MuseAgent: Grounded Multimodal Understanding of Music Scores and Performance Audio

Anonymous ACL submission

Abstract

Despite recent advances in multimodal large language models (MLLMs), their ability to understand and interact with music remains limited. Music understanding requires grounded reasoning over symbolic scores and expressive performance audio, which general-purpose MLLMs often fail to handle due to insufficient perceptual grounding. We introduce MuseAgent, a music-centric multimodal agent that augments language models with structured symbolic representations derived from sheet music images and performance audio. By integrating optical music recognition and automatic music transcription modules, MuseAgent enables multi-step reasoning and interaction over fine-grained musical content. To systematically evaluate music understanding capabilities, we further propose MuseBench, a benchmark covering music theory reasoning, score interpretation, and performance-level analysis across text, image, and audio modalities. Experiments show that existing MLLMs perform poorly on these tasks, while MuseAgent achieves substantial improvements, highlighting the importance of structured multimodal grounding for interactive music understanding.

1 Introduction

"O Muses, O high genius, now help me!"
— Dante, *Inferno*, Canto II, line 7

Music is a structured yet expressive domain, making it a compelling testbed for artificial intelligence. It spans symbolic representations such as notated scores and expressive acoustic realizations such as performances, each requiring distinct perceptual and reasoning capabilities. As both a formal system and an emotional medium, music challenges AI models to reason across modalities with high precision and contextual nuance (Essid and Richard, 2012; Corrêard et al., 2021). These requirements expose fundamental limitations of current multimodal systems, particularly large language models that lack structured grounding in symbolic and temporal representations.

Among musical forms, piano music occupies a uniquely central role in the Western repertoire, supported by centuries of standardized notation and extensive performance archives. Its wide pitch range, dense polyphony, and two-hand coordination make piano music a particularly demanding stress test for multimodal reasoning systems (Hawthorne et al., 2019a). Understanding piano music requires jointly modeling complex symbolic structures—such as pitch, rhythm, and dynamics—and expressive performance attributes including timing, articulation, and rubato. This integration is essential for interactive applications such as transcription, accompaniment, digital archiving, and music education (Benetos et al., 2019).

Despite significant progress in isolated tasks, existing approaches remain insufficient for holistic and interactive music understanding. Prior work has advanced single-modality problems such as Optical Music Recognition (OMR) for scores and Automatic Music Transcription (AMT) for audio. Large-scale datasets like MAESTRO (Hawthorne et al., 2019a) and generative models such as MusicLM (Agostinelli et al., 2023) demonstrate the feasibility of modeling symbolic and acoustic music representations, yet they fall short of enabling fine-grained cross-modal reasoning. Recent Multimodal Large Language Models (MLLMs), including GPT-4o (OpenAI, 2023) and Gemini (Google Gemini Team, 2023), promise general cross-modal capabilities but perform poorly on music understanding tasks that require precise symbolic parsing or long-form performance analysis. This failure stems largely from the lack of domain-specific perceptual grounding and the inability to align symbolic and acoustic modalities at high temporal resolution.

Several recent systems have attempted to bridge this gap by augmenting LLMs with external tools. AudioGPT (Huang et al., 2024) and MusicAgent

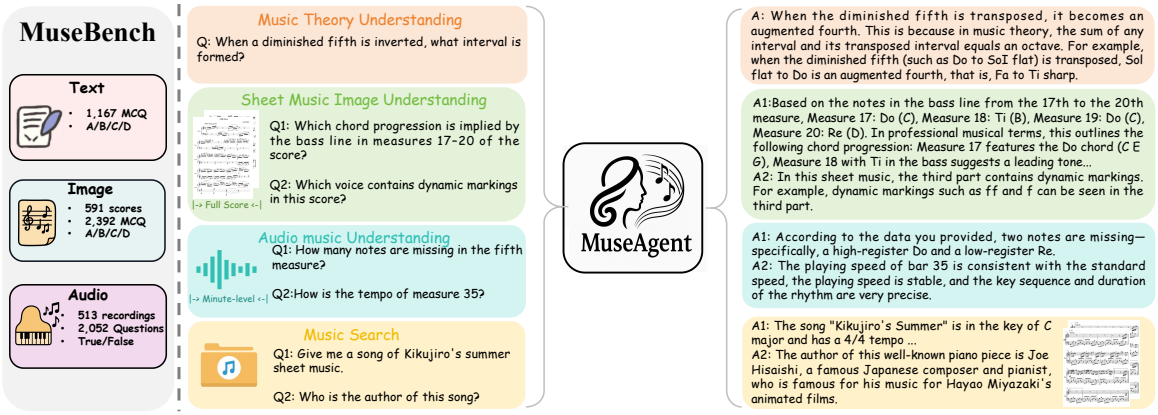

Figure 1: Overview of **MuseBench** and **MuseAgent**. MuseBench consists of multimodal music understanding tasks across text, image, and audio modalities, covering music theory, sheet score analysis, performance interpretation. MuseAgent integrates these modalities via sheet symbolic recognition, audio alignment, and retrieval modules, enabling large language models to answer complex music questions.

(Yu et al., 2023) represent early efforts toward music-aware agents by coupling language models with domain-specific components. However, these systems often struggle with complex notation, long-duration performance recordings, and structured interaction, and they lack systematic evaluation protocols tailored to music understanding. Meanwhile, advances in Retrieval-Augmented Generation (RAG) (Lewis et al., 2020; Guu et al., 2020; Borgeaud et al., 2022) demonstrate that grounding language models in structured external representations can substantially improve domain-specific reasoning and reduce hallucination. Existing multimodal RAG frameworks (Shuster et al., 2022; Luo et al., 2023; Liu et al., 2023b), however, primarily focus on text–vision–speech settings and rarely address music, where symbolic scores and expressive audio must be tightly aligned.

To address these challenges, we propose MuseAgent, a multimodal retrieval-augmented agent designed specifically for music understanding and interaction. MuseAgent integrates a large language model with specialized perceptual front-ends: (i) a measure-wise OMR module that converts sheet music images into symbolic representations such as ABC notation (Yuan et al., 2024a), (ii) an AMT-based performance analysis module that aligns audio recordings with MusicXML scores and extracts expressive features in structured JSON formats, and (iii) a retrieval module that enables both explicit and implicit access to symbolic and audio libraries. These components ground the language model in structured multimodal representations, while a memory bank supports long-context, multi-turn reasoning over musical content.

Evaluating such music understanding agents poses a non-trivial challenge. Existing datasets and benchmarks focus primarily on transcription or generation, and do not assess an agent's ability to reason interactively across symbolic scores and performance audio. To enable systematic evaluation, we introduce MuseBench, the first benchmark designed to assess multimodal music understanding agents. Centered on piano repertoire, MuseBench includes tasks such as score–audio alignment, performance error detection, and expressive deviation analysis, repurposing resources like MAESTRO into high-level reasoning tasks suitable for evaluating agentic music understanding.

Our experiments on MuseBench reveal that general-purpose MLLMs exhibit limited capabilities in fine-grained symbolic and performance-level music reasoning. In contrast, MuseAgent demonstrates substantial improvements on challenging image- and audio-based tasks, validating the effectiveness of structured perceptual grounding and agent-based multimodal reasoning for interactive music understanding.

2 Related Work

Music Understanding Systems and Agents. Recent advances in general-purpose agents, such as ReAct, Auto-GPT, and Gorilla, have demonstrated the effectiveness of tool-augmented reasoning for complex tasks. In the music domain, early systems such as MuseNet (Payne, 2019) and MusicLM (Agostinelli et al., 2023) primarily focus on symbolic or audio music generation, rather

than music understanding or interactive reasoning. More recent music-oriented multimodal language models adopt domain-specific training strategies. For example, MusiLingo (Deng et al., 2024) targets music captioning and question answering via instruction-tuned audio–language modeling, MuMu-LLaMA (Liu et al., 2024) unifies music audio, images, and language within a single multimodal framework, and NotaGPT (Tang et al., 2025) focuses on music notation understanding through a vision–language model. SymphonyNet (Liu et al., 2022) further demonstrates symbolic music generation at orchestral scale.

