emphasizing verifiable facts[24, 11, 7, 15, 27]. We use an automated agreement score to compute accuracy, enabling efficient and replicable evaluation. Empirically, our 32B domain-continued model (**Qwen2.5-32B-MuCPT**) reaches **0.7759** accuracy, surpassing GPT-4o[8], Qwen3-235B-A22B-Instruct[25], and DeepSeek-v3[13] on this task; notably, gains over the strongest baseline are +0.022 absolute (+2.92).

- **A scalable music-domain data pipeline.** We combine large open and in-house sources with domain-first filtering, high-relevance mining, and entity-aware processing to construct a unified corpus for music continued pretraining.
- **token-level soft scoring for quality control.** We propose an RM-based loss normalization and filtering strategy that removes or downweights low-quality tokens using a single scoring criterion for both selection and optimization.
- **Factual music QA evaluation.** Using MUSICSIMPLEQA, our 32B model achieves state-of-the-art accuracy among compared systems on factual music QA, and we provide training-paradigm and data-recipe comparisons that illuminate where gains come from.

2 Dataset Curation

To enable continual pretraining in the music-related natural language domain, we adopt a domain-first data pipeline: a music-domain classifier is trained to filter and weight large generic corpora. We then apply multi-stage cleaning (language normalization, lightweight quality scoring, near-duplicate removal, privacy masking) to mitigate domain drift.

For the in-house WeChat Dataset, we anchor retrieval to top ten millions song names to mine related candidate articles, apply in-domain filtering, and perform entity-level alignment among singers, songs, and fine-grained passages. In parallel, we construct multi-source *song–text* alignments: 3.5M song–snippet pairs covering 1.7M songs and comments coverage for 0.86M songs. Each song is further associated with weakly supervised tags (mood, genre, instrumentation, BPM) derived via a rules+statistics+LLM pipeline. We analyze the singer–song–document tri-graph and modestly upsample tails and new releases to curb head bias. This end-to-end pipeline provides fine-grained, verifiable supervision that matches the factuality emphasis of MUSICSIMPLEQA. Full implementation details and statistics appear in the appendix C.

3 Token-level soft scoring and selection with Reference Model (RM)

This section describes our token-level soft scoring for quality control for the pretraining corpus in the music domain. Building on the insight that not all tokens are equally useful, RHO-1[12] selects high-information tokens via a Reference Model (RM) and drop out uninformative tokens directly. We borrow this idea but go further by using RM scores to dynamically down-weight tokens in a soft way instead of filtering out tokens in a hard way in RHO-1. We follow a simple pipeline: (i) select a high-quality seed set, (ii) train a language model on this seed to serve as a RM that fits the target distribution, and (iii) use the RM to score tokens in the full corpus and apply filtering or weighting accordingly.

Formally, let x_t denote the token at position t and $x_{<t}$ its left context. We use the standard autoregressive negative log-likelihood (NLL) for the RM:

$$\mathrm{CE}_{\mathrm{RM}}(x_t) = -\log p_{\mathrm{RM}}(x_t \mid x_{<t}).$$

The RM is trained on high-quality seed set from WeChat wiki corpus of singers, songs and music-related entities.

For the tokens from music-domain data, we apply RM-normalized loss that reduces the weight in a soft way when the RM loss is high. Specifically, the loss is

$$L_d(x_t) \;=\; -\alpha \, \frac{\log p(x_t \mid x_{<t})}{\log p_{\mathrm{RM}}(x_t \mid x_{<t})},$$

where $\alpha > 0$ is a scaling coefficient. Intuitively, where the RM loss at a token is large, meaning that the token departs from the distribution of the high-quality seed set, the denominator increases and the effective contribution of that token is reduced. As the following simple example shows the gray noisy words would be down-weighted while training.

A real exemplar short article with noisy token: <Try Everything> is a song performed by Colombian singer Shakira and serves as the theme song of the animated movie Zootopia. When I first watched Zootopia, I was deeply impressed by this theme song. After hearing the cover version by the One Voice Children's Choir, I realized that this song is not only catchy in melody, but also has lyrics that are youthful, sunny, and uplifting. Click the border to bring up the video toolbar and scan the QR code to follow us to get more exciting content. Our address is: xxx, and phone is xxx.