While these domain-specific models highlight the value of multimodal alignment and specialized training, they are typically designed as monolithic architectures and do not explicitly support structured interaction, tool invocation, or multi-step reasoning. In parallel, agent-based systems such as MusicAgent (Yu et al., 2023) and AudioGPT (Huang et al., 2024) begin to couple large language models with domain-specific tools for processing music inputs. Although promising, these systems are not designed for fine-grained music understanding that requires jointly reasoning over symbolic scores and expressive performance audio, nor do they target complex piano repertoire. In contrast, our proposed *MuseAgent* advances this line of work by introducing a music-centric agent architecture that integrates structured perceptual grounding, retrieval-based reasoning, and agentic orchestration across symbolic and acoustic modalities.

Music Understanding Benchmarks. A critical challenge in developing music understanding agents lies in evaluation. Existing music datasets, such as MAESTRO (Hawthorne et al., 2019b) and URMP (Li et al., 2018), provide aligned score–audio pairs and have been widely used for transcription, alignment, and generation tasks. However, these resources lack task-oriented evaluation protocols that assess high-level reasoning or interactive understanding. Recent multimodal question-answering datasets, including MUSIC-AVQA (Li et al., 2022) and MuChoMusic (Weck et al., 2024), extend music benchmarks toward Q&A settings, but do not emphasize detailed symbolic score interpretation or nuanced performance-level reasoning. In contrast, our proposed *MuseBench* is designed explicitly to evaluate music understanding agents, assessing their ability to reason jointly over text, score images, and performance audio across a unified set of piano-centric tasks.

General Multimodal Language Models. General-purpose multimodal language models, such as GPT-4o (OpenAI, 2023), Gemini (Anil et al., 2023), Qwen (Team, 2024), and LLaVA (Liu et al., 2023a), achieve strong performance on text–vision benchmarks. However, recent studies (Weck et al., 2024) demonstrate that these models struggle with music understanding tasks, particularly when confronted with structured musical notation or expressive performance audio. In the absence of explicit symbolic grounding, such models often rely on linguistic priors or hallucinate musical content. These limitations motivate the need for music-aware agents that integrate domain-specific perception and structured reasoning, as exemplified by our MuseAgent.

Distinctiveness of Our Work. In summary, prior research on music understanding has largely focused on isolated modalities, monolithic domain models, or generation-oriented objectives, while existing benchmarks lack the ability to evaluate interactive, agent-based reasoning over music. Our work uniquely combines a music-centric multimodal agent with an agent-oriented benchmark. MuseAgent enables structured reasoning over both symbolic scores and expressive performance audio, while MuseBench provides a unified evaluation framework tailored to assessing such capabilities. Together, they establish a new foundation for studying fine-grained, interactive multimodal music understanding.

3 MuseAgent

We propose **MuseAgent**, a multimodal retrieval-augmented agent designed for structured and interactive music understanding. Unlike fixed perception–reasoning pipelines, MuseAgent adopts an *agentic orchestration loop* in which a large language model dynamically coordinates domain-specific perceptual modules, retrieval operations, and symbolic reasoning based on user intent. This design enables MuseAgent to perform multi-step reasoning over symbolic scores and expressive performance audio, addressing the challenges posed by agent-oriented music understanding tasks such as those defined in MuseBench.

As illustrated in Figure 2, MuseAgent integrates three core components: (i) perceptual ground-

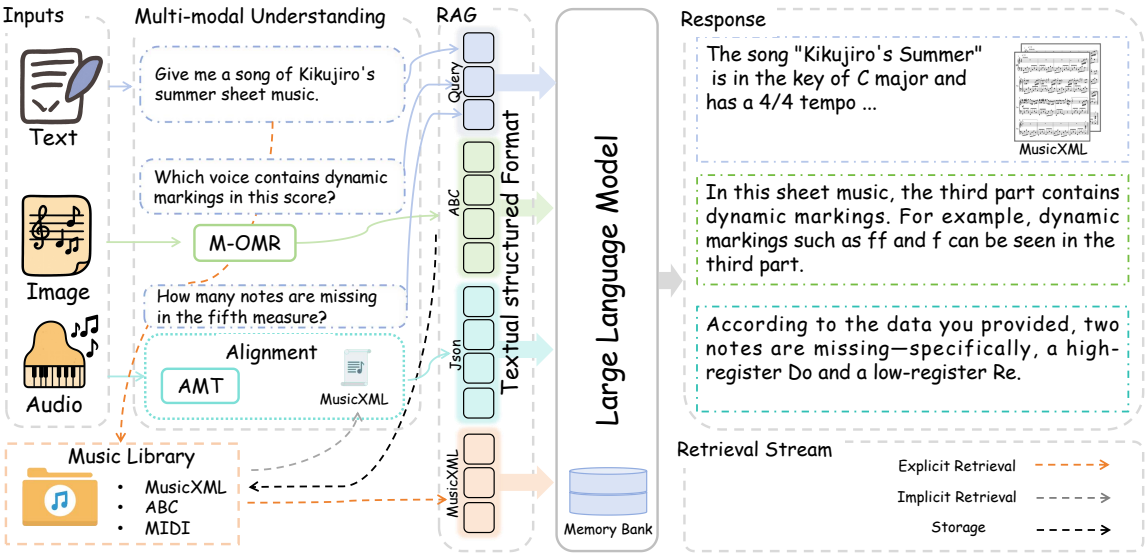

Figure 2: The MuseAgent framework integrates M-OMR, AMT, and music retrieval (explicit/implicit) into a unified large language model (LLM)-based system. Each perceptual module converts raw multimodal inputs into structured symbolic representations (e.g., ABC, MusicXML, JSON), which are incorporated into a retrieval-augmented generation (RAG) pipeline. The LLM acts as an agentic controller that dynamically orchestrates module usage depending on user intent, while a memory bank supports multi-turn dialogue and retrieval of prior outputs.

ing modules that convert raw multimodal inputs into structured symbolic representations, (ii) a retrieval-augmented generation (RAG) mechanism that grounds reasoning in external music knowledge, and (iii) a memory bank that supports long-context, multi-turn interaction.

3.1 Agentic Workflow

At the core of MuseAgent is an LLM-based controller that interprets user queries and dynamically determines which perceptual or retrieval modules to invoke. For example, queries concerning harmonic structure or notation trigger the measure-wise OMR module, while questions about timing, tempo stability, or performance accuracy invoke AMT-based alignment. Retrieval may be initiated explicitly by user requests or implicitly by the agent when additional symbolic context is required. This agentic workflow enables flexible, intent-driven reasoning across modalities rather than static, predefined processing pipelines.

3.2 Measure-wise Optical Music Recognition

A key challenge for multimodal large language models (MLLMs) in music understanding is the modality gap between high-resolution score images and the symbolic reasoning required for musical analysis. Unlike natural images, music scores are densely structured and domain-specific, encoding hierarchical elements such as pitch, rhythm, and dynamics that general vision-language models struggle to interpret directly.

To address the challenge of structured music score recognition, we propose a Measure-wise Optical Music Recognition (M-OMR) module based on a "divide-and-combine" strategy as a perceptual grounding module. The input sheet image is first divided into individual measures using visual layout cues such as staff lines and barlines. Each measure is treated as an independent visual unit, encoded into symbolic representation through localized recognition. These measure-level outputs are then combined to reconstruct the complete musical piece. The final result is expressed in ABC notation, a compact and structured symbolic format well-suited for downstream language model processing.

Unlike prior OMR approach, NotaGPT (Tang et al., 2025), which split score images at the note level and rely on frozen vision and text encoders, we adopt a measure-wise segmentation strategy. The different between them as shown in Figure 3. By treating each measure as a semantic unit, our model preserves musical structure and reduces noise from overly granular splitting. We train a ResNet-based (He et al., 2016) visual encoder, jointly trained with an LSTM (Yu et al., 2019) over the measure sequence to capture intra-score de-

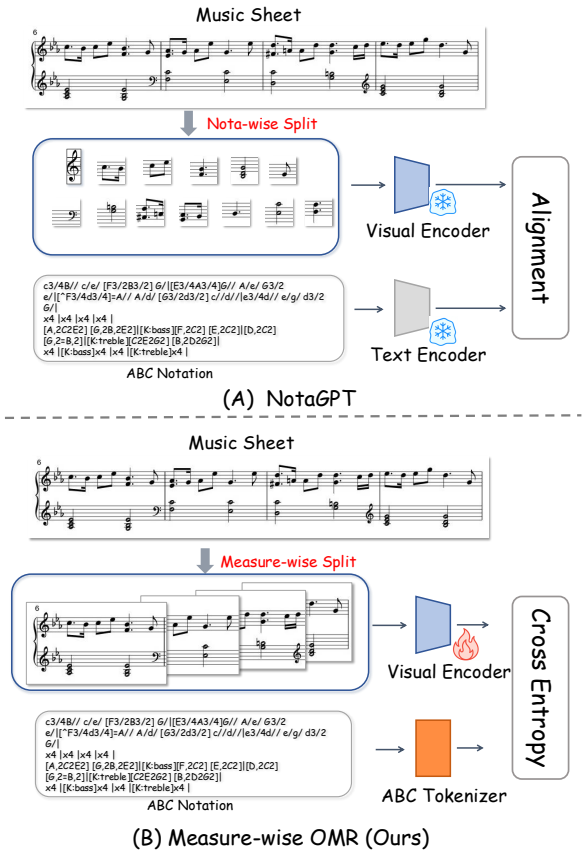

Figure 3: Comparison between (A) *NotaGPT*, which performs note-level segmentation with frozen visual and text encoders, and (B) our proposed *Measure-wise OMR* approach. The flame symbol denotes trainable modules, while the snowflake symbol indicates frozen components.

pendencies. Furthermore, we introduce a custom-designed ABC tokenizer tailored for ABC-notation representation. This tokenizer captures over hundreds of ABC-notation variants of music-specific constructs (e.g., key, meter, chords), producing more compact and structurally meaningful token sequences compared to general-purpose text encoders. More details are shown in Appendix D.

3.3 AMT and Alignment

To understand expressive performance audio, MuseAgent incorporates an Automatic Music Transcription (AMT) module and an audio-to-score alignment component as the audio perceptual grounding module. The AMT module transcribes raw audio into a symbolic representation (e.g., MusicXML) by extracting time-frequency features via the Constant-Q Transform (Schörkhuber and Klapuri, 2010), and applying neural transcription models.

The resulting symbolic sequence is temporally aligned to a reference score using a hierarchical Hidden Markov Model (H-HMM) (Nakamura et al., 2015), which is robust to expressive timing variations, ornaments, and structural deviations such as repeats or skips. The alignment process produces structured outputs in JSON format, capturing onset timings, note correspondences, and expressive parameters.

These alignment outputs are then fused with user prompts and MusicXML files retrieved implicitly from the music library, forming the input to a retrieval-augmented generation (RAG) module. The RAG component composes these multimodal elements into an enriched prompt, enabling the language model to reason over both symbolic and auditory performance data. Implementation details, including model architecture and training configurations, are provided in Appendix E.

3.4 Music Retrieval Module

The MuseAgent supports both explicit and implicit retrieval from a large-scale symbolic music library in formats such as ABC, MusicXML, and MIDI.

For **explicit retrieval**, users can issue direct natural language queries (e.g., "Give me a song of *Kikujiro's Summer*") to fetch matching scores. For **implicit retrieval**, the system performs internal searches conditioned on audio, and sheet context, selecting relevant symbolic files (e.g., auio-paired MusicXML) to be integrated into the the RAG pipeline.

Unlike traditional information retrieval methods, retrieval here is embedded into the agent loop: the LLM may explicitly respond to user queries or implicitly call the retrieval API to ground its reasoning. This design realizes agentic RAG for multimodal music.

3.5 Music Theory Understanding and Dialogue Context

In addition to the aforementioned capabilities, MuseAgent harnesses the intrinsic musical knowledge embedded in large language models to answer music-theoretical questions (e.g., "What interval results from inverting a diminished fifth?" or "Which mode begins on E in the C major scale?"). The effectiveness of this ability may vary across used LLMs. For evaluations of music theory understanding in different LLMs, please refer to Sec. 5.1.2.

The MuseAgent also supports memory capabilities for multi-turn conversations, for which we maintain a lightweight **memory bank** that stores

intermediate module outputs, retrieved files, and previous model responses. The memory bank not only supports multi-turn reasoning, but also enables retrieval of prior structured outputs.

4 MuseBench

To enable systematic evaluation of *music understanding agents*, we introduce **MuseBench**, a multimodal benchmark designed to assess structured and interactive reasoning over music. MuseBench targets agents that integrate perception, retrieval, and language-based reasoning, and evaluates their ability to jointly reason over **text**, **sheet music images**, and **performance audio**. Rather than measuring isolated recognition or generation skills, MuseBench focuses on high-level music understanding tasks that require symbolic grounding, cross-modal alignment, and multi-step reasoning—capabilities central to agent-based frameworks such as MuseAgent.

4.1 Data Sources

To support agent-oriented evaluation, MuseBench is constructed from high-quality symbolic, visual, and acoustic music data. The benchmark covers a wide range of styles, eras, and difficulty levels, including Baroque, Classical, Romantic, and contemporary piano repertoire. An overview of the dataset is shown in Figure 1, with further source details provided in Appendix B.1.

4.2 Dataset Construction

4.2.1 Preprocessing

We initially collected ∼3,000 candidate scores from multiple open repositories (see Appendix B.1). Scores were filtered for completeness, readability, and resolution quality. After normalization (resolution adjustment, background noise removal, and staff-line correction), ∼600 sheet images were retained. For audio, we collected 513 high-quality performance recordings. Each audio file was standardized to a uniform sampling rate and post-processed (denoising, normalization) to ensure clarity. To avoid copyright infringement, only recordings distributed under public licenses or explicitly provided by musicians with written consent were included.

4.2.2 Annotation

Each sheet music image was paired with an ABC-format symbolic file containing metadata such as title, composer, key, time signature, note durations, and rhythm. Expert musicians further annotated technical difficulty and performance-related elements. Each piece was aligned with professional piano audio recordings and converted into MusicXML with bar-level score–audio alignment, forming a standardized metadata pool for subsequent task construction. All annotations were performed by trained musicians with at least five years of formal music education. To ensure reliability, multiple annotators cross-validated the labels, achieving a Cohen's κ of 0.87.

4.2.3 Definition

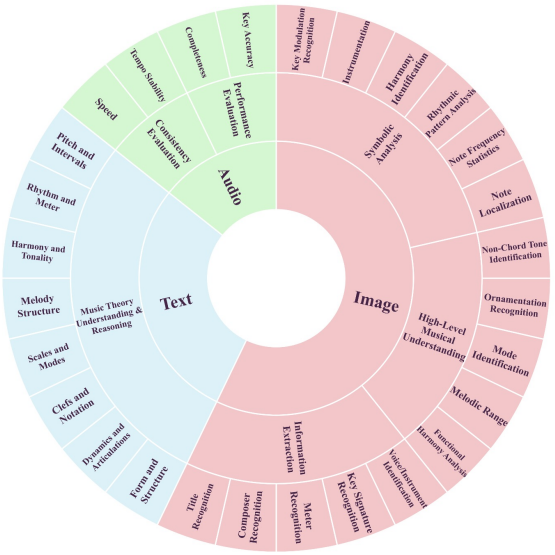

Figure 4: Distribution of questions in MuseBench. It consists of 28 task types across three modalities. Tasks are relatively evenly distributed to ensure balanced evaluation.

To evaluate the capabilities of MLLMs in music understanding and reasoning, we construct **MuseBench**, a benchmark collaboratively designed with expert musicians. It is structured around three core dimensions: (1) **music theory understanding**, (2) **sheet music understanding**, and (3) **performance audio analysis**. Task definitions and evaluation criteria were established under expert consensus to ensure relevance and rigor.