Figure 1: A real exemplar short article with noisy token

To mitigate catastrophic forgetting in general knowledge, general-domain samples are mixed where we keep the usual autoregressive loss unchanged:

$$L_g(x_t) \;=\; -\log p(x_t \mid x_{<t}).$$

Thus, the music-domain and general-domain samples are trained with different supervisions.

4 Experiments

4.1 Factual Music QA on MusicSimpleQA: Setup and Findings

Training Configuration. Models are continual pre-trained in Qwen2.5 series models. Matrix-music dataset and WeChat-music dataset are combined as the final music-related natural language domain corpus which contains 40B tokens in total. To avoid catastrophic forgetting in general knowledge, data from general-domain pretraining dataset UltraFineWeb[23] is sampled with meticulous mixture ratio tuning in small LLMs. We train for 2 epochs with 256 H20 GPUs, an initial learning rate of 6×10^{-5}, cosine scheduling with a minimum learning rate of 3×10^{-5}, and a warmup over the first 0.05% of steps.

Evaluation. We evaluate factual music knowledge with the **MusicSimpleQA** benchmark, which follows a short-form, single-answer design emphasizing verifiable facts. The final set contains 500 questions: 300 are drawn from currently popular artists to ensure practical relevance, and 200 are sampled by popularity evenly to broaden genre and era coverage. Evaluation is automated: a strong LLM (DeepSeek-v3) measures agreement between each model's prediction and the reference answer to produce an accuracy score. This construction enables efficient, replicable assessment of factuality in the music domain.

Models	MusicSimpleQA
GPT-4o	0.6632
DeepSeek-v3	0.7539
Qwen3-235B-A22B-Instruct	0.6719
Qwen2.5-32B-Instruct	0.3599
Qwen2.5-32B-MuCPT	**0.7759**

Table 1: Results on MusicSimpleQA (accuracy; higher is better).

Results. We compare GPT-4o, DeepSeek-v3, Qwen3-235B-A22B-Instruct, Qwen2.5-32B-Instruct[21], and our domain-continued model **Qwen2.5-32B-MuCPT**. Results are shown in Table 1: GPT-4o 0.6632, DeepSeek-v3 0.7539, Qwen3-235B-A22B-Instruct 0.6719, Qwen2.5-32B-Instruct 0.3599, and **Qwen2.5-32B-MuCPT 0.7759**. Relative to the strongest baseline (DeepSeek-v3),

our model improves by +0.0220 absolute (+2.92%). Gains vs. GPT-4o, Qwen3-235B-A22B-Instruct, and Qwen2.5-32B-Instruct are +16.99%, +15.48%, and +115.6% respectively. These results indicate that domain-adaptive data and objectives can outweigh sheer parameter count on factual music QA. Notably, a 32B model with strong domain adaptation outperforms a 235B instruction model and GPT-4o on this task, reinforcing the effectiveness of domain-incremental training when data and objectives are tightly matched to the target domain.

4.2 Ablation study for our token-level soft scoring

In this part, we compare our token-level soft scoring with plain next token prediction and RHO-1[12]. RHO-1 implements selective language modeling where a reference model scores tokens in the domain corpus with excess loss and drop out tokens in a hard way. The plain next token prediction treats every tokens equally.

Results are summarized in Table 2, under the same inference and evaluation protocol. At **1.5B** model scale, RHO-1 improves over *NextTokenPrediction* (0.4439 $\rightarrow$ 0.4759; +7.21%), while **MuCPT** reaches **0.5259** (+10.51% vs. RHO-1; +18.47% vs. NextTokenPrediction). At **7B** model scale, RHO-1 raises accuracy from 0.5499 to 0.5699 (+3.64%); **MuCPT** attains **0.6119** (+7.37% vs. RHO-1; +11.28% vs. NextTokenPrediction).