A detailed description of each sub-task and its design rationale is provided in Appendix C. In total, as shown in Figure 4, we define **28 specific tasks** across the three modalities and six sub-dimensions, ensuring a balanced distribution of question–answer pairs and task formats. Our design draws inspiration from evaluation frameworks such as MMMU (Yue et al., 2024) and OmniBench (Li et al., 2024), while explicitly account-

ing for task difficulty and modality diversity to enable robust benchmarking.

5 Experiment

5.1 Comparison on MuseBench

5.1.1 Baseline

We evaluate MuseAgent on MuseBench against 17 representative multimodal language models to examine the effectiveness of agent-based grounding for music understanding. The selected baselines span a wide range of model paradigms, including: (i) **general-purpose language models** such as GPT-4.1, GLM-4, and Phi-4; (ii) **omni-modal models** including GPT-4o, Gemini 2.5-Pro, and Qwen2.5-Omni; (iii) **vision–language models** such as LLaVA, VisualGLM, and Qwen2.5-VL (7B/32B/72B); (iv) **audio-capable models** including Qwen2-audio and MuMu-LLaMA; and (v) **music-specialized models** such as NotaGPT for music notation understanding.

While these models differ in modality coverage and domain specialization, they are primarily designed as monolithic models without explicit agentic orchestration, structured perceptual grounding, or multi-step retrieval-based reasoning. This diverse set of baselines allows us to systematically assess the advantages of MuseAgent's agent-based design over non-agent and weakly grounded multimodal systems on fine-grained music understanding tasks.

5.1.2 Results Analysis

Text Modality. On purely textual tasks, general-purpose language models perform strongly. GPT-4.1 achieves the highest accuracy (86.7%), followed closely by GPT-4o (85.5%) and Gemini 2.5-Pro (83.9%). Scaling model size yields moderate gains, as seen in Qwen2.5-72B (81.4%) and Qwen2.5-32B (79.8%), indicating that music theory questions largely align with the native capabilities of large language models.

Audio Modality. In contrast, general-purpose omni-modal models struggle with performance-level audio reasoning. GPT-4o (55.9%) and Gemini 2.5-Pro (53.1%) perform only marginally above random, while Qwen2.5-Omni (50.6%) and the audio-specialized Qwen2-Audio (51.4%) show similarly limited performance. These results suggest that raw audio inputs alone are insufficient for fine-grained music understanding. By explicitly incorporating AMT and score–audio alignment, MuseAgent equipped with GPT-4.1 achieves 79.1% accuracy, demonstrating that structured perceptual grounding is essential for reasoning about expressive performance.

Table 1: MuseBench performance comparison. MuseAgent significantly improves image and audio understanding over omni-modal and music-specialized baselines.

Modality	Model	Accuracy (%)
Text	GPT-4.1 (OpenAI, 2025)	**86.7**
	GPT-4o (OpenAI, 2023)	85.5
	Gemini2.5-Pro (Comanici et al., 2025)	83.9
	GPT-4.1-mini (OpenAI, 2025)	80.5
	GPT-4.1-nano (OpenAI, 2025)	73.3
	GLM-4-PLUS (GLM et al., 2024)	78.2
	GLM-4-FlashX (GLM et al., 2024)	60.3
	Qwen2.5-72B (Team, 2024)	81.4
	Qwen2.5-32B (Team, 2024)	79.8
	Qwen2.5-7B (Team, 2024)	71.3
	Qwen2.5-Omni-7B (Xu et al., 2025)	63.3
	Phi-4-14B (Abdin et al., 2024)	67.2
	Random	25.0
Audio	MuseAgent (w/ GPT-4.1)	**79.1**
	MuseAgent (w/ GPT-4o-mini)	78.9
	MuseAgent (w/ GLM-4-FlashX)	77.2
	MuseAgent (w/ GPT-4.1-Nano)	63.9
	GPT-4o (OpenAI, 2024)	55.9
	Gemini2.5-Pro (Comanici et al., 2025)	53.1
	MuMu-LLaMA (Liu et al., 2024)	51.7
	Qwen2-audio (Team, 2024)	51.4
	Qwen2.5-Omni-7B (Xu et al., 2025)	50.6
	Random	50.0
Image	MuseAgent (w/ GPT-4.1)	**74.1**
	MuseAgent (w/ GLM-4-FlashX)	72.7
	NotaGPT-7B (Tang et al., 2025)	68.1
	GPT-4.1 (OpenAI, 2025)	66.1
	GPT-4o (OpenAI, 2024)	64.2
	GPT-4.1-mini (OpenAI, 2025)	54.8
	Gemini2.5-Pro (Comanici et al., 2025)	62.1
	Qwen2.5-VL-72B (Team, 2024)	58.9
	Qwen2.5-VL-32B (Team, 2024)	55.7
	Qwen2.5-Omni-7B (Xu et al., 2025)	44.6
	LLaVA-v1.5-13B (Liu et al., 2023a)	38.9
	GLM-4V-9B (GLM et al., 2024)	37.1
	Random	25.0

Image Modality. Vision–language models exhibit substantial difficulty in interpreting symbolic music notation. Models such as LLaVA (38.9%) and GLM-4V (37.1%) perform poorly, reflecting the challenges posed by dense and domain-specific score representations. Larger omni-modal models, including GPT-4o (64.2%) and Gemini 2.5-Pro (62.1%), achieve improved but still limited accuracy, with GPT-4.1 reaching 66.1%. When integrated with the proposed measure-wise OMR (M-OMR) module, MuseAgent attains 74.1% accuracy, outperforming both generalist models and music-specialized baselines such as NotaGPT (68.1%).

5.2 Performance Evaluation of M-OMR Against VLMs

To isolate the contribution of structured visual perception within MuseAgent, we further evaluate the

Figure 5: Image-to-ABC Conversion Comparison. Results are evaluated on the standardized benchmark introduced in NotaGPT (Tang et al., 2025).

Model	Levenshtein Distance
VisualGLM-6B	643.72
DeepSeek-VL-7B-Chat	308.27
LLaVA-v1.5-13B	147.47
LLaVA-v1.6-Vicuna-13B	918.94
Qwen-VL	439.82
NotaGPT-7B	59.47
Gemini-pro-vision	354.30
GPT-4V	655.45
M-OMR (ours)	**18.39**

Figure 6: Comparisons of open-source models and API-based models.

Model	LSA	ROUGE-1	ROUGE-L	METEOR	Avg
InternVL-Chat-v1.5	14.96	19.71	13.32	19.68	16.92
VisualGLM-6B	10.36	21.61	13.21	18.19	15.84
DeepSeek-VL-7B-base	9.92	16.43	11.60	13.81	12.94
InstructBLIP-Vicuna-7B	8.28	22.23	14.93	16.74	15.55
InstructBLIP-Vicuna-13B	8.37	20.29	14.18	14.17	14.25
Qwen-VL	9.58	15.21	10.37	12.56	11.93
Qwen-VL-Chat	9.66	16.80	11.37	14.42	13.06
NotaGPT-7B	12.46	22.63	15.53	18.34	17.24
Gemini-pro-vision	15.88	22.21	15.09	**20.31**	18.37
GPT-4V	14.03	18.49	11.36	19.94	15.96
GPT-4o	**15.92**	18.27	11.35	20.26	16.45
MuseAgent (w/ M-OMR)	15.75	**24.92**	**15.76**	20.17	**19.15**

proposed M-OMR module against state-of-the-art visual language models (VLMs) on music score understanding tasks. Following the evaluation protocol of NotaGPT (Tang et al., 2025), we consider two settings: (i) **closed-set** conversion of sheet music into ABC notation, evaluated using Levenshtein Distance, and (ii) **open-set** visual music analysis, assessed with semantic metrics including LSA, ROUGE, and METEOR. This controlled comparison allows us to assess whether measure-wise symbolic grounding provides advantages over monolithic vision–language modeling for structured music understanding.

5.2.1 Closed-set Image-to-ABC Notation

We evaluate eight representative MLLMs, including API-based models (e.g., GPT-4V, Gemini Pro) and open-source models (e.g., LLaVA, VisualGLM, Qwen-VL-32B, NotaGPT-7B). As shown in Table 5, M-OMR achieves the lowest Levenshtein Distance (18.39), far surpassing all baselines such as NotaGPT-7B (59.47) and LLaVA-13B (147.47). These results demonstrate M-OMR's superior structural accuracy in symbolic notation conversion, highlighting its robustness in closed-set tasks.

5.2.2 Open-set Score Understanding

For open-set tasks, we compare models on semantic similarity and content relevance (Table 6). MuseAgent with M-OMR achieves the best average score (19.15), outperforming both strong API baselines such as GPT-4o (16.45) and Gemini Pro (18.37), as well as open-source vision–language models. Gains are consistent across metrics: higher ROUGE-1 and METEOR reflect better content coverage and fluency, while improved LSA highlights M-OMR's ability to capture nuanced musical semantics. Together, these results establish M-OMR as a robust and reliable solution for score interpretation within MuseAgent.

6 Conclusion

We introduced **MuseBench**, a comprehensive benchmark for multimodal music understanding, and **MuseAgent**, a modular agent that integrates symbolic score parsing and performance audio transcription. MuseBench spans 28 tasks across theory, score, and performance dimensions, offering a rigorous testbed for evaluating the reasoning capabilities of MLLMs. Experiments show that while general-purpose LLMs perform strongly on text-based tasks, they struggle with fine-grained score and audio understanding. By incorporating modality-specific modules such as M-OMR and AMT, MuseAgent achieves substantial gains in both image and audio modalities, demonstrating the necessity of domain-aware perceptual front-ends. These findings highlight the limits of pure scaling in generalist models and confirm the effectiveness of modular integration for complex music reasoning. We hope this work establishes a foundation for future research in AI-assisted music analysis, composition, and education, and for extending multimodal benchmarks beyond text, vision, and speech into the rich domain of music.

Limitations

Our benchmark and model currently focus on piano-related tasks, a deliberate design choice motivated by the availability of large-scale data and the standardized nature of piano notation. While this enables controlled evaluation, it limits direct coverage of other instruments and notational systems. Nevertheless, both MuseBench and MuseAgent are designed to be extensible to other domains given appropriate task-specific data.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. *Preprint*, arXiv:2412.08905.

Andrea Agostinelli, Timo I. Denk, Zal'an Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. 2023. MusicLM: Generating music from text. *arXiv preprint arXiv:2301.11325*.

Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, and 1 others. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Emmanouil Benetos, Simon Dixon, Dimitrios Giannoulis, Hanno Kirchhoff, and Anssi Klapuri. 2019. Automatic music transcription: Challenges and future directions. *Journal of Intelligent Information Systems*, 41(3):407–434.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, and 1 others. 2022. Improving language models by retrieving from trillions of tokens. *arXiv preprint arXiv:2112.04426*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Florian Corrêard, Yuki Wang, Sebastian Ewert, Manuel Moussallam, and Mark Sandler. 2021. Crossmodal transformers for music audio representation learning. *Proceedings of the 22nd ISMIR*.

Zihao Deng, Yinghao Ma, Yudong Liu, and 1 others. 2024. Musilingo: Bridging music and text with pre-trained language models for music captioning and query response. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Slim Essid and Gaël Richard. 2012. Music information retrieval: From symbolic to semantic audio analysis. *IEEE Signal Processing Magazine*, 29(1):118–129.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, and 37 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *Preprint*, arXiv:2406.12793.

Google Gemini Team. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *International Conference on Machine Learning*.

Curtis Hawthorne, Erich Elsen, Jialin Song, Adam Roberts, Ian Simon, Colin Raffel, Jesse Engel, Sageev Oore, and Douglas Eck. 2017. Onsets and frames: Dual-objective piano transcription. *arXiv preprint arXiv:1710.11153*.

Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. 2019a. Enabling factorized piano music modeling and generation with the MAESTRO dataset. In *International Conference on Learning Representations (ICLR)*.

Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, and 1 others. 2019b. Enabling factorized piano music modeling and generation with the maestro dataset. In *International Conference on Learning Representations (ICLR)*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jiawei Huang, Jinglin Liu, Yi Ren, Zhou Zhao, and Shinji Watanabe. 2024. AudioGPT: Understanding and generating speech, music, sound, and talking head. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*. To appear, arXiv:2304.12995.

IMSLP / Petrucci Music Library. International music score library project (imslp). `https://imslp.org`. Accessed: 2025-05-09.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Siddharth Kulkarni, Alessandro Miaschi, Veselin Stoyanov, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*.

Bochen Li, Xinzhao Liu, Karthik Dinesh, and 1 others. 2018. Creating a multi-track classical music performance dataset for multi-modal music analysis: Challenges, insights, and applications. *IEEE Transactions on Multimedia*, 20(9):2382–2395.

Guangyao Li, Yake Wei, Yapeng Tian, and 1 others. 2022. Learning to answer questions in dynamic audio-visual scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5672–5682.

Yizhi Li, Ge Zhang, Yinghao Ma, Ruibin Yuan, Kang Zhu, Hangyu Guo, Yiming Liang, Jiaheng Liu, Zekun Wang, Jian Yang, and 1 others. 2024. Omnibench: Towards the future of universal omni-language models. *arXiv preprint arXiv:2409.15272*.

Haotian Liu, Chunyuan Li, and Qiang Liu. 2023a. Llava: Large language and vision assistant. *arXiv preprint arXiv:2304.08485*.

Jiafeng Liu, Yuanliang Dong, Zehua Cheng, and 1 others. 2022. Symphony generation with permutation invariant language model. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*.

Shansong Liu, Zhenhua Ma, Yixiao Zheng, and 1 others. 2024. Mumu-llama: Multi-modal music understanding and generation via large language models. *arXiv preprint arXiv:2412.06660*.

Zekai Liu, Yiming Yu, Yizeng Li, Chen Sun, Hao Ma, and Tie-Yan Yu. 2023b. Visual rag: Retrieval-augmented generation for vision-language models. *arXiv preprint arXiv:2305.17590*.

Zujie Luo, Shuo Yang, Jingang Yuan, Yankai Lin, and Maosong Sun. 2023. Multimodal retrieval-augmented generation for knowledge-intensive tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

MuseScore Team. Musescore official website. `https://musescore.org`. Accessed: 2025-05-09.

Tomohiko Nakamura, Eita Nakamura, and Shigeki Sagayama. 2015. Real-time audio-to-score alignment of music performances containing errors and arbitrary repeats and skips. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(2):329–339.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

OpenAI. 2024. Gpt-4o mini: advancing cost-efficient intelligence.

OpenAI. 2025. Introducing gpt-4.1 in the api.

Christine Payne. 2019. Musenet. OpenAI Blog.

Christian Schörkhuber and Anssi Klapuri. 2010. Constant-q transform toolbox for music processing. In *7th sound and music computing conference, Barcelona, Spain*, pages 3–64. SMC.

Kurt Shuster, Samuel Humeau, Stephen Roller, Emily Dinan, and Jason Weston. 2022. Language models that seek for knowledge: Modular search & generation for multimodal qa. *arXiv preprint arXiv:2203.05482*.

Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. 2019. The performance of lstm and bilstm in forecasting time series. In *2019 IEEE International conference on big data (Big Data)*, pages 3285–3292. IEEE.

Mingni Tang, Jiajia Li, Lu Yang, Zhiqiang Zhang, Jinghao Tian, Zuchao Li, Lefei Zhang, and Ping Wang. 2025. Nota: Multimodal music notation understanding for visual large language model. *Preprint*, arXiv:2502.14893.

Qwen Team. 2024. Qwen2.5: A party of foundation models. Accessed: 2025-05-15.

The Mutopia Project. The mutopia project: Free sheet music. `http://www.mutopiaproject.org`. Accessed: 2025-05-09.

Rejin Varghese and M Sambath. 2024. Yolov8: A novel object detection algorithm with enhanced performance and robustness. In *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, pages 1–6. IEEE.