Method	MusicSimpleQA
Qwen-1.5B-NextTokenPrediction	0.4439
Qwen-1.5B-RHO-1	0.4759
Qwen-1.5B-MuCPT (ours)	**0.5259**
Qwen-7B-NextTokenPrediction	0.5499
Qwen-7B-RHO-1	0.5699
Qwen-7B-MuCPT (ours)	**0.6119**

Table 2: MusicSimpleQA accuracy for NextTokenPrediction, RHO-1, and MuCPT at 1.5B and 7B scales.

Both RHO-1 and our token-level soft scoring in **MuCPT** surpass the next token prediction, showing that the not all tokens are equally contributed to the domain task. RHO-1 follows a "select-before-learn" principle that drops out noisy or uninformative tokens which may interrupting semantic coherence. Instead, our **MuCPT** retains token-level soft scoring for quality control where the semantic coherence is kept well.

4.3 Comparing Data Recipes for Domain-Continued Pretraining

We compare three different data sources in a fixed budget: (i) **BAAI/IndustryCorpus2-film-entertainment**[19] (a film & entertainment subset that is broader than music) , (ii) **Matrix-music dataset** [28], and (iii) **WeChat-music dataset**. Results are reported in Table 3.

WeChat-music achieves **0.4579**, whereas **Matrix-music** obtains 0.3659 and **IndustryCorpus2_film_entertainment** reaches 0.2899. Relative to Matrix-music, WeChat-music yields an absolute gain of 0.0920 (+25.14%); relative to IndustryCorpus2_film_entertainment, the gain is 0.1680 (+57.95%). Matrix-music also outperforms the broader film/entertainment subset by 0.0760 (+26.22%).

Models	MusicSimpleQA
IndustryCorpus2[19]	0.2899
Matrix-music (ours)	0.3659
WeChat-music (ours)	**0.4579**

Table 3: MusicSimpleQA accuracy with 7B-token data recipes for a 1.5B model.

Under the same token budget, the differences likely reflect a combination of factors such as task–distribution match, corpus quality and cleaning standards, sampling choices, and noise levels. In general, data recipes that align more closely with the knowledge distribution required by music QA tend to be more stable on MUSICSIMPLEQA.

5 Conclusion

We presented a unified route to specialization in the music–entertainment domain that couples a *domain-first* data pipeline with *reference-model, token-level soft scoring for quality control*, and evaluates progress with the MusicSimpleQA benchmark for short-form factuality. The data pipeline prioritizes in-domain signals during sampling, integrates rigorous multi-stage cleaning and deduplication, and organizes high-relevance sources into a coherent substrate for continued pretraining and alignment. The RM-normalized objective provides a single, per-token scoring principle that governs both selection and dynamic down-weighting, producing cleaner gradients and enabling efficient specialization while largely preserving general world knowledge. Taken together, these components deliver a scalable and auditable recipe for building music-domain LLMs that treat "music as a second language" without requiring task-specific modules.

References

[1] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

[2] Keshav Bhandari, Abhinaba Roy, Kyra Wang, Geeta Puri, Simon Colton, and Dorien Herremans. Text2midi: Generating symbolic music from captions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23478–23486, 2025.

[3] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. *Advances in Neural Information Processing Systems*, 36:47704–47720, 2023.

[4] Shuangrui Ding, Zihan Liu, Xiaoyi Dong, Pan Zhang, Rui Qian, Junhao Huang, Conghui He, Dahua Lin, and Jiaqi Wang. Songcomposer: A large language model for lyric and melody generation in song composition. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7108–7127, 2025.

[5] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024.

[6] Jeff Ens and Philippe Pasquier. Mmm: Exploring conditional multi-track music generation with the transformer. *arXiv preprint arXiv:2008.06048*, 2020.

[7] Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Yao Fu, et al. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. *Advances in Neural Information Processing Systems*, 36:62991–63010, 2023.

[8] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[9] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

[10] Shun Lei, Yixuan Zhou, Boshi Tang, Max WY Lam, Hangyu Liu, Jingcheng Wu, Shiyin Kang, Zhiyong Wu, Helen Meng, et al. Songcreator: Lyrics-based universal song generation. *Advances in Neural Information Processing Systems*, 37:80107–80140, 2024.

[11] Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. Cmmlu: Measuring massive multitask language understanding in chinese. *arXiv preprint arXiv:2306.09212*, 2023.

[12] Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, et al. Rho-1: Not all tokens are what you need. *arXiv preprint arXiv:2404.07965*, 2024.

[13] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[14] Hong-Hsiang Liu and Yi-Wen Liu. Agent-driven large language models for mandarin lyric generation. In *2024 27th Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA)*, pages 1–6. IEEE, 2024.

[15] Xinwei Long, Kai Tian, Peng Xu, Guoli Jia, Jingxuan Li, Sa Yang, Yihua Shao, Kaiyan Zhang, Che Jiang, Hao Xu, et al. Adsqa: Towards advertisement video understanding. *arXiv preprint arXiv:2509.08621*, 2025.

[16] Yinghao Ma, Anders Øland, Anton Ragni, Bleiz MacSen Del Sette, Charalampos Saitis, Chris Donahue, Chenghua Lin, Christos Plachouras, Emmanouil Benetos, Elona Shatri, et al. Foundation models for music: A survey. *arXiv preprint arXiv:2408.14340*, 2024.

[17] Philippe Pasquier, Jeff Ens, Nathan Fradet, Paul Triana, Davide Rizzotti, Jean-Baptiste Rolland, and Maryam Safi. Midi-gpt: A controllable generative model for computer-assisted multitrack music composition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1474–1482, 2025.

[18] Seungyeon Rhyu, Kichang Yang, Sungjun Cho, Jaehyeon Kim, Kyogu Lee, and Moontae Lee. Practical and reproducible symbolic music generation by large language models with structural embeddings. *arXiv preprint arXiv:2407.19900*, 2024.

[19] Xiaofeng Shi, Lulu Zhao, Hua Zhou, and Donglin Hao. Industrycorpus2, 2024.

[20] Qwen Team. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

[21] Qwen Team. Qwen2.5: A party of foundation models, September 2024.

[22] Yufei Tian, Anjali Narayan-Chen, Shereen Oraby, Alessandra Cervone, Gunnar Sigurdsson, Chenyang Tao, Wenbo Zhao, Yiwen Chen, Tagyoung Chung, Jing Huang, et al. Unsupervised melody-to-lyric generation. *arXiv preprint arXiv:2305.19228*, 2023.

[23] Yudong Wang, Zixuan Fu, Jie Cai, Peijun Tang, Hongya Lyu, Yewei Fang, Zhi Zheng, Jie Zhou, Guoyang Zeng, Chaojun Xiao, et al. Ultra-fineweb: Efficient data filtering and verification for high-quality llm training data. *arXiv preprint arXiv:2505.05427*, 2025.

[24] Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

[25] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[26] Ruibin Yuan, Hanfeng Lin, Yi Wang, Zeyue Tian, Shangda Wu, Tianhao Shen, Ge Zhang, Yuhang Wu, Cong Liu, Ziya Zhou, et al. Chatmusician: Understanding and generating music intrinsically with llm. *arXiv preprint arXiv:2402.16153*, 2024.

[27] Sihang Zeng, Kai Tian, Kaiyan Zhang, Junqi Gao, Runze Liu, Sa Yang, Jingxuan Li, Xinwei Long, Jiaheng Ma, Biqing Qi, et al. Reviewrl: Towards automated scientific review with rl. *arXiv preprint arXiv:2508.10308*, 2025.

[28] Ge Zhang, Scott Qu, Jiaheng Liu, Chenchen Zhang, Chenghua Lin, Chou Leuang Yu, Danny Pan, Esther Cheng, Jie Liu, Qunshu Lin, et al. Map-neo: Highly capable and transparent bilingual large language model series. *arXiv preprint arXiv:2405.19327*, 2024.