Chris Walshaw. 2021. The abc music standard 2.1 (dec 2011).

Benno Weck, Ilaria Manco, Emmanouil Benetos, and Elio Quinton. 2024. Muchomusic: Evaluating music understanding in multimodal audio-language models. *arXiv preprint arXiv:2408.01337*.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, and 1 others. 2025. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*.

Dingyao Yu, Kaitao Song, Peiling Lu, Tianyu He, Xu Tan, Wei Ye, Shikun Zhang, and Jiang Bian. 2023. MusicAgent: An AI agent for music understanding and generation with large language models. *arXiv preprint arXiv:2310.11954*.

Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. 2019. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270.

Ruibin Yuan, Hanfeng Lin, Yi Wang, Zeyue Tian, Shangda Wu, Tianhao Shen, Ge Zhang, Yuhang Wu, Cong Liu, Ziya Zhou, Liumeng Xue, Ziyang Ma, Qin Liu, Tianyu Zheng, Yizhi Li, Yinghao Ma, Yiming Liang, Xiaowei Chi, Ruibo Liu, and 13 others. 2024a. ChatMusician: Understanding and generating music intrinsically with LLM. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6252–6271, Bangkok, Thailand. Association for Computational Linguistics.

Ruibin Yuan, Hanfeng Lin, Yi Wang, Zeyue Tian, Shangda Wu, Tianhao Shen, Ge Zhang, Yuhang Wu, Cong Liu, Ziya Zhou, and 1 others. 2024b. Chatmusician: Understanding and generating music intrinsically with llm. *arXiv preprint arXiv:2402.16153*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, and 1 others. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Li Yujian and Liu Bo. 2007. A normalized levenshtein distance metric. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1091–1095.

Zhiyuan Zhao, Hengrui Kang, Bin Wang, and Conghui He. 2024. Doclayout-yolo: Enhancing document layout analysis through diverse synthetic data and global-to-local adaptive perception. *arXiv preprint arXiv:2410.12628*.

A Appendix

B Dataset Details

B.1 Data Sources and Selection Criteria

To ensure both diversity and legal compliance, we selected music scores that are either (i) public domain works (composers deceased before 1954), or (ii) explicitly released under open licenses such as Creative Commons. No copyrighted material from the last 75 years was included. All recordings either originated from public-domain datasets or were contributed by professional musicians with signed consent agreements.

The dataset includes:

- **Sheet Music Images:**
  - **MuseScore** (MuseScore Team): ∼300 community-created scores (released under CC licenses), including modern and popular music.
  - **IMSLP** (IMSLP / Petrucci Music Library): ∼200 high-resolution classical scores spanning 1600–1920, guaranteed public domain.
  - **Mutopia Project & Project Gutenberg** (The Mutopia Project): ∼100 scores with public licenses, covering canonical works by Bach, Mozart, Beethoven, Chopin.
- **Textual Descriptions:** Metadata for each score includes title, composer, key, meter, and rhythmic structure, automatically extracted from symbolic files and verified by expert annotators.
- **Audio Files:** 513 piano performance recordings across classical, modern, and popular genres. All recordings are either public-domain (older archive sources) or provided directly by performers under Creative Commons licenses.

B.2 Benchmark License and Usage

All components of **MuseBench** are released under strict legal and ethical compliance:

- **Sheet music** is drawn exclusively from public-domain repositories (IMSLP, Mutopia, Project Gutenberg) or from MuseScore where contributors licensed works under Creative Commons. Only works by composers deceased before 1954 are included.
- **Audio recordings** originate from public-domain archives or were directly contributed by professional musicians under written consent and Creative Commons licenses. No copyrighted recordings from the last 75 years are included.
- **Annotations** (ABC, MusicXML, task prompts) were prepared by trained musicians. Annotators provided informed consent, and inter-annotator agreement reached $\kappa = 0.87$.

License: MuseBench is released for non-commercial research and educational purposes under the **CC BY-NC 4.0** license. Redistribution or reuse of individual scores or recordings must comply with the original source licenses. Upon acceptance, we will publicly release all data, annotations, and evaluation scripts.

C Detailed Benchmark Creation

C.0.1 Detailed Task Definition

Music Theory Understanding. This dimension focuses on textual comprehension of symbolic and conceptual music knowledge. It includes two subtasks:

- **Music Theory Recognition:** Evaluates understanding of basic music theory concepts, including key signatures, time signatures, note durations, and rhythmic structures.
- **Music Theory Reasoning:** Involves inferential questions that require deeper reasoning over symbolic descriptions of music, such as determining harmonic progression or identifying musical forms.

Sheet Music Understanding. This dimension assesses the model's ability to interpret notated music from sheet images, and includes:

- **Information Extraction:** Transcription of basic musical metadata such as clefs, key signatures, and tempo markings from visual inputs.

- **Symbolic Analysis:** Understanding note symbols, their spatial and rhythmic relationships, and staff-based structural elements.

- **High-Level Interpretation:** Analyzing expressive or stylistic cues, such as articulation, phrasing, and functional roles in the musical context.

Performance Audio Analysis. This dimension assesses the model's ability to analyze expressive and structural characteristics in real performance recordings. It includes:

- **Performance Evaluation:** Judging the accuracy and completeness of a musical performance, including rhythmic precision, dynamic variation, and articulation clarity.

- **Consistency Evaluation:** Analyzing temporal stability, pitch consistency, and smoothness in expressive transitions across the performance.

C.0.2 Generation

Based on the annotated metadata, we constructed multimodal question–answer pairs. Candidate questions were first generated using a combination of rule-based templates and GPT-4o (OpenAI, 2024) prompting to increase linguistic variety and naturalness. Ground-truth answers were derived deterministically from symbolic metadata through retrieval, calculation, and statistics, ensuring reproducibility. Expert musicians then reviewed and refined all question–answer pairs to guarantee correctness, clarity, and balanced difficulty.

The resulting dataset covers text, image, and audio modalities, offering comprehensive multimodal evaluation. Unlike prior datasets such as MusicTheoryBench (Yuan et al., 2024b), which focus exclusively on text-based symbolic theory questions and include only a few hundred examples, MuseBench introduces multimodal QA tasks that demand joint reasoning over scores, audio, and metadata. This process yields a large-scale benchmark comprising 591 sheet music images, 513 audio recordings, and 2,052 expert-verified question–answer pairs.

C.1 Dataset Compliance and Licensing

To ensure ethical and legal compliance, all components of MuseBench are sourced from either public-domain repositories (e.g., IMSLP, Mutopia, Project Gutenberg) or Creative Commons–licensed platforms (e.g., MuseScore), ensuring full legal compliance. Performance recordings are either public-domain or contributed with explicit consent under CC licenses. Further details on data sources, selection criteria, and license terms are provided in Appendix B.1.

By integrating tasks across text, image, and audio modalities, the **MuseBench** dataset offers a comprehensive evaluation platform for multimodal large language models, spanning every facet of music understanding—from music theory comprehension to the assessment of real performance characteristics.

D Implementation Details of the M-OMR

The M-OMR module bases on a "divide-and-combine" strategy that serves as a visual encoder specialized for music score.

Divide. The input score image is initially segmented into individual measures through a combination of staff line detection and barline localization. Each segmented measure is then treated as an independent visual unit for localized recognition. Specifically, a YOLOv8-based detector (Varghese and Sambath, 2024) is employed to identify and localize each measure,

Process. Each measured image is encoded into a high-dimensional embedding using a ResNet-50 backbone (He et al., 2016), capturing fine-grained visual features of musical symbols, including clefs, staves, barlines, key signatures, and time signatures. These embeddings are then sequentially decoded into ABC-format symbolic sequences using an LSTM-based decoder (Yu et al., 2019) trained for note-level transcription.

Combine. The measure-level symbolic sequences are aggregated to reconstruct the full musical piece. During this step, time signatures extracted during pre-processing are aligned with each measure to ensure consistent rhythmic context. The final output is a well-formed ABC representation that preserves both temporal structure and notational correctness.