[29] Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Pengfei Li, et al. A survey of reinforcement learning for large reasoning models. *arXiv preprint arXiv:2509.08827*, 2025.

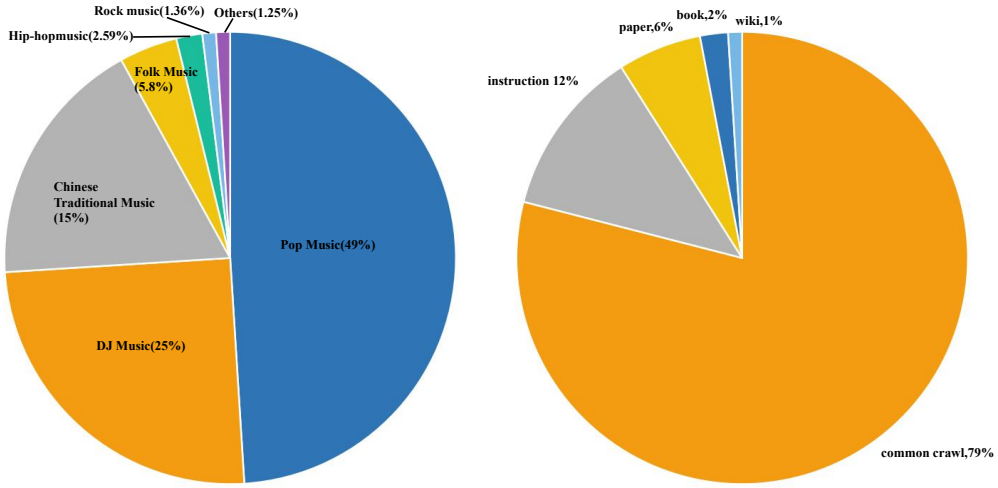
A Related Work

In recent years, propelled by breakthroughs in generative modeling, large language models (LLMs) have been introduced into the music domain, where incremental pre-training equips them with specialized musical knowledge and skills[26, 16, 29]. Continuing to train an LLM on domain-specific data—so-called domain-incremental training—has become the mainstream strategy for enhancing professional competence. For example, the ChatMusician[26] model is further pre-trained on top of LLaMA-2 and fine-tuned with the text-friendly ABC notation, enabling the model to treat music as a "second language" to be understood and generated. This approach exploits MusicPile[26], a 4-billion-token music-language corpus that combines multimodal text—web-scraped music encyclopedias, music-theory books, YouTube metadata, and lyrics—with a large collection of scores encoded in ABC.

To align musical and linguistic information, ChatMusician cleans and converts the data, using plain-text scores (ABC) to avoid extra multimodal modules. The model also adopts an incremental scheme that blends synthetic and public data, for instance augmenting the corpus with GPT-4-generated music Q&A pairs and track summaries. A similar strategy appears in projects such as "MusicGPT", which represent music as discrete symbol sequences and continue training a causal language model on those sequences: researchers feed large MIDI libraries, encoded with formats such as MMM Track, into a GPT-2-style model by flattening multitrack music into linear sequences[6, 17, 18]. These results show that sustained pre-training on music data and careful data alignment allow an LLM to internalize music theory and symbolic structure, compressing and representing musical information without additional task-specific modules[2].

Building on these pre-training strategies, LLMs have already demonstrated a range of capabilities in the music domain, spanning creation, analysis, and assistance. In music creation, large models can generate new compositions from user-provided text or other cues. MusicLM and MusicGen, for instance, enable text-to-music generation, producing audio in a requested style directly from natural-language descriptions[3, 1]. Their outputs cover a wide spectrum—from Baroque choral works to modern pop—while maintaining musical correctness and coherence. LLMs similarly excel at lyric writing: by learning from vast lyric corpora, they can craft rhymed lyrics on a given topic or emotional tone[4, 14, 22]. In music analysis, LLMs demonstrate an understanding of and reasoning about musical knowledge—answering theory questions, parsing score structures, and recognizing style or affect after domain-incremental training. As auxiliary tools, large language models are becoming valuable assistants in music creation and education[10]. Conversationally, they retrieve musical information (e.g., historical facts, piece backgrounds) or act as virtual teachers answering theory queries; combined with tool invocation, they can call specialized audio-processing or generation models to accomplish complex tasks. Overall, whether composing autonomously or supporting human creativity and learning, LLMs are bringing the vision of "music as dialogue" to life.