Table 2: Defined tasks in MuseBench. The question format is randomly selected from a format pool for each task. The question types "MCQ," and "T/F" represent multiple-choice questions and judge true or false.

Modality	Ability Dimension	Sub-task	Example Question	Type
Text	Music Theory Understanding & Reasoning	Pitch and Intervals	How many half steps are present in an augmented sixth interval?	MCQ
		Rhythm and Meter	Which time signature represents a compound quadruple meter?	MCQ
		Harmony and Tonality	Which pivot chord enables smooth modulation from C major to G major?	MCQ
		Melody Structure	In a period structure, what describes the second phrase that resolves the first?	MCQ
		Scales and Modes	The Dorian mode starting on D is derived from which major scale?	MCQ
		Clefs and Notation	In alto clef, which pitch class is on the fourth space?	MCQ
		Dynamics and Articulations	How should a musician perform staccato notes marked with a crescendo?	MCQ
		Form and Structure	Which structural pattern best describes sonata-rondo form?	MCQ
Image	Information Extraction	Title Recognition	What is the title of the piece?	MCQ
		Composer Recognition	Who is credited as the composer of the piece titled "Classical Rag"?	MCQ
		Meter Recognition	What is the meter signature of the piece titled "The Waltz on my bum"?	MCQ
		Key Signature Recognition	What is the key signature of the piece at the beginning of the score?	MCQ
		Voice/Instrument Identification	Which voices are assigned to the bass clef?	MCQ
	Symbolic Analysis	Note Localization	In which measure does voice V:1 first play a chord containing the note E natural above middle C?	MCQ
		Note Frequency Statistics	In voice V:3, which pitch class appears most frequently as a sounding note (excluding rests and grace notes)?	MCQ
		Rhythmic Pattern Analysis	In the first four measures of the melody line (V:1), which rhythmic pattern is predominantly used?	MCQ
		Harmony Identification	In measure 18, which chord is formed by the soprano (V:1) and bass (V:3)?	MCQ
		Instrumentation	Which staves in this score are written in the bass clef?	MCQ
	High-Level Musical Understanding	Functional Harmony Analysis	What is the harmonic function of the raised A note (â) in E-flat major?	MCQ
		Mode Identification	Considering key signature and accidentals, which mode is implied?	MCQ
		Melodic Range	What is the melodic range of voice V:1?	MCQ
		Ornamentation Recognition	Which ornamentation is most consistently applied?	MCQ
		Non-Chord Tone Identification	In measure 2 of Voice 1, which non-chord tone acts as a passing tone?	MCQ
		Key Modulation Recognition	At which measure does modulation from E-flat major occur?	MCQ
Audio	Performance Evaluation	Key Accuracy	Measure 11 confirms full key accuracy.	T/F
		Completeness	Measure 8 has no missing notes.	T/F
	Consistency Evaluation	Tempo Stability	Measure 11's tempo matches reference stability.	T/F
		Speed	Measure 8's speed is significantly slower than required.	T/F

Recent studies have demonstrated the effectiveness of YOLO-based models in structured document analysis tasks (Zhao et al., 2024).

Then, each segmented measure image is passed through a ResNet-50 (He et al., 2016) encoder to obtain a latent visual embedding $\mathbf{x}_t$. The decoder is implemented as a unidirectional LSTM, which autoregressively generates the corresponding ABC sequence token-by-token.

$$\mathbf{h}_t = \text{LSTM}(\mathbf{h}_{t-1}, \mathbf{x}_t; \theta), \tag{1}$$

After decoding all measures, the symbolic output is reconstructed via:

$$\text{ABC}_{\text{full}} = \text{ConcatMeasures}\left\{(\text{ABC}_i, \text{TimeSig}_i)\right\}_{i=1}^{n}, \tag{2}$$

where TimeSig_i is the pre-detected time signature of measure i, and n is the total number of measures.

Datasets. To construct a robust optical music recognition (OMR) module, we curated a large-scale dataset derived from the MuseScore platform, comprising over 80,000 music scores. Each score was first converted from MusicXML format to ABC notation (Walshaw, 2021), and subsequently rendered into SVG images. To further increase data diversity and model robustness, we performed structured data augmentation by randomly shuffling and replacing ABC bars, resulting in a synthetic corpus of 2.3 million ABC samples. Following image generation, we employed YOLO-based (Varghese and Sambath, 2024) segmentation to automatically detect and extract individual bars from the SVGs, ultimately yielding over **10 million** of image-bar pairs.

Training Configurations. The training was conducted over 100 epochs using a batch size of 12 and a learining rate of 1e-4. Our model achieved near accuracy (approximately 98%) on our held-out validation set, demonstrating both the scale and effectiveness of our training pipeline.

The visualization samples of ABC notation can be found in Figure 7. From the figure, we observe that MuseAgent, equipped with the M-OMR module, is able to accurately transcribe the entire sheet music into ABC notation. In contrast, other large language models struggle to extract complete and precise ABC representations, often missing structural or symbolic details.

E Implementation Details of AMT and Audio-to-Score Alignment

This appendix provides the detailed implementation of the Automatic Music Transcription (AMT) and Audio-to-Score Alignment modules used in

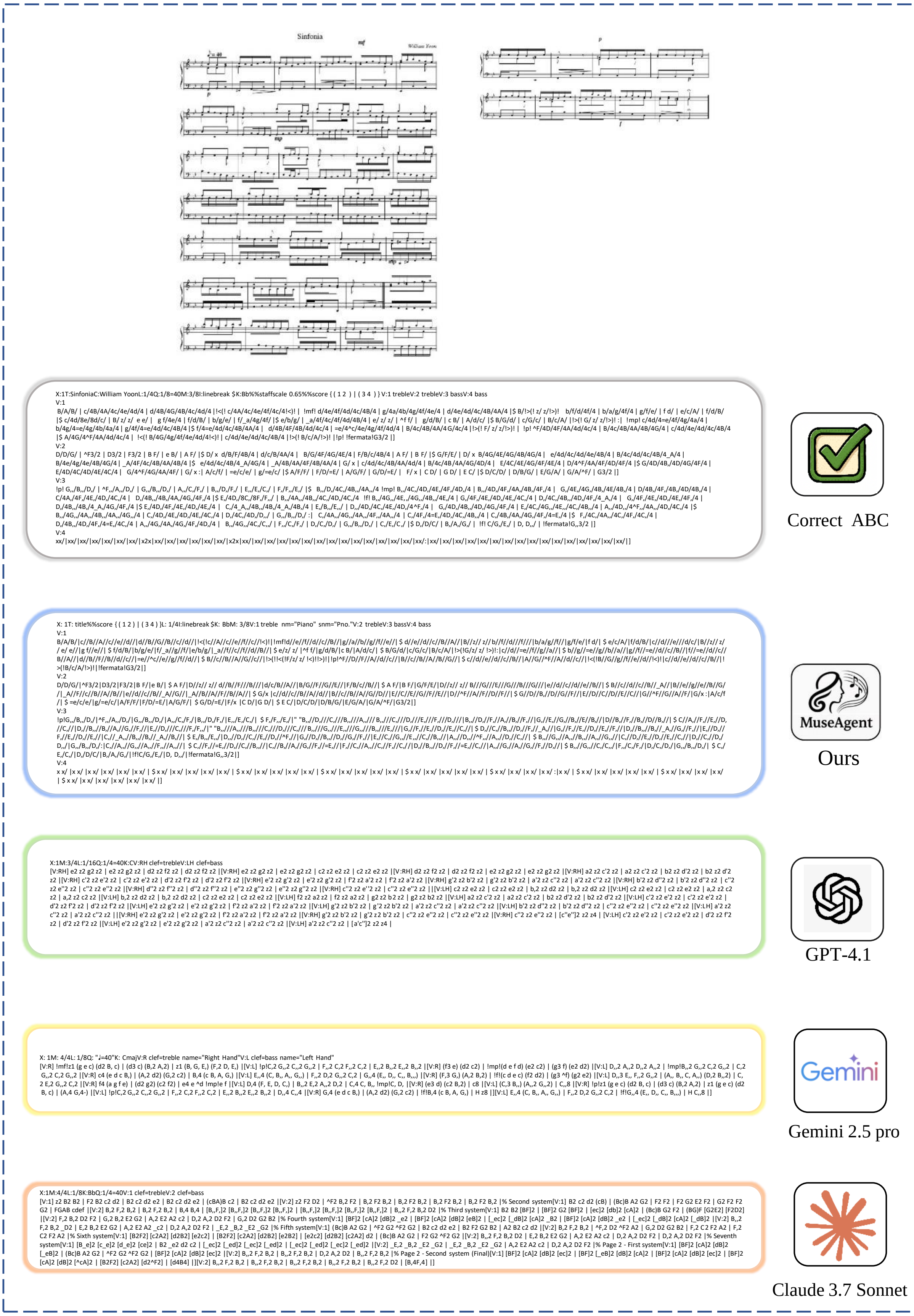

Figure 7: Performance of different LLMs on converting sheet music images to ABC notation