B User Preference Analysis and the Motivation for Music–Entertainment CPT



(a) User Preference: Pop Dominance (Motivating Music-Entertainment Continue Training)

(b) Composition of the Matrix-music Corpus (Mined Proportions)

Figure 2: User preferences and corpus composition. (a) shows that most users prefer pop music, motivating our focus on music-entertainment continual pretraining; (b) summarizes the proportional makeup of the open-source music corpus mined for this work.

Figure 2 summarizes aggregate user-behavior signals (e.g., plays, searches, saves, skips) and shows that **pop music dominates overall preference**, far exceeding other categories (DJ/electronic, folk, hip-hop, rock, Chinese traditional, etc.). Guided by this observation, we prioritize *music–entertainment* scenarios for continual pretraining (CPT) and construct our dataset accordingly: sources and samples

that are closely tied to everyday entertainment-oriented music consumption receive higher sampling priority and stricter cleaning.

C Details of Data Curation

Our objective in this stage is continual learning for the music domain. We adopt a domain-first strategy: we first train a domain classifier and use it to filter and weight large-scale generic corpora before continued pretraining and task-oriented augmentation. The classifier is based on Qwen2.5-0.5B and trained as a balanced binary model with roughly 250k positive and 250k negative examples. Positive instances include content about songs, artists, catalogs, reviews, styles, and production notes; negatives span diverse non-music topics. During sampling, the classifier's confidence serves as a routing signal that assigns higher weights to in-domain text and gates expensive instruction-style or alignment-oriented augmentations.

For the Matrix-music dataset, we aggregate and clean about 20 billion tokens of open data . Book contributes roughly 332 million tokens, common crawl about 17.4 billion, instruction about 2.7 billion, paper about 1.3 billion, and wiki about 270 million. Cleaning follows a multi-stage pipeline: language identification and normalization; heuristic and lightweight quality scoring to prune templated or low-salience spans; locality-sensitive hashing/MinHash for near-duplicate removal; and privacy-preserving masking for potential identifiers.

Figure 2 presents the source composition of the cleaned and normalized Matrix-music corpus: Common Crawl dominates (79%), followed by instruction (12%), paper (6%), book (2%), and wiki (1%). The figure characterizes the *base distribution* adopted after a multi-stage cleaning pipeline ("broad coverage + lightweight quality scoring + near-duplicate removal + privacy masking").

To strengthen high-relevance Chinese coverage, we build a WeChat-music Dataset via a mining pipeline. Using approximately ten millions top song names from the last year (from a wechat-listen source) as anchors, we retrieve and match candidate articles, apply the music classifier for in-domain filtering, and then perform entity-level alignment that links singers and song titles to fine-grained passages with disambiguation. This pipeline yields roughly 20 billion tokens across about 20.5 million documents. We analyze the singer–song–document tri-graph to understand coverage and long-tail effects, and we modestly upsample tail and newly released works in subsequent sampling to reduce head-content dominance.

To directly learn alignments between text and musical works, we construct song–text pairs from multiple sources. On the comment side, we select reliable public comments and filter coarse noise and toxicity with an LLM, resulting in coverage for roughly 0.86 million songs; these naturally encode perceptual descriptors such as timbre, mood, style, and instrumentation, providing supervised attribute-alignment signals. On the article side, we segment articles into fragments, yielding about 3.5 million "song–snippet" pairs that cover approximately 1.7 million songs. For each song, we further generate or aggregate content-understanding tags—mood, genre, instrument, and BPM—using a multimodal music understanding model; tagging follows a hybrid "rules + statistics + LLM cross-check" procedure: lexicon- and rule-based retrieval to form candidates, statistical co-occurrence and contrastive models for scoring, and an LLM pass for consistency and readability checks.