MuseAgent. While we adopt methods inspired by prior work (Hawthorne et al., 2017; Nakamura et al., 2015), we report all relevant architecture and configuration details to facilitate reproducibility and downstream integration.

E.1 Automatic Music Transcription (AMT)

Input Representation. We use the constant-Q transform (CQT) to extract time-frequency features from raw audio. A CNN–BiLSTM (Siami-Namini et al., 2019) architecture predicts both onset and frame-level note activations, following the structure of Onsets-and-Frames (Hawthorne et al., 2017).

Formally, given input features $\mathbf{X}_t$, the onset probability is predicted as:

$$\mathbf{O}_t = \sigma(\text{BiLSTM}(\text{CNN}(\mathbf{X}_t))), \tag{3}$$

and the framewise activation is computed as:

$$\mathbf{F}_t = \sigma(\text{BiLSTM}([\text{CNN}(\mathbf{X}_t), \mathbf{O}_t])). \tag{4}$$

The parameters are listed in Table 3.

Table 3: CQT Configuration for AMT

Parameter	Value
Sample Rate	16 kHz
Hop Length	512 samples
Frequency Bins	88 (covering A0–C8)
Bins per Octave	12
Window Function	Hann
Normalization	Log-magnitude

Network Architecture. The AMT model processes the CQT input through:

- **CNN Frontend:** 3 convolutional layers (kernel size: 3×3, stride: 1, padding: 1), each followed by ReLU and batch normalization.
- **BiLSTM Layer:** One bidirectional LSTM with 128 hidden units per direction.
- **Onset Head:** Fully connected layer with sigmoid activation to predict per-frame note onsets.
- **Frame Head:** Similar layer conditioned on onset features, used to predict framewise note activations.

Training Details.

- **Loss:** Binary cross-entropy loss applied independently to onset and frame predictions.
- **Optimizer:** Adam with learning rate 1×10^{-4}.
- **Training Epochs:** 50 on MAESTRO-V3 (Hawthorne et al., 2019a).
- **Batch Size:** 8.

Post-Processing. Binary predictions are thresholded at 0.5. Onset and frame activations are merged into MIDI note events using the following heuristic: - A note onset is declared if the onset activation exceeds the threshold. - Note duration is extended over consecutive frames with active predictions.

The final output is exported in either MIDI or MusicXML format. We further convert to ABC notation when needed for symbolic alignment.

E.2 Audio-to-Score Alignment

Model Overview. We adopt the dual-layer HMM approach from (Nakamura et al., 2015), which allows robust alignment between symbolic scores and AMT-derived audio events. The alignment process maximizes the posterior probability of the score position p_t given observed acoustic features x_t:

$$p(p_t \mid x_t) = \frac{p(x_t \mid p_t) \cdot p(p_t)}{p(x_t)}. \tag{5}$$

Structure.

- **Top-layer HMM:** Models transitions between score positions (e.g., measures or note groups).
- **Bottom-layer HMM:** Captures fine-grained temporal dynamics within a note (onset, sustain, silence).

Observation Model. The likelihood $p(x_t \mid p_t)$ of observing acoustic feature x_t given score position p_t is modeled by a Gaussian Mixture Model (GMM):

- Number of components: 8
- Covariance: Diagonal
- Input: PCA-reduced CQT (dimension = 30)
- Training: Expectation-Maximization on aligned score–audio pairs

Transition Model. We define a transition matrix A that supports:

- **Self-loop:** Sustains the current note position.
- **Forward transition:** Normal sequential progression.
- **Backward jump:** Repeat sections or corrections.
- **Forward skip:** Skipping sections.

These transitions are encoded as probabilities:

$$A_{ij} = p(p_t = j \mid p_{t-1} = i), \quad \text{with non-zero mass for } |i - j| \geq 1$$

Inference. We use Viterbi decoding to compute the most probable alignment path:

$$p^*_{1:T} = \arg\max_{p_{1:T}} \prod_{t=1}^{T} p(x_t \mid p_t) \cdot A_{p_{t-1}, p_t}$$

This algorithm effectively handles expressive timing, omission, repetition, and incorrect notes, making it robust for alignment in real-world performance scenarios.

E.3 Evaluation Results of AMT and Alignment Algorithms

This section presents the evaluation results of the Automatic Music Transcription (AMT) and alignment algorithms, which are recorded in a JSON format. These results provide a detailed assessment of various transcription metrics, such as overall accuracy, note matching, speed, stability, and tempo synchronization.

The evaluation results for measure 37 of a random sample are summarized in the table below:

Metric	Value
Overall Evaluation (eva_all)	0.9252619743347168
Note Evaluation (eva_note)	1.0
Speed Evaluation (eva_speed)	1.0
Stability Evaluation (eva_stability)	0.7282252907752991
Tempo Synchronization (eva_tempo_sync)	1.0
Extra Notes Count	0
Matched Notes Count	2
Missing Notes Count	0

Table 4: Evaluation Results for Measure 37 of a Random Sample

Additionally, the following figure 8 provides a visual representation of the performance comparison across the various transcription metrics.

F Evaluation Details

F.1 Evaluation Details in our Benchmark

We present the different prompts used for three modalities: text, image, and audio. The following table summarizes the specific prompts for each modality.

F.2 Evaluation Metrics Used in Contrast Experiment

In this appendix, we present the evaluation metrics used in our M-OMR, comparing it with different models for converting images to ABC notation text, utilizing levenshtein distance. Additionally, we analyze music content using semantic similarity and word matching metrics.

Levenshtein Distance. The Levenshtein Distance (Yujian and Bo, 2007) is used as the evaluation metric for converting images to ABC notation text. It refers to the minimum number of single-character operations required to transform model responses into the correct answer sequence.

Let D be a matrix of size $(|R| + 1) \times (|A| + 1)$, where $|R|$ and $|A|$ represent the lengths of the response and answer sequences, respectively. $D[i][j]$ denotes the minimum edit distance between the first i characters of R and the first j characters of A.

The subsequent values of D are computed using the following recurrence relation:

$$D[i][j] = \min \begin{cases} D[i-1][j] + 1 & \text{(delete)} \\ D[i][j-1] + 1 & \text{(insert)} \\ D[i-1][j-1] + \text{cost} & \text{(substitute)} \end{cases}$$

where the cost is 0 if $R[i-1] = A[j-1]$, otherwise it is 1.

Semantic Similarity and Word Matching Metrics. Our Experiment also uses two categories of metrics: semantic similarity and word matching, for analyzing music content.

For semantic similarity, we use Latent Semantic Analysis (LSA), which measures the semantic similarity of text by computing the cosine similarity between vectors. The cosine similarity is given by:

For word matching, we use the following metrics:

- **ROUGE-1**: Calculates the number of unigram matches between the generated and reference text.

Figure 8: Results of Performance Comparison Across Transcription Metrics

- **ROUGE-L**: Measures the longest common subsequence (LCS) match between the generated and reference text.
- **METEOR**: Calculates synonym matches and uses a combination of unigram matches, longest common subsequences, and synonym matches.