In summary, we link preference-driven domain filtering, scalable high-relevance mining, entity-level alignment with verifiable task construction. The $\sim$22B-token base corpus ensures breadth, the $\sim$21B-token WeChat-music Dataset supplies high-relevance Chinese context, and multi-source song/comment/article pairs plus mood/genre/instrument/bpm tags provide fine-grained supervision.

D Construction of the Music Factuality Question-Answer Evaluation Set

In the construction of the music factuality question-answer evaluation set, we drew inspiration from OpenAI's SimpleQA evaluation set and the approach outlined in [24], with the aim of developing a tool capable of efficiently and automatically evaluating the performance of large language models in the music domain — the MusicSimpleQA evaluation set. The core design of this evaluation set centers around the concept of "factuality," constructing a series of concise, clear, and unique question-answer pairs, ensuring that each question has a single, unambiguous answer, which is crucial for assessing the model's ability to handle real-world factual knowledge.

To generate the questions, we first employed the Deepseek-v3 model to automatically extract information from extensive singer encyclopedic data and generate common factual questions related to music artists. These questions cover basic biographical and factual information about the artists, such as "Where is Jay Chou from?", "What is the name of Jay Chou's first solo album?", "Which film did

Jay Chou win the Best Newcomer Award at the Golden Horse Awards?", and "Which talent show did Zhou Shen debut in?". The answers to these questions are unique and verifiable facts.
Following the generation of questions, we conducted a filtering and validation process to ensure that the answers were both unique and clear. Specifically, all generated questions underwent manual or automated consistency checks, where those questions that had multiple possible answers or were ambiguously phrased were filtered out, ensuring that only questions with clear and unique answers remained. This filtering process ensures that the evaluation set maintains high quality and precision, preventing ambiguous or non-unique answers from interfering with the evaluation results.
The final evaluation set consists of 500 questions. Of these, 300 questions are based on popular artists within the current music industry to ensure that the evaluation set is relevant and realistic. The remaining 200 questions are popularity evenly sampled, covering different genres and eras, to increase the diversity of the set and to assess the model's ability to handle a wide range of musical knowledge.
The evaluation method involves comparing the standard answer with the model's predicted answer. The DeepSeek-v3 model is used to automatically evaluate the consistency between the standard and predicted answers. Specifically, DeepSeek-v3 compares the matching degree between the given standard answers and the model-generated answers, providing an automatic score for the prediction's accuracy.
By employing this automated and efficient evaluation method, we provide a new evaluation tool for large language models in the music domain — MusicSimpleQA. This evaluation set allows us to comprehensively assess the performance of models in answering factual questions in the music domain. The construction of this evaluation set not only contributes to the development of models in the music field but also provides a replicable framework for future music data processing and knowledge inference tasks.

E General-Ability Evaluation: Impact on World Knowledge

Models	C-Eval	CMMLU
Qwen2.5-32B-Instruct	86.46	85.79
Qwen2.5-32B-MuCPT	84.17	84.34

Table 4: World-knowledge accuracy on C-EVAL and CMMLU for Qwen2.5-32B-Instruct vs. Qwen2.5-32B-MuCPT.

We assess whether music-domain continual pretraining harms broad world knowledge by comparing **Qwen2.5-32B-Instruct** and **Qwen2.5-32B-MuCPT** on two comprehensive benchmarks, C-EVAL and CMMLU. As summarized in Table 4, MuCPT shows a small drop on C-EVAL (86.46 $\rightarrow$ 84.17; absolute -2.29, relative -2.65%) and a similarly minor drop on CMMLU (85.79 $\rightarrow$ 84.34; absolute -1.45, relative -1.69%). Overall, both changes are within $< 3\%$, indicating no catastrophic forgetting of world knowledge. In practice, MuCPT maintains general ability while delivering substantial gains on music-domain factual QA, demonstrating a favorable specialization–generalization trade-off